

Пономарев В.Ф.

Математическая логика

Учебное пособие

Издание второе, исправленное и дополненное

Рекомендовано Ученым советом университета в качестве учебного пособия по дисциплине «Математическая логика и теория алгоритмов» для студентов, обучающихся по направлению 552800 – «Информатика и вычислительная техника» и по специальности 654600 – «Информатика и вычислительная техника».

Калининград  
2005

ББК 60  
УДК 53:630.11

Пономарев В.Ф. Математическая логика. Учебное пособие. – 2-е изд., испр. т доп. – Калининград: изд-во КГТУ, 2005г., 201с.

В разделе «Классическая логика» дано описание логики высказываний и предикатов. Исследованы возможности вывода дедуктивным методом и методом резолюции. Каждый пример сопровождается графом вывода. Для многих примеров составлены и отлажены программы на языке Prolog. В этом разделе приведены также основы реляционных алгебры и исчисления с переменными-кортежами. Приведены основы языка SQL и его инструкции для исполнения алгебраических и логических операций.

В разделе «Неклассическая логика» изложены основы нечёткой и модальной логик. Показаны особенности исполнения нечётких алгебраических операций и нечёткого вывода в исчислении, приведены примеры экспертных систем. Рассмотрены возможности применения модальных операторов для формального описания того, что «должно быть», и того, что «может быть». Кратко дано описание темпоральной и алгоритмической логик, которые нашли широкое применение в информационных технологиях.

## Оглавление

<b>Предисловие</b> -----	<b>6</b>
<b>Глава 1. Логика классическая</b> -----	<b>9</b>
<i>1.1. Логика высказываний</i> -----	<i>10</i>
1.1.1. Алгебра высказываний -----	12
1.1.1.1. Логические операции -----	13
1.1.1.2. Правила записи сложных формул -----	19
1.1.1.3. Законы алгебры высказываний -----	28
1.1.1.4. Эквивалентные преобразования формул ---	31
1.1.1.5. Нормальные формы формул -----	37
1.1.2. Исчисление высказываний -----	42
1.1.2.1. Интерпретация формул -----	43
1.1.2.2. Аксиомы исчисления высказываний -----	45
1.1.2.3. Метод дедуктивного вывода -----	49
1.1.2.4. Метод резолюции -----	59
<i>Вопросы и задачи</i> -----	<i>67</i>
<i>Расчетно-графическая работа</i> -----	<i>70</i>
<i>1. 2. Логика предикатов</i> -----	<i>72</i>
1.2.1. Алгебра предикатов -----	79
1.2.1.1. Логические операции -----	80
1.2.1.2. Правила записи сложных формул -----	84
1.2.1.3. Законы алгебры предикатов -----	85
1.2.1.4. Эквивалентные преобразования формул ---	89
1.2.1.2. Предварённая нормальная форма -----	91
1.2.1.3. Сколемовская стандартная форма -----	96

1.2.2. Исчисление предикатов-----	99
1.2.2.1. Интерпретация формул -----	99
1.2.2.2. Аксиомы исчисления предикатов-----	102
1.2.2.3. Правила унификации предикатов-----	104
1.2.2.4. Метод дедуктивного вывода -----	105
1.2.2.5. Метод резолюции -----	111
1.2.3. Логическое программирование -----	118
1.2.3.1. Основы логического программирования --	118
1.2.3.2. Подготовка среды Visual Prolog для работы -----	127
1.2.3.3. Описание логических задач на языке Prolog -----	129
<i>Вопросы и задачи -----</i>	<i>160</i>
<i>Расчетно-графическая работа -----</i>	<i>110</i>
<b>1.3. Логика реляционная -----</b>	<b>112</b>
1.3.1. Реляционная алгебра -----	116
1.3.1.1. Унарные операции -----	117
1.3.1.2. Бинарные операции -----	122
1.3.1.3. Правила реляционной алгебры -----	128
1.3.2. Реляционное исчисление -----	129
1.3.3. Языки реляционной логики -----	132
<i>Вопросы и задачи -----</i>	<i>140</i>
<i>Расчетно-графическая работа -----</i>	<i>144</i>
<b>Глава 2. Неклассическая логика-----</b>	<b>155</b>
<b>2.1. Нечёткая логика -----</b>	<b>156</b>
2.1.1. Нечёткие множества -----	156
2.1.2. Нечёткая алгебра -----	166

---

2.1.2.1. Операции над нечёткими множествами ---	166
2.1.2.2. Законы нечёткой алгебры -----	183
2.1.2.3. Свойства нечётких отношений -----	185
2.1.2. Нечёткое исчисление -----	188
4.4.2. Экспертные системы -----	193
<i>Вопросы и задачи -----</i>	<i>201</i>
<i>Расчетно-графическая работа -----</i>	<i>203</i>
2.2. Модальная логика -----	208
2.2.1. Темпоральная (или временная) логика. -----	211
2.2.2. Алгоритмическая логика -----	216
2.2.3. Алгоритмическая логика Хоара -----	219
<i>Ответы и решения -----</i>	<i>176</i>
<i>Литература -----</i>	<i>213</i>
<i>Предметный указатель -----</i>	<i>215</i>

«Упорядочение и упрощение суть  
первые шаги к овладению знанием».  
Томас Манн.

### Предисловие

Логика как наука впервые упоминается в трудах греческого философа Аристотеля (384-322г. до н.э.). С тех времен эта наука учит, как надо правильно рассуждать, правильно делать умозаключения и получать верные суждения. Поэтому было признано, что логика должна содержать счетное и конечное множество правил рассуждения для вывода новых, но верных высказываний. Основное правило, установленное еще в трудах Аристотеля утверждает, что «из двух известных и правильных суждений можно получить новое и верное суждение» [15]. Это правило получило название *силлогизм*<sup>\*</sup>.

Без серьезных изменений эта наука просуществовала более двадцати столетий. Традиционную логику стали подвергать критике лишь в конце XVII века, когда немецкий математик Г. Лейбниц (1646-1716) впервые ввел в логику математические символы и некоторые математические правила. Это позволило заменить некоторые выводы вычислениями. Дальнейшее развитие в логику внесли английский математик Д. Буль (1815-1864), американский математик Пост, немецкие математики Г. Фреге (1848-1925) и Д. Гильберт (1862-1943), итальянский математик Д. Пеано (1858-1932) и многие другие. Так возникла математическая логика как формальная система, носителем ко-

---

<sup>\*</sup> гр. syllogismos – умозаключение, состоящее из двух суждений (посылок), из которых следует третье суждение (вывод),

торой явились символы и последовательности символов, а правила математики позволили дать точное и удобное определение математического суждения<sup>\*\*</sup>.

Применение символов в логике позволило представить суждения философов в компактной и удобной форме, а применение правил математики - дать простые алгоритмы вывода суждений. Центральным понятием, введенным математикой, явилось доказательство – «из положенного с необходимостью вытекает нечто, отличное от положенного» [15].

Во второй половине XX века резко возрос интерес к математической логике. При решении любой задачи нужно, прежде всего, перевести ее смысл и содержание с естественного языка на язык математической логики, с языка математической логики на язык алгоритмов, с языка алгоритмов на язык программирования, а затем применить компьютер для вывода заключения. «Во множестве формальных языков программирования, математической лингвистики и искусственного интеллекта, сменяющихся каждые десять лет, он - язык математической логики - является своего рода скалой среди айсбергов» [15].

В математической логике принято выделять два основных направления: логику классическую и логику неклассическую. Классическая логика является истиннозначной, т.е. любое умозаключение должно иметь только одно значение: «истина» или «ложь» и быть описано через истинные или ложные значения высказываний и суждений. А неклассическая логика во многом снимает эти ограничения и допускает континуальное множество ис-

---

<sup>\*\*</sup> по материалам [10], [15], [21].

тинностных значений высказываний или указывает отношение субъекта высказывания к окружающей действительности (например, к возможности или необходимости в прошлом или будущем появлению факта или события).

Математическая логика нашла широкое применение в информационных технологиях, программировании, математической лингвистике, а также в системах искусственного интеллекта, управления базами данных, в принятии решений и управлении организационно-экономическими системами.

Раздел «Классическая логика» включает главы «Логика высказываний», «Логика предикатов» и «Логика реляционная».

В главе «Логика высказываний» изложены основы алгебры и исчисления высказываний, показаны правила формирования сложных суждений из элементарных высказываний и их преобразования в алгебраические формулы. Исследованы правила исполнения алгебраических операций, поиска равносильных формул и их эквивалентного преобразования. В исчислении высказываний рассмотрены метод дедуктивного вывода и метод вывода по принципу резолюции. Для демонстрации каждого шага вывода использованы графы.

В главе «Логика предикатов» введены новые понятия: предикаты и кванторы, что позволило формировать частные и общие суждения, используя предметные переменные и предметные постоянные. Приведены основные законы алгебры кванторов, показаны способы преобразования алгебраических формул к виду предварённых нормальных и совершенных сколемовских формам. Исследо-

ваны особенности в логике предикатов операций подстановки и унификации. В исчислении предикатов рассмотрены дедуктивный метод вывода заключения и метод по принципу резолюции. Для демонстрации каждого шага вывода также использованы графы. В параграфе 1.2.3 «Логическое программирование», написанном Ю.Ю. Шелухиным под руководством автора, дано описание подготовки среды Visual Prolog для логического вывода и написаны программы для большинства примеров логики высказываний и предикатов. Для ссылки на соответствующие примеры этих глав они нумерованы.

В главе «Реляционная логика» даны основы реляционных алгебры и реляционного исчисления для возможного использования в работе с множеством таблиц-отношений. На конкретных примерах объяснены сложные алгебраические преобразования таблиц-отношений. Изложены основы реляционного исчисления с переменными-кортежами и приведены правила написания логических формул на языке логики предикатов для вывода таблиц-отношений. Приведены основы языка SQL и его инструкции для исполнения алгебраических и логических операций при выводе новых отношений.

Раздел «Неклассическая логика» включает главы «Нечёткая логика» и «Модальная логика».

В главе «Нечёткая логика» введено понятие континуального множества значений «истина» на интервале  $[0, 1]$ , т.е. стало возможным нечётко описывать в высказывании свойства объектов и/или отношений между ними. Показаны особенности исполнения алгебраических операций над нечеткими высказываниями и правила формирования не-

четких сложных суждений. Приведены особенности нечеткого исчисления. Это позволило даже при нечетких знаниях о предмете исследования делать достаточно близкие к истине заключения. Аппарат нечеткой логики нашел широкое применение в экспертных системах и системах искусственного интеллекта. В главе дано краткое описание двух экспертных систем.

В главе «Модальная логика» с помощью кванторов «необходимо» и «возможно» раскрыто отношение субъекта высказывания к ожидаемому или прошедшему факту или событию. Правила вывода модальной логики позволили отобразить структуру вывода заключения на временной оси. Так был осуществлен переход к темпоральной или временной логике для моделирования взаимодействия одного или нескольких линейных и/или ветвящихся процессов. В дальнейшем идеи модальной логики дали развитие алгоритмической логике, правила вывода которой на логическом уровне формализовали семантику языков программирования. Законы темпоральной и алгоритмической логики стали использовать в доказательстве правильности программ (верификации) до их тестирования. Поэтому эти логики нашли применение в информационных технологиях, в параллельных вычислениях, в использовании общих ресурсов вычислительными системами.

«Речи мои представляют отнюдь  
не пачкотню, а вереницу прочно  
упакованных силлогиз-  
мов...»

М. Булгаков. «Мастер и Маргарита».

### Глава 1. Логика классическая

В классической математической логике можно выделить три основных подраздела: логику высказываний, логику предикатов и реляционную логику.

Предметом логики высказываний (propositional logic) являются предложения (или фразы), взятые целиком без учета их длины, состава и структуры и утверждающие или отрицающие какой-либо факт, явление или событие.

Предметом логики предикатов (predicate logic) также являются предложения произвольной длины, но утверждающие или отрицающие какой-либо отдельный (частный) факт или всеобщее явление в зависимости от значений элементов, формирующих состав и структуру предложения.

И наконец, предметом реляционной логики (relation logic) являются множества однородных предложений, имеющих одинаковую длину, одинаковые имена элементов, формирующих одинаковый состав и структуру предложений (совместимых кортежей), объединяемых в таблицы (отношения), но каждое из них утверждает какой-то отдельный факт и соответствующие ему значения имен элементов.

### 1.1. Логика высказываний

Всякое повествовательное предложение естественного языка называют **высказыванием**. Высказыванию, которое описывает верное или неверное суждение о факте, явлении или событии в определенных условиях времени и места, присваивают значение «истина» или «ложь». Например, «3 - простое число» = истина, а «3.14... - рациональное число» = ложь, «Колумб открыл Америку» = истина, а «Киев - столица Узбекистана» = ложь, «Число 6 делится на 2 и 3» = истина, а «сумма чисел 2 и 3 равна 6» = ложь и т.п. Такие высказывания называют простыми или **элементарными**.

При исследовании сложных текстов понятие «простые высказывания» замещают понятием «**пропозициональные\* переменные**», которые обозначают прописными буквами латинского алфавита A, B, C, ... Истинность или ложность пропозициональной переменной будем отмечать символами «и» – истина или «л» – ложь.

#### Пример 1.1.

- если  $A_1 ::= \text{«3 - простое число»}$ , то  $A_1 = \text{и}$ ,
  - если  $A_2 ::= \text{«3 - вещественное число»}$ , то  $A_2 = \text{и}$ ,
  - если  $A_3 ::= \text{«3 - целое число»}$ , то  $A_3 = \text{и}$ ,
  - если  $B_1 ::= \text{«3, 14... - рациональное число»}$ , то  $B_1 = \text{л}$ ,
  - если  $B_2 ::= \text{«3, 14... - не рациональное число»}$ , то  $B_2 = \text{и}$ ,
- и,
- если  $C ::= \text{«Колумб открыл Америку»}$ , то  $C = \text{и}$ ,
  - если  $D ::= \text{«Киев - столица Узбекистана»}$ , то  $D = \text{л}$ ,
  - если  $E_1 ::= \text{«Число 6 делится на 1, 2 и 3»}$ , то  $E_1 = \text{и}$ ,

---

\* лат. propositio – предложение и др.,

•если  $E_2 ::= \text{«Число 6 есть сумма чисел 1, 2, 3»}$ , то  $E_2 =$   
и.

Примечание: символ « $::=$ » означает, что пропозициональная переменная слева есть формальное описание высказывания, стоящего справа, а её значение принадлежит множеству {и, л}.

Высказывания, которые на естественном языке создают из элементарных высказываний с помощью грамматических связок «не», «и», «или», «если..., то...», «... тогда и только тогда, когда...» и т.п. называют **сложными**. Для формального обозначения этих связок вводят символы, которые называют **логическими связками**. Например, « $\vee$ » ::= «или», « $\&$ » ::= «и», « $\neg$ » ::= «не», « $\rightarrow$ » ::= «если..., то...», « $\leftrightarrow$ » ::= «...тогда и только тогда, когда...». На естественном языке символы математической логики называют так: « $\vee$ » - *дизъюнкция*<sup>\*</sup>, « $\&$ » - *конъюнкция*<sup>\*\*</sup>, « $\neg$ » - *отрицание*, « $\rightarrow$ » - *импликация*<sup>\*\*\*</sup>, « $\leftrightarrow$ » - *эквивалентность*<sup>\*\*\*\*</sup>.

Для построения более сложных высказываний используют вспомогательные символы «(», «)» - скобки.

Пример 1.2. Пусть обозначения элементарных высказываний  $A_1, A_2, A_3, B_1, B_2, E_1, E_2$  взяты из предыдущего примера.

•если высказывание «3 – вещественное и целое число», то  $(A_1 \& A_2) =$  и,

•если высказывание «3,14... - рациональное число»,

---

\* лат. disjunctivus – разделительный,

\*\* лат. conjunctivus – соединительный,

\*\*\* лат. implicatio – сплетение, переплетение,

\*\*\*\* лат. aequus – равный + valentis, – имеющий значение, цену

то  $V_1 = \text{л}$  или  $\neg V_1 = \text{и}$ ,

- если высказывание «число 6 делится на 1, 2, 3 и представляет сумму делителей 1, 2, 3», то  $(E_1 \& E_2) = \text{и}$ ,

- если высказывание «если 3 - целое число, то оно вещественное», то  $(A_3 \rightarrow A_2) = \text{и}$ ,

- если высказывание «если 3 – простое число, то оно целое», то  $(A_1 \rightarrow A_3) = \text{и}$ ,

- если высказывание «3 - простое число, тогда и только тогда, когда оно целое», то  $(A_1 \leftrightarrow A_2) = \text{и}$ .

Логические связки обозначают исполнение *логических операций* над высказываниями. Поэтому правила исполнения таких операций формируют *алгебру высказываний*. Правила построения сложных высказываний в виде последовательности пропозициональных переменных, логических связок и вспомогательных символов определяют возможность формального описания многих высказываний и суждений естественного языка. Но при формальном описании сложного высказывания всегда нужно внимательно следить за смыслом и содержанием высказывания.

Правила вывода новых высказываний, основанные на известных высказываниях и отношениях между ними, представляют *исчисление высказываний*. В этом случае высказывания, из которых делают вывод новых высказываний, называют *посылками*, а получаемое новое высказывание – *заключением*.

### 1.1.1. Алгебра высказываний

Множество пропозициональных переменных  $T = \{A, B, C, \dots\}$  и множество логических операторов и отношений  $\Sigma = \{\neg, \&, \vee, \rightarrow, \leftrightarrow, \equiv\}$  представляют алгебру высказыва-

ний:

$$A_{\text{выск.}} = \langle T, \Sigma, \text{и}, \text{л} \rangle,$$

где и, л – значения «истина» или «ложь» для пропозициональных переменных и правильно построенных алгебраических выражений. Символы логических операций: « $\neg$ » - отрицание, « $\&$ » - конъюнкция, « $\vee$ » - дизъюнкция, « $\rightarrow$ » - импликация, « $\leftrightarrow$ » - эквивалентность служат для формального описания сложных высказываний, а символ « $\equiv$ » - равносильности формирует отношение сравнения двух правильно построенных выражений.

Любая пропозициональная переменная есть *элементарная формула*, т. е.  $A_i = F_i$ .

Высказывание, которое может быть получено из элементарных высказываний посредством логических операций, называют *формулой* алгебры логики, т.е. если  $F_1$  и  $F_2$  – формулы, то  $\neg F_1$ ,  $\neg F_2$ ,  $(F_1 \& F_2)$ ,  $(F_1 \vee F_2)$ ,  $(F_1 \rightarrow F_2)$  и  $(F_1 \leftrightarrow F_2)$ ,  $(F_1 \equiv F_2)$  - также формулы.

Никаких других формул в логике высказываний нет.

Значение формулы полностью определяется значениями входящих в нее пропозициональных переменных.

#### 1.1.1.1. Логические операции

Логические операции бывают *унарными* (или одноместными) и *бинарными* (или двухместными). Если значения операндов принадлежат множеству {и, л}, то и результаты логических операций также принадлежат множеству {и, л}. Исполнение логических операций удобно описывать *таблицами истинности*.

**Отрицание** ( $\neg F$ ) есть одноместная операция, посред-

ством которой ее значение есть отрицание значения операнда.

В программировании для этого используют оператор NOT, т.е. (NOT F). Если F - высказывание, то  $\neg F$  также высказывание. Если  $\neg F$  есть высказывание, то  $\neg(\neg F)$  также есть высказывание.

F	$\neg F$
и	л
л	и

**Пример 1.3.** Верно ли, что  $A := \langle\langle 4 - \text{простое число} \rangle\rangle$  или  $A = \text{и}$ ?  
 «Нет, неверно, что 4 – простое число»,  
 т.е.  $\neg A = \text{и}$ .

**Конъюнкция** ( $F_1 \& F_2$ ) есть двухместная операция, посредством которой из двух формул  $F_1$  и  $F_2$  получают новую формулу  $F$ , описывающую сложное высказывание. Формула  $F = (F_1 \& F_2)$  истинна тогда и только тогда, когда истинны значения двух операндов  $F_1$  и  $F_2$ .

$F_1$	$F_2$	$F_1 \& F_2$
		и
л	л	л
л	и	л
и	л	л
и	и	и

В программировании для этого используют оператор AND, т.е. ( $F_1\_AND\_F_2$ ).

На естественном языке эта операция описывается соединительными словами: «... и ...», «... также ...», «как ..., так...», «... несмотря на ...» и др.

Из определения операций конъюнкции и отрицания очевидно, что  $(F \& \neg F) = \text{л}$ .

Если даны  $F_1, F_2, \dots, F_n$ , то формула  $F = (F_1 \& F_2 \& \dots \& F_n) = \text{и}$  тогда и только тогда, когда истинны все формулы  $F_1, F_2, \dots, F_n$ .

**Пример 1.4.** Даны высказывания  $A := \langle\langle \text{компьютер содержит основной микропроцессор} \rangle\rangle$ ,  $B := \langle\langle \text{компьютер содержит оперативную память} \rangle\rangle$ ,  $C := \langle\langle \text{компьютер содержит контроллеры} \rangle\rangle$ ,  $D := \langle\langle \text{компьютер содержит порты ввода –} \rangle\rangle$

вывода».

Тогда формула  $F=(A\&B\&C\&D)$  отражает высказывание «компьютер содержит основной микропроцессор, оперативную память, контроллеры и порты ввода-вывода» [17].

**Дизъюнкция ( $F_1\vee F_2$ )** есть двухместная операция, посредством которой из двух формул  $F_1$  и  $F_2$  получают новую формулу  $F$ , описывающую сложное высказывание. Формула  $F=(F_1\vee F_2)$  ложно тогда и только тогда, когда ложны значения двух операндов  $F_1$  или  $F_2$ .

F 1	F 2	$F_1\vee F_2$ 2
л	л	л
л	и	и
и	л	и
и	и	и

В программировании для этого используют оператор OR, т.е. ( $F_1\_OR\_F_2$ ).

Из определения операций дизъюнкции и отрицания, очевидно, что  $(F\vee\neg F)=и$ .

Если даны  $F_1, F_2, \dots, F_n$ , то формула  $F=(F_1\vee F_2\vee \dots \vee F_n)=л$  тогда и только тогда когда ложны все формулы  $F_1, F_2, \dots, F_n$ .

Следует обратить внимание, что в повседневной речи союз «или» употребляется в двух смыслах: «**исключающее или**», когда истинность сложного высказывания определяется истинностью только одного высказываний (второе должно быть ложным), и «**не исключающее или**», когда истинность сложного высказывания определяется истинностью хотя бы одного из них. В математической логике используют только «**не исключающее или**».

Пример 1.5. Даны высказывания  $A:=$  «в компьютере применяют матричный принтер»,  $B:=$  «в компьютере применяют струйный принтер»,  $C:=$  «в компьютере применяют лазерный принтер»,  $D:=$  «в компьютере применяют ли-терный принтер».

Тогда формула  $F=(A\vee B\vee C\vee D)$  отражает высказывание «в компьютере применяют матричный, струйный, лазерный или литерный принтеры» [17].

Пример 1.6. Даны высказывания  $A:=$  «монитор есть машинная программа, которая наблюдает, регулирует, контролирует или проверяет операции в системе обработки данных»,  $B:=$  «монитор в языках программирования - это высокоуровневый механизм взаимодействия и синхронизации процессов, обеспечивающих доступ к неразделяемым ресурсам»,  $C:=$  «монитор - это дисплей, используемый для контроля процессов и управления системой».

Тогда формула  $F=(A\vee B\vee C)$  отражает высказывание «монитор есть машинная программа, которая наблюдает, регулирует, контролирует или проверяет операции в системе обработки данных или в языках программирования – это высокоуровневый механизм взаимодействия и синхронизации процессов, обеспечивающих доступ к неразделяемым ресурсам или дисплей, используемый для контроля процессов и управления системой» [17].

**Импликация ( $F_1\rightarrow F_2$ )** есть двуместная операция, посредством которой из двух формул  $F_1$  и  $F_2$  получают формулу  $F$ , описывающую сложное высказывание. Формула  $F=(F_1\rightarrow F_2)$  ложно тогда и только тогда, когда истинно  $F_1$  и ложно  $F_2$ . При этом  $F_1$  называют *посылкой*, а  $F_2$  – *заключением*.

В программировании используют оператор IMPLIES, т. е. ( $F_1\_IMPLIES\_F_2$ ).

F <sub>1</sub>	F <sub>2</sub>	F <sub>1</sub> →F <sub>2</sub>
л	л	и
л	и	и
и	л	л
и	и	и

На естественном языке эта операция выражается словами «если ..., то ...», «тогда ..., когда ...», «постольку ..., поскольку ...», «при наличии ..., следует ...», «из ... следует ...», «... влечет ...» и т. п. Высказывание, находящееся слева от логической связки  $\rightarrow$ , называют *условием* или *посылкой*, а справа – *заключением* или *следствием* выражениями.

Импликация играет важную роль в математической логике, так как позволяет формировать цепочку рассуждения при доказательстве истинности заключения. Использование импликации в языке математической логики отличается от ее употребления в обыденной речи на естественном языке. Если посылка в обыденной речи ложная, то импликация вообще не имеет смысла. А если посылка – истинная, то импликация имеет смысл только при истинном значении заключения.

Пример 1.7. Даны высказывания  $A :=$  «по проводнику протекает электрический ток» и  $B :=$  «вокруг проводника есть магнитное поле». Тогда формула  $F = (A \rightarrow B)$  отражает высказывание «если по проводнику протекает электрический ток, то вокруг проводника возникает магнитное поле».

Пример 1.8. Пусть даны высказывания  $A :=$  «на упругое тело оказывают влияние внешние силы» и  $B :=$  «в упругом теле возникают внутренние силы, препятствующие изменению формы». Тогда формула  $F = (A \rightarrow B)$  отражает

высказывание «если на упругое тело оказывают влияние внешней силы, то в нем возникают внутренние силы, препятствующие изменению формы».

**Эквивалентность** ( $F_1 \leftrightarrow F_2$ ) есть двухместная операция, посредством которой из двух формул  $F_1$  и  $F_2$  получают формулу  $F$ , описывающую сложное высказывание. Формула  $F=(F_1 \leftrightarrow F_2)$  истинна тогда и только тогда, когда оба операнда  $F_1$  и  $F_2$  имеют одинаковые значения.

F <sub>1</sub>	F <sub>2</sub>	$F_1 \leftrightarrow F_2$
л	л	и
л	и	л
и	л	л
и	и	и

В программировании для этого используют оператор IFF, т.е. ( $F_1\_IFF\_F_2$ ). На естественном языке эквивалентность выражается фразами «... эквивалентно ...», «для того чтобы ..., необходимо и достаточно, чтобы ...», «... лишь при условии ...», «..., если и только если ...», и т.п.

Пример 1.9. Даны высказывания  $A:=$  «быть четным числом» и  $B:=$  «число делится на два». Тогда формула  $F=(A \leftrightarrow B)$  отображает высказывание «для того, чтобы число было четным необходимо и достаточно, чтобы оно делилось на два».

Пример. Даны высказывания  $A:=$  «выполнить загрузку в компьютер операционной системы» и  $B:=$  «установить в компьютер дискету с записанной операционной системой».

Тогда формула  $F=(A \leftrightarrow B)$  отображает высказывание «для того, чтобы выполнить загрузку операционной системы в компьютер, необходимо и достаточно установить в компьютер дискету с записанной операционной системой».

Пример 1.10. Даны высказывания  $A :=$  «урожай будет стабильным ежегодно» и  $B :=$  «выполнены все ирригационные работы».

Тогда формула  $F = (A \leftrightarrow B)$  отображает высказывание «урожай будет ежегодно стабильным тогда и только тогда, когда будут выполнены все ирригационные работы».

Пример 1.11. Даны высказывания  $S :=$  «полная система функций математической логики»,  $A :=$  «система функций содержит хотя бы одну нелинейную функцию»,  $B :=$  «система функций содержит хотя бы одну немонотонную функцию»,  $C :=$  «система функций содержит хотя бы одну не самодвойственную функцию»,  $D :=$  «система функций содержит хотя бы одну функцию, не сохраняющую «0»»,  $E :=$  «система функций содержит хотя бы одну функцию, не сохраняющую «1»». Тогда формула  $F = (S \leftrightarrow (A \& B \& C \& D \& E))$  есть образ сложного высказывания: «для того чтобы система функций математической логики была полной, необходимо и достаточно, чтобы она содержала хотя бы по одной нелинейную, немонотонную и не самодвойственную функции, а также функции, не сохраняющие «0» и «1»».

#### 1.1.1.2. Правила записи сложных формул

Для определения истинности сложного суждения необходимо анализировать значение истинности каждого элементарного высказывания и вычислять значения истинности каждой подформулы, входящей в формулу этого суждения.

Это удобно вычислять с помощью **таблиц истинности**, столбцами которой являются имена всех пропозициональных переменных и подформул формулы сужде-

ния, а в строках таблицы указывают значения истинности подформул для всевозможных наборов значений истинности пропозициональных переменных.

Пример 1.12. Суждение «если инвестиции на текущий год не изменятся, то возрастает расходная часть бюджета или возникнет безработица, а если возрастет расходная часть бюджета, то налоги не будут снижены и, наконец, если налоги не будут снижены и инвестиции не изменятся, то безработица не возникнет» [21].

В этом суждении есть четыре простых повествовательных предложения: A:= «инвестиции на текущий год не изменятся», B:= «возрастет расходная часть бюджета», C:= «возникнет безработица», D:= «налоги не снизятся». Тогда формула суждения имеет вид:

$$F = (A \rightarrow (B \vee C)) \& (B \rightarrow D) \& ((D \& A) \rightarrow \neg C).$$

Таблица 1.1

A	B	C	D	$\neg C$	$A \& B$	$B \vee C$	$A \rightarrow B \vee C$	$B \rightarrow D$	$D \& A$	$\neg(D \& A) \vee \neg C$	$F$
1	2	3	4	5	6	7	8	9	10	11	12
л	л	л	л	и	л	л	и	и	и	и	<b>и</b>
л	л	л	и	и	л	л	и	и	и	и	<b>и</b>
л	л	и	л	л	л	и	и	и	и	и	<b>и</b>
л	л	и	и	л	л	и	и	и	и	и	<b>и</b>
л	и	л	л	и	л	и	и	л	и	л	л
л	и	л	и	и	л	и	и	и	и	и	<b>и</b>
л	и	и	л	л	л	и	и	л	и	л	л
л	и	и	и	л	л	и	и	и	и	и	<b>и</b>
и	л	л	л	и	л	л	л	и	и	л	л
и	л	л	и	и	и	л	л	и	и	л	л
и	л	и	л	л	л	и	и	и	и	и	<b>и</b>

И	Л	И	И	Л	И	И	И	И	Л	И	Л
И	И	Л	Л	И	Л	И	И	Л	И	Л	Л
И	И	Л	И	И	И	И	И	И	И	И	И
И	И	И	Л	Л	Л	И	И	Л	И	Л	Л
И	И	И	И	Л	И	И	И	И	Л	И	Л

Для удобства описания формулы таблицей истинности каждый столбец таблицы 1.1 пронумерован, а исполнение логических операций (подформул) выполняется с индексами столбцов.

В 12-ом столбце таблицы выделены полужирным шрифтом те строки, в которых формула  $F$  имеет значение «истины» для различных наборов значений пропозициональных переменных  $A, B, C$  и  $D$ .

Анализ таблицы показывает: чтобы не возникла безработица при неизменных инвестициях и налогах необходимо увеличить расходную часть бюджета. Это показано в таблице выделенной строкой.

Пример 1.13. Суждение «если цены высокие ( $A$ ), то и заработная плата должна быть также высокой ( $B$ ). Цены повышают или применяют регулирование цен ( $C$ ). Если применяют регулирование цен, то нет инфляции ( $\neg D$ ). Инфляция есть. Следовательно, заработная плата должна быть высокой» [19].

Формулы первых четырех высказываний  $(A \rightarrow B)$ ,  $(A \vee C)$ ,  $(C \rightarrow \neg D)$ ,  $D$  есть посылки, а формула пятого высказывания – заключение  $B$ . Посылки и заключение разделены между собой чертой:

$$\frac{(A \rightarrow B), (A \vee C), (C \rightarrow \neg D), D}{B}$$

Для анализа этого суждения построим таблицу истинности

(см. табл. 1.2).

Таблица 1.2

A	B	C	D	$1 \rightarrow$ 2	$1 \vee$ 3	$\neg 4$ 7	$3 \rightarrow$ 7
1	2	3	4	5	6	7	8
л	л	л	л	и	л	и	и
л	л	л	и	и	л	л	и
л	л	и	л	и	и	и	и
л	л	и	и	и	и	л	л
л	и	л	л	и	л	и	и
л	и	л	и	и	л	л	и
л	и	и	л	и	и	и	и
л	и	и	и	и	и	л	л
и	л	л	л	л	и	и	и
и	л	л	и	л	и	л	и
и	л	и	л	л	и	и	и
и	л	и	и	л	и	л	л
и	и	л	л	и	и	и	и
<b>и</b>	<b>и</b>	<b>л</b>	<b>и</b>	<b>и</b>	<b>и</b>	<b>л</b>	<b>и</b>
и	и	и	л	и	и	и	и
и	и	и	и	и	и	л	л

Анализ выделенной строки таблицы показывает, что заработная плата при высоких ценах и наличии инфляции должна быть высокой и не должно быть регулирования цен. Только в этом случае посылки  $(A \rightarrow B)$ ,  $(A \vee C)$ ,  $(C \rightarrow \neg D)$  и D и заключение B имеют значение истины.

Пример 1.14. Суждение: "Контракт будет выполнен (A) тогда и только тогда, когда дом будет сдан в эксплуа-

тацию (B). Если дом будет сдан в декабре, то в январе можно переезжать в новые квартиры (C). Если в январе квартиросъемщики не переезжают, то они не оплачивают квартирную плату. Даже если контракт не выполнен, то квартиросъемщики должны внести квартирную плату. Квартиросъемщики внесут квартирную плату”[19].

В этом суждении пять высказываний. Формулы первых четырех высказываний  $(A \leftrightarrow B)$ ,  $(B \rightarrow C)$ ,  $(\neg C \rightarrow \neg D)$  и  $(\neg A \rightarrow D)$  есть посылки, а формула пятого высказывания – заключение D. Посылки и заключение также разделены между собой чертой.

$$\frac{(A \leftrightarrow B), (B \rightarrow C), (\neg C \rightarrow \neg D), (\neg A \rightarrow D)}{D}$$

Анализ выделенной строки таблицы 1.3 показывает, при каких условиях (посылках) заключение должно быть истинным D.

Таблица 1.3

A	B	C	D	$\neg A$	$\neg B$	$\neg C$	$\neg D$
1	2	3	4	5	6	7	8
п	п	п	п	и	и	и	п
п	п	п	и	и	и	п	и
п	п	и	п	и	и	и	п
п	п	и	и	и	и	и	и
п	и	п	п	п	п	и	п
п	и	п	и	п	п	п	и
п	и	и	п	п	и	и	п
п	и	и	и	п	и	и	и
и	п	п	п	п	и	и	и
и	п	п	и	п	и	п	и
и	п	и	п	п	и	и	и
и	п	и	и	п	и	и	и
и	и	п	п	и	п	и	и
и	и	п	и	и	п	п	и
и	и	и	п	и	и	и	и

Таблица 1.3

A	B	C	D	$\neg A$	$\neg B$	$\neg C$	$\neg D$
И	И	И	И	И	И	И	И

Пример 1.15. Суждение «если курс ценных бумаг возрастет (A) или процентная ставка снизится (B), то курс акций упадет (C) или налоги не повысятся (D). Курс акций падает тогда и только тогда, когда растут курс ценных бумаг и налоги. Если процентная ставка снизится, то либо курс акций не понизится, либо курс ценных бумаг не возрастет. Следовательно, если налоги повысить, то не вырастет курс ценных бумаг и вырастет курс акций» [19].

В этом суждении четыре сложных высказывания, три из которых являются посылками  $(A \vee B) \rightarrow (C \vee D)$ ,  $C \leftrightarrow (A \& \neg D)$  и  $B \rightarrow (\neg C \vee \neg A)$ , а одно – заключением  $\neg D \rightarrow (\neg A \& \neg C)$ . Посылки и заключение также разделены между собой чертой.

$$\frac{(A \vee B) \rightarrow (C \vee D), C \leftrightarrow (A \& \neg D), B \rightarrow (\neg C \vee \neg A)}{\neg D \rightarrow (\neg A \& \neg C)}$$

Для анализа этого суждения построим таблицу истинности (см. табл. 1.4).

В таблице приведены все возможные наборы пропозициональных переменных (элементарных высказываний) и всех промежуточных формул посылок и заключения. Выделенные строки таблицы показывают варианты, для каких истинны посылки и заключение.

Для выбора только одного варианта из шести возможных следует рассмотреть вариант, когда курс ценных бумаг растет ( $A=и$ ). При этом курс ценных бумаг может расти либо падать

Таблица 1.4

A	B	C	D	$1 \vee 1 \& \neg 3 \vee 5 \rightarrow 3 \leftrightarrow \neg 3 \vee \neg 1 2 \rightarrow 1 \neg 1 \& \neg 4 \rightarrow$
				2 4 4 7 6 0 3 12
1	2	3	4	5 6 7 8 9 10 11 12 13
л	л	л	л	л л л и и и и и и
л	л	л	и	л л и и и и и и и
л	л	и	л	л л и и л и и л л
л	л	и	и	л л и и л и и л и
л	и	л	л	и л и и и и и и и
л	и	л	и	и л и и и и и и и
л	и	и	л	и л и и л и и и л л
л	и	и	и	и л и и л и и и л и
и	л	л	л	и и л л л л и и л л
и	л	л	и	и и л и и и и и и и
и	и	л	л	и и л л л л и и л л
и	и	л	и	и л и и и и и и и л и
и	и	и	л	и и и и и л л л л
и	и	и	и	и л и и л л л л и

Приведенные примеры позволяют сформулировать некоторые правила записи сложных суждений:

- каждое вхождение логической связки « $\rightarrow$ » относится к пропозициональной переменной или формуле, следующей непосредственно за логической связкой справа,
- каждое вхождение логической связки « $\&$ » после расстановки скобок связывает пропозициональные переменные или формулы, непосредственно окружающие логическую связку,
- каждое вхождение логической связки « $\vee$ » после расста-

новки скобок связывает пропозициональные переменные или формулы, непосредственно окружающие эту связку и т.д.,

- в формулах нет двух рядом стоящих логических связок - они должны быть разъединены формулами,
- в формулах нет двух рядом стоящих формул - они должны быть разъединены логической связкой,
- логические связки по силе и значимости упорядочены так:  $\neg$ ,  $\&$ ,  $\vee$ ,  $\rightarrow$ ,  $\leftrightarrow$ , т.е. самой сильной связкой является отрицание, затем конъюнкция, дизъюнкция, импликация и, наконец, эквивалентность.

Знания силы логических связок позволяет опускать скобки, без которых очевиден порядок исполнения операций.

Пример 1.16. Пусть дана формула  $F = (((F_1 \vee (\neg F_2)) \rightarrow F_3) \leftrightarrow F_4)$ .

Зная силу логических связок, упростить запись формулы.

- удалить внешние скобки формулы, так как они не определяют старшинство никакой операции:

$$F = ((F_1 \vee (\neg F_2)) \rightarrow F_3) \leftrightarrow F_4,$$

- удалить скобки, охватывающие формулу импликации, так как операция эквивалентности будет исполняться только после выполнения операции импликации:

$$F = (F_1 \vee (\neg F_2)) \rightarrow F_3 \leftrightarrow F_4,$$

- удалить скобки, охватывающие формулу дизъюнкции, так как операция импликации будет исполняться только после выполнения операции дизъюнкции:

$$F = F_1 \vee (\neg F_2) \rightarrow F_3 \leftrightarrow F_4,$$

- удалить скобки, охватывающие формулу отрицания, так как операция дизъюнкции будет исполняться только после выполнения операции отрицания:

$$F = F_1 \vee \neg F_2 \rightarrow F_3 \leftrightarrow F_4,$$

Последовательность исполнения операций определяется силой логических связок: сначала вычислить значение  $(\neg F_2)$ , затем  $(F_1 \vee (\neg F_2))$ ,  $((F_1 \vee (\neg F_2)) \rightarrow F_3)$  и, наконец,  $- ((F_1 \vee (\neg F_2)) \rightarrow F_3) \leftrightarrow F_4$ .

Пример 1.17.

Дана формула

$F = F_1 \& F_2 \& F_3 \vee \neg F_1 \rightarrow F_3 \leftrightarrow F_1$ . Расставить вспомогательные символы - скобки.

- поставить скобки для операции отрицания:

$$F_1 \& F_2 \& F_3 \vee (\neg F_1) \rightarrow F_3 \leftrightarrow F_1;$$

- поставить скобки для операции конъюнкция:

$$F = ((F_1 \& F_2) \& F_3) \vee (\neg F_1) \rightarrow F_3 \leftrightarrow F_1;$$

- поставить скобки для операции дизъюнкции:

$$F = (((F_1 \& F_2) \& F_3) \vee (\neg F_1)) \rightarrow F_3 \leftrightarrow F_1;$$

- поставить скобки для операции импликация:

$$F = (((((F_1 \& F_2) \& F_3) \vee (\neg F_1)) \rightarrow F_3) \leftrightarrow F_1);$$

- поставить скобки для операции эквивалентности:

$$F = ((((((F_1 \& F_2) \& F_3) \vee (\neg F_1)) \rightarrow F_3) \leftrightarrow F_1).$$

Скобки позволяют организовать вычисление значения всей формулы, последовательно определяя значение каждой вложенной подформулы: сначала найти значение  $(\neg F_1)$ , затем  $- (F_1 \& F_2)$ ,  $((F_1 \& F_2) \& F_3)$ ,  $((F_1 \& F_2) \& F_3) \vee (\neg F_1)$ ,  $((F_1 \& F_2) \& F_3) \vee (\neg F_1) \rightarrow F_3$ , и наконец,  $- (((F_1 \& F_2) \& F_3) \vee (\neg F_1)) \rightarrow F_3 \leftrightarrow F_1$ .

## 1.1.1.3. Законы алгебры высказываний

Ещё во времена Аристотеля в классической логике были признаны два основных закона: **закон тождества** – «используемые понятия не должны ни изменяться, ни подменяться в ходе одного и того же рассуждения» и **закон непротиворечия** – «невозможно, чтобы одно и то же в одно и то же время было и не было присуще одному и тому же в одном и том же отношении» [15]. С внедрением в логику высказываний алгебраических методов потребовались аксиомы, представляющие законы алгебры высказываний.

Две формулы  $F_i$  и  $F_j$  называют **равносильными** ( $F_1 \equiv F_2$ ), если они имеют одинаковое значение «истина» или «ложь» при одинаковых наборах пропозициональных переменных. Если две формулы равносильны  $F_1 \equiv F_2$ , то они эквивалентны между собой, т.е.  $(F_1 \leftrightarrow F_2)$ . И наоборот, если формулы эквивалентны  $(F_1 \leftrightarrow F_2)$ , то они равносильны ( $F_1 \equiv F_2$ ).

Таблица 1.5

Имя закона	Равносильные формулы $F_i \equiv F_j$
Коммутативности	$(F_1 \vee F_2) \equiv (F_2 \vee F_1)$ $(F_1 \& F_2) \equiv$
Ассоциативности	$F_1 \vee (F_2 \vee F_3) \equiv (F_1 \vee F_2) \vee F_3,$ $F_1 \& (F_2 \& F_3) \equiv (F_1 \& F_2) \& F_3$
Дистрибутивности	$F_1 \vee (F_2 \& F_3) \equiv (F_1 \vee F_2) \& (F_1 \vee F_3),$ $F_1 \& (F_2 \vee F_3) \equiv (F_1 \& F_2) \vee (F_1 \& F_3)$
Идемпотентности	$F \vee F \equiv F$ $F \& F \equiv F$
Исключенного	$F \vee \neg F \equiv \text{и}$
Противоречия	$F \& \neg F \equiv \text{л}$
Де Моргана	$\neg(F_1 \vee F_2) \equiv \neg F_1 \& \neg F_2$ и $\neg(F_1 \& F_2) \equiv$
Поглощения	$F_1 \vee (F_1 \& F_2) \equiv F_1$ и $F_1 \& (F_1 \vee F_2) \equiv F_1$
Поречского	$F_1 \vee (\neg F_1 \& F_2) \equiv (F_1 \vee F_2)$ и $F_1 \& (\neg F_1 \vee F_2) \equiv$

Обобщенного склеивания	$F_1 \& F_2 \vee \neg F_1 \& F_2 \equiv F_1 \& F_2 \vee \neg F_1 \& F_2 \vee F_2,$ $(F_1 \vee F_2) \& (\neg F_1 \vee F_2) \equiv$ $(F_1 \vee F_2) \& (\neg F_1 \vee F_2) \& F_2.$
Дополнения	$\neg(\neg F) \equiv F$
Константы	$F \vee \perp \equiv F$ $F \vee \text{и} \equiv \text{и}$ $F \& \perp \equiv \perp$

Справедливость некоторых законов следует проверить таблицами истинности.

Пример 1.19. Даны две формы закона поглощения.

$F_1 \vee (F_1 \& F_2) = F_1$				$F_1 \& (F_1 \vee F_2) = F_1$				Сравните значения логических функций в первом и четвертом столб-
F <sub>1</sub>	F <sub>2</sub>	1 &	1 ∨ 2	F <sub>1</sub>	F <sub>2</sub>	1 ∨ 2	1 &	
1	2	3	4	1	2	3	4	
п	п	п	п	п	п	п	п	
п	и	п	п	п	и	и	п	
и	п	п	и	и	п	и	и	
и	и	и	и	и	и	и	и	

Пример 1.20. Даны две формы закона де Моргана.

$\neg(F_1 \vee F_2) \equiv \neg F_1 \& \neg F_2$				$\neg(F_1 \& F_2) \equiv \neg F_1 \vee \neg F_2$				Сравните значения логических функций в третьем и четвертом столб-
F <sub>1</sub>	F <sub>2</sub>	1 ∨ 2	1 &	F <sub>1</sub>	F <sub>2</sub>	1 & 2	1 ∨ 2	
1	2	3	4	1	2	3	4	
п	п	и	и	п	п	и	и	
п	и	п	п	п	и	и	и	
и	п	п	п	и	п	и	и	
и	и	и	и	и	и	и	и	

Пример 1.21. Даны две формы закона дистрибутивности.

$F_1 \vee (F_2 \& F_3) = (F_1 \vee F_2) \& (F_1 \vee F_3)$								$F_1 \& (F_2 \vee F_3) = (F_1 \& F_2) \vee (F_1 \& F_3)$							
F <sub>1</sub>	F <sub>2</sub>	F <sub>3</sub>	2 &	1 ∨ 2	1 ∨ 3	1 ∨ 2 & 3	6 &	F <sub>1</sub>	F <sub>2</sub>	F <sub>3</sub>	2 ∨ 3	1 & 2	1 & 3	1 & 2 & 3	6 ∨ 7
1	2	3	4	5	6	7	8	1	2	3	4	5	6	7	8

п	п	п	п	п	п	п	п	п	п	п	п	п	п
п	п	и	п	п	п	и	п	п	п	и	п	п	п
п	и	п	п	п	и	п	п	п	п	п	п	п	п
п	и	и	и	и	и	и	и	и	и	и	и	и	и
и	п	п	п	и	и	и	и	и	и	п	п	п	п
и	и	п	п	и	и	и	и	и	и	п	и	и	и
и	и	и	и	и	и	и	и	и	и	и	и	и	и

Сравните значения логических функций в пятом и восьмом таблицы.

**Пример 1.22.** Даны две формы закона Порецкого

$F_1 \vee (\neg F_1 \& F_2) =$						$F_1 \& (\neg F_1 \vee F_2)$									
F <sub>1</sub>	F <sub>2</sub>	1	2	3	4	5	6	F <sub>1</sub>	F <sub>2</sub>	1	2	3	4	5	6
п	п	и	п	п	п	п	п	п	п	и	и	п	п	п	п
п	и	и	и	и	и	и	и	п	и	и	и	п	п	п	п
и	п	п	п	и	и	и	и	и	п	п	п	п	п	п	п
и	и	п	п	и	и	и	и	и	и	п	и	и	и	и	и

Сравните значения логических функций в пятом и

**Пример 1.23.** Даны две формы обобщенного склеивания.

$F_1 \& F_2 \vee \neg F_1 \& F_2 \equiv$							$(F_1 \vee F_2) \& (\neg F_1 \vee F_2) \equiv$										
$\equiv F_1 \& F_2 \vee \neg F_1 \& F_2 \vee F_2$							$\equiv (F_1 \vee F_2) \& (\neg F_1 \vee F_2) \& F_2$										
F <sub>1</sub>	F <sub>2</sub>	1	2	3	4	5	6	7	F <sub>1</sub>	F <sub>2</sub>	1	2	3	4	5	6	7

Сравните значения логических функций в шестом и седьмом столбцах каждой таблицы

Если формула F содержит подформулу F<sub>1</sub>, которая, в

свою

очередь, имеет эквивалентную формулу  $F_j$ , т.е.  $(F_i \leftrightarrow F_j)$ , то возможна подстановка в формулу  $F$  всюду вместо формулы  $F_i$  формулы  $F_j$ .

Этот факт описывают так:  $\int_{F_i}^{F_j} F(F_i)$

где символ « $\int_{F_i}^{F_j} F$ » означает подстановку формулы  $F_j$  вместо  $F_i$ .

Пример 1.18. Даны формулы  $F=(A \& C \rightarrow A)$  и  $A \equiv (C \rightarrow \neg B)$ . Выполнить подстановку.

$$\frac{\int_A^{C \rightarrow \neg B} (A \& (C \rightarrow A))}{(C \rightarrow \neg B) \& (C \rightarrow (C \rightarrow \neg B)) = (C \rightarrow \neg B)}$$

Любая замена переменной или подформулы в формуле определяется **правилом подстановки**, а любые эквивалентные преобразования формул **законами алгебры высказываний**, которые приведены в таблице 1.5.

#### 1.1.1.4. Эквивалентные преобразования формул

Знание законов алгебры высказываний позволяет выполнять эквивалентные преобразования любой правильно построенной формулы для замены одних логических операций на другие, сохраняя ее значение для всех наборов пропозициональных переменных.

Пример 1.24. Дано  $(F_1 \rightarrow F_2) \equiv (\neg F_1 \vee F_2) \equiv \neg(F_1 \& \neg F_2)$ .

Часто необходимо удалять символ « $\rightarrow$ » для последующих преобразований формул.

$F_1$	$F_2$	$\neg F_1$	$\neg F_2$	$1 \rightarrow 2$	$3 \vee 2$	$1 \& 4$	$\neg 7$
1	2	3	4	5	6	7	8
л	л	и	и	и	и	л	и
л	и	и	л	и	и	л	и
и	л	л	и	л	л	и	л
и	и	л	л	и	и	л	и

Равносильность или эквивалентность этих формул показана таблицей истинности. Сравните логические функции в пятом, шестом и восьмом столбцах.

Пример 1.25. Дано

$$F_1 \leftrightarrow F_2 \equiv (F_1 \rightarrow F_2) \& (F_2 \rightarrow F_1) \equiv (\neg F_1 \vee F_2) \& (\neg F_2 \vee F_1) \equiv \neg(\neg(\neg F_1 \vee F_2) \vee \neg(\neg F_2 \vee F_1)) \equiv F_1 \& F_2 \vee \neg F_1 \& \neg F_2.$$

Эквивалентность или равносильность формул показана в таблице истинности.

$F$	$F$	$F_1 \leftrightarrow F_2$	$F_1 \rightarrow F_2$	$F_2 \rightarrow F_1$	$4 \& 5$	$\neg F_1 \vee \neg F_2$	$\neg F_2 \vee F_1$	$7 \& 8$	$\neg 7 \vee \neg 8$	$\neg 1$
1	2	3	4	5	6	7	8	9	10	11
п	п	и	и	и	и	и	и	и	п	и
л	и	п	и	л	л	и	л	п	и	п
и	л	п	л	и	л	л	и	п	и	п
и	и	и	и	и	и	и	и	и	л	и

Сравните логические функции в третьем, шестом, девятом и одиннадцатом столбцах. Это показывает, что исполнение операции эквивалентности всегда можно заместить исполнением операциями импликации и конъюнкции или дизъюнкции и конъюнкции или дизъюнкции и отрицания.

Пример 1.26.

Дано

$$F = (F_1 \rightarrow F_2) \rightarrow ((F_2 \rightarrow F_3) \rightarrow (F_1 \vee F_2 \rightarrow F_3)). \text{ Упростить выражение}$$

ние формулы.

- Удалить всюду логическую связку « $\rightarrow$ »:

$$F = \neg(\neg F_1 \vee F_2) \vee (\neg(\neg F_2 \vee F_3) \vee (\neg(F_1 \vee F_2) \vee F_3)),$$

- опустить отрицание до элементарных формул по закону де Моргана:

$$F = F_1 \& \neg F_2 \vee F_2 \& \neg F_3 \vee \neg F_1 \& \neg F_2 \vee F_3,$$

- выполнить преобразование по закону дистрибутивности:

$$F = (F_1 \vee \neg F_1) \& \neg F_2 \vee F_2 \& \neg F_3 \vee F_3,$$

- удалить член  $(F_1 \vee \neg F_1)$ , так как  $(F_1 \vee \neg F_1) = \text{и}$ :

$$F = \neg F_2 \vee F_2 \& \neg F_3 \vee F_3,$$

- выполнить преобразование по закону Порецкого:

$$F = \neg F_2 \vee \neg F_3 \vee F_3,$$

- так как  $(F_3 \vee \neg F_3) = \text{и}$ , то

$$F = \neg F_2 \vee \text{и} = \text{и},$$

Пример \_\_\_\_\_ 1.27.

Дано

$F = \neg(F_1 \rightarrow F_2) \& (\neg F_3 \vee \neg F_4) \vee \neg(F_1 \vee F_2) \& \neg(F_3 \& F_4)$ . Упростить выражение формулы.

- Удалить логическую связку « $\rightarrow$ »:

$$F = \neg(\neg F_1 \vee F_2) \& (\neg F_3 \vee \neg F_4) \vee \neg(F_1 \vee F_2) \& \neg(F_3 \& F_4),$$

- опустить отрицание на элементарные формулы по закону де Моргана:

$$F = F_1 \& \neg F_2 \& (\neg F_3 \vee \neg F_4) \vee \neg F_1 \& \neg F_2 \& (\neg F_3 \vee \neg F_4),$$

- выполнить преобразование по закону дистрибутивности:

$$F = (F_1 \vee \neg F_1) \& \neg F_2 \& (\neg F_3 \vee \neg F_4),$$

- удалить член  $(F_1 \vee \neg F_1) = \text{и}$ :

$$F = \neg F_2 \& (\neg F_3 \vee \neg F_4).$$

$$\text{Итак, } \neg(F_1 \rightarrow F_2) \& (\neg F_3 \vee \neg F_4) \vee \neg(F_1 \vee F_2) \& \neg(F_3 \& F_4) \equiv \neg F_2 \& (\neg F_3 \vee \neg F_4).$$

Пример 1.28. Дано суждение «или верно, что Петр поступил в университет (А), и при этом неверно, что Петр не поступил и Андрей не поступил, или Петр поступил и Семен поступил (С), или даже Петр поступил и Семен поступил, и Андрей поступил (В)». Кто же один из друзей поступил в университет? [15].

Формула этого высказывания имеет вид:

$$F = A \& \neg(\neg A \& \neg B) \vee A \& C \vee A \& B \& C.$$

Для ответа на поставленный вопрос необходимо:

- преобразовать формулу, используя закон де Моргана:

$$F = A \& (A \vee B) \vee A \& C \vee A \& B \& C,$$

- применить закон идемпотентности:

$$F = A \& (A \vee B) \vee A \& A \& C \vee A \& B \& C,$$

- применить закон дистрибутивности:

$$F = A \& ((A \vee B) \vee A \& C \vee B \& C),$$

$$F = A \& ((A \vee B) \vee C \& (A \vee B)),$$

$$F = (A \vee B) \& (A \vee C),$$

$$F = A \vee (B \& C).$$

Полученная формула равносильна исходной, но имеющая более простое выражение. Если в университет поступил только один из трех друзей (по условию задачи), то  $A = \text{и}$ , а  $(B \& C) = \text{л}$ . Следовательно, в университет поступил Петр.

Пример 1.29. Шесть школьников - Андрей, Борис, Григорий, Дмитрий, Евгений и Семен - участвовали в

олимпиаде. Двое из них решили все задачи. На вопрос, кто решил все задачи, последовали ответы: 1) Андрей и Дмитрий, 2) Борис и Евгений, 3) Евгений и Андрей, 4) Борис и Григорий, 5) Семен и Андрей. В четырех из этих ответов одна часть неверна, другая верна. В одном - обе части неверны. Кто же решил все задачи? [15].

Введем обозначения: А:= Андрей решил все задачи, Б:= Борис решил все задачи, Г:=Григорий решил все задачи, Д:= Дмитрий решил все задачи, Е:= Евгений решил все задачи, С:= Семен решил все задачи.

Так как в одном из ответов обе части неверны, а в остальных - одна, то необходимо составить пять формул, отражающих пять различных вариантов высказываний:

1)

$$\neg A \& \neg D \& (\neg B \& E \vee B \& \neg E) \& (\neg E \& A \vee E \& \neg A) \& (\neg B \& \Gamma \vee B \& \neg \Gamma) \& (\neg C \& A \vee C \& \neg A),$$

2)

$$\neg B \& \neg E \& (\neg A \& D \vee A \& \neg D) \& (\neg E \& A \vee E \& \neg A) \& (\neg B \& \Gamma \vee B \& \neg \Gamma) \& (\neg C \& A \vee C \& \neg A),$$

3)

$$E \& \neg A \& (\neg A \& D \vee A \& \neg D) \& (\neg B \& E \vee B \& \neg E) \& (\neg B \& \Gamma \vee B \& \neg \Gamma) \& (\neg C \& A \vee C \& \neg A),$$

4)

$$B \& \neg \Gamma \& (\neg A \& D \vee A \& \neg D) \& (\neg B \& E \vee B \& \neg E) \& (\neg E \& A \vee E \& \neg A) \& (\neg C \& A \vee C \& \neg A),$$

5)

$$\neg C \& \neg A \& (\neg A \& D \vee A \& \neg D) \& (\neg B \& E \vee B \& \neg E) \& (\neg E \& A \vee E \& \neg A) \& (\neg B \& \Gamma \vee B \& \neg \Gamma).$$

Если допустить, что  $\neg A = \text{и}$  и  $\neg D = \text{и}$ , то первая формула равна

$\neg A \& \neg D \& (\neg B \& E \vee B \& \neg E) \& E \& \neg A \& (\neg B \& \Gamma \vee B \& \neg \Gamma) \& C \& \neg A$ , так как  $\neg E \& A = \text{л}$ .

Если допустить, что  $\neg B = \text{и}$  и  $\neg E = \text{и}$ , то вторая формула равна  $\neg B \& \neg$

$E \& (\neg A \& D \vee A \& \neg D) \& \neg E \& A \& \neg B \& \Gamma \& (\neg C \& A \vee C \& \neg A)$ , так как  $E \& \neg A = \text{л}$  и  $B \& \neg \Gamma = \text{л}$ .

Если допустить, что  $\neg E = \text{и}$  и  $\neg A = \text{и}$ , то третья формула равна  $\neg E \& \neg A \& \neg A \& D \& B \& \neg E \& (\neg B \& \Gamma \vee B \& \neg \Gamma) \& C \& \neg A$ , так как  $A \& \neg D = \text{л}$ ,  $\neg B \& E = \text{л}$ , и  $\neg C \& A = \text{л}$ .

Если допустить, что  $\neg B = \text{и}$  и  $\neg \Gamma = \text{и}$ , то четвертая формула равна  $\neg B \& \neg \Gamma \& (\neg A \& D \vee A \& \neg D) \& \neg B \& E \& (\neg E \& A \vee E \& \neg A) \& (\neg C \& A \vee C \& \neg A)$ , так как  $B \& \neg E = \text{л}$ .

И наконец, если допустить, что  $\neg C = \text{и}$  и  $\neg A = \text{и}$ , то пятая формула равна  $\neg C \& \neg A \& \neg A \& D \& (\neg B \& E \vee B \& \neg E) \& E \& \neg A \& (\neg B \& \Gamma \vee B \& \neg \Gamma)$ , так как  $A \& \neg D = \text{л}$ .

То есть система формул имеет вид:

- 1)  $\neg A \& \neg D \& (\neg B \& E \vee B \& \neg E) \& E \& \neg A \& (\neg B \& \Gamma \vee B \& \neg \Gamma) \& C \& \neg A$ ,
- 2)  $\neg B \& \neg E \& (\neg A \& D \vee A \& \neg D) \& \neg E \& A \& \neg B \& \Gamma \& (\neg C \& A \vee C \& \neg A)$ ,
- 3)  $\neg E \& \neg A \& \neg A \& D \& B \& \neg E \& (\neg B \& \Gamma \vee B \& \neg \Gamma) \& C \& \neg A$ ,
- 4)  $\neg B \& \neg \Gamma \& (\neg A \& D \vee A \& \neg D) \& \neg B \& E \& (\neg E \& A \vee E \& \neg A) \& (\neg C \& A \vee C \& \neg A)$ ,
- 5)  $\neg C \& \neg A \& \neg A \& D \& (\neg B \& E \vee B \& \neg E) \& E \& \neg A \& (\neg B \& \Gamma \vee B \& \neg \Gamma)$ .

Применив законы дистрибутивности, идемпотентности и поглощения, эти формулы можно упростить так:

- 1)  $A \& \neg D \& \neg B \& E \& G \& C$ ,
- 2)  $\neg B \& \neg E \& \neg D \& \neg C \& A \& G$ ,
- 3)  $\neg E \& \neg A \& \neg G \& D \& C \& B$ ,
- 4)  $\neg B \& \neg G \& \neg A \& D \& E \& C$ ,
- 5)  $\neg C \& \neg A \& \neg B \& D \& E \& G$ .

По условиям задачи только два участника решили все задачи. Поэтому формулы, содержащие три пропозициональные переменные без отрицания ложны, а одна, содержащая только две пропозициональные переменные без отрицания, отвечает поставленным условиям. Это  $\neg B \& \neg E \& \neg D \& \neg C \& A \& G$ . Следовательно, все задачи на олимпиаде решили Андрей (А) и Григорий (Г).

Примеры показывают, что всякую формулу алгебры логики можно заместить равносильной формулой, содержащей только две логических связки: дизъюнкцию и отрицание или конъюнкцию и отрицание или три логических связки: дизъюнкции, конъюнкции и отрицания. Этот факт показывает, что эти логические связки представляют функционально полные алгебраические системы.

#### 1.1.1.5. Нормальные формы формул

В алгебре высказываний используют две нормальные формы: *дизъюнктивную* (ДНФ) и *конъюнктивную нормальные формы* формул (КНФ).

ДНФ есть формула, равносильная формуле исходной логической функции и записанная в виде дизъюнкции *элементарных конъюнкций*, построенных на пропозициональных переменных, т.е.  $F = K_1 \vee K_2 \vee K_3 \vee \dots$ , где каждая  $K_i = (A \& B \& C \& \dots)$ .

В элементарной конъюнкции нет двух одинаковых

пропозициональных переменных, так как  $A \& A \equiv A$ , а в ДНФ нет двух одинаковых элементарных конъюнкций, так как  $K \vee K \equiv K$ . Если одна из элементарных конъюнкций содержит  $(A \& \neg A)$ , то ее следует удалить из формулы ДНФ по закону противоречия.

Пример 1.30. Если  $F = F_1 \& (F_2 \vee \neg F_2) \vee F_2 \& (F_1 \vee \neg F_1)$ , то по закону

дистрибутивности

имеем

$$F = F_1 \& F_2 \vee F_1 \& \neg F_2 \vee F_1 \& F_2 \vee \neg F_1 \& F_2 =$$

$F_1 \& F_2 \vee F_1 \& \neg F_2 \vee \neg F_1 \& F_2$ . Так сформирована ДНФ.

КНФ есть формула, равносильная формуле исходной логической функции и записанная в виде конъюнкции *элементарных дизъюнкций*, построенных на пропозициональных переменных, т.е.  $F = D_1 \& D_2 \& D_3 \& \dots$ , где каждая  $D_i = (A \vee B \vee C \vee \dots)$ .

В элементарной дизъюнкции нет двух одинаковых пропозициональных переменных, так как  $A \vee A \equiv A$ , а в КНФ нет двух одинаковых элементарных дизъюнкций,  $D \& D \equiv D$ . Если одна из элементарных дизъюнкций содержит  $(F \vee \neg F)$ , то ее следует удалить из КНФ по закону исключенного третьего.

Пример 1.31. Если  $F = F_1 \& (F_1 \vee F_2) \vee \neg F_2 \& (F_1 \vee F_2)$ , то по закону дистрибутивности имеем  $F = (F_1 \vee F_2) \& (F_1 \vee \neg F_2)$ . Так сформирована КНФ.

Наибольшее распространение в логике высказываний получили формулы КНФ, элементарные дизъюнкции которых  $D_i$  принято называть *дизъюнктами*, а члены каждого дизъюнкта  $A, B, C$  — *атомами*.

**Алгоритм приведения формулы к нормальной**

**форме.**

Шаг 1. Устранить « $\leftrightarrow$ » и « $\rightarrow$ » всюду по правилам:

$$\begin{aligned}(F_1 \leftrightarrow F_2) &\equiv (F_1 \rightarrow F_2) \& (F_2 \rightarrow F_1) \equiv (\neg F_1 \vee F_2) \& (\neg F_2 \vee F_1) \equiv \\ &\equiv \neg(\neg(\neg F_1 \vee F_2) \vee \neg(\neg F_2 \vee F_1)) \equiv \\ &F_1 \& F_2 \vee \neg F_1 \& \neg F_2,\end{aligned}$$

$$(F_1 \rightarrow F_2) \equiv (\neg F_1 \vee F_2) \equiv \neg(F_1 \& \neg F_2).$$

Шаг 2. Продвинуть отрицание до пропозициональной переменной по правилам:

$$\neg(\neg F) \equiv F, \quad \neg(F_1 \vee F_2) \equiv (\neg F_1 \& \neg F_2), \quad \neg(F_1 \& F_2) \equiv (\neg F_1 \vee \neg F_2).$$

Шаг 3. Применить закон дистрибутивности:

а) для КНФ:  $F_1 \vee F_2 \& F_3 \equiv (F_1 \vee F_2) \& (F_1 \vee F_3),$

б) для ДНФ:  $F_1 \& (F_2 \vee F_3) \equiv F_1 \& F_2 \vee F_1 \& F_3.$

Пример 1.32.  $F = ((F_1 \rightarrow (F_2 \vee \neg F_3)) \rightarrow F_4).$  Преобразовать формулу к виду КНФ.

- Удалить всюду символ « $\rightarrow$ »:

$$F = \neg(\neg F_1 \vee F_2 \vee \neg F_3) \vee F_4,$$

- выполнить логическую операцию отрицания формулы:

$$F = F_1 \& \neg F_2 \& F_3 \vee F_4,$$

- применить закон дистрибутивности:

$$F = (F_1 \vee F_4) \& (\neg F_2 \vee F_4) \& (F_3 \vee F_4). \text{ Это – КНФ формулы.}$$

Пример 1.33.  $F = \neg(F_1 \& F_2) \& (F_1 \vee F_2).$  Преобразовать формулу к виду ДНФ.

- Выполнить операцию отрицания формулы:

$$F = (\neg F_1 \vee \neg F_2) \& (F_1 \vee F_2),$$

- применить закон дистрибутивности:

$$F = (F_1 \& \neg F_1 \vee F_1 \& \neg F_2 \vee F_2 \& \neg F_1 \vee F_2 \& \neg F_2),$$

- применить закон противоречия:

$$F = F_1 \& \neg F_2 \vee F_2 \& \neg F_1. \text{ Это – ДНФ формулы.}$$

Если каждая элементарная конъюнкция (или элемен-

тарная дизъюнкция) формулы содержат символы всех пропозициональных переменных, то такая формула называется совершенной. Есть *совершенные дизъюнктивные нормальные формы* формулы (СДНФ) и *совершенные конъюнктивные нормальные формы* формулы (СКНФ).

### Алгоритм преобразования ДНФ к виду СДНФ.

**Шаг 1:** если в элементарную конъюнкцию  $F$  не входит подформула  $F_i$  или  $\neg F_i$ , то дополнить элементарную конъюнкцию формулой  $(F_i \vee \neg F_i)$  и выполнить преобразование формулы по закону дистрибутивности  $F \& (F_i \vee \neg F_i) \equiv F \& F_i \vee F \& \neg F_i$ ,

**Шаг 2:** если в элементарную конъюнкцию  $F$  не входит подформула  $F_j$  или  $\neg F_j$ , то повторить шаг 1, иначе – конец.

Пример.

1.34

Дано

$$F = F_1 \& \neg F_2 \vee F_1 \& \neg F_3 \& F_4 \vee F_1 \& F_2 \& F_3 \& \neg F_4.$$

Преобразовать формулу к виду СДНФ:

- $F = F_1 \& \neg F_2 \& (F_3 \vee \neg F_3) \vee F_1 \& \neg F_3 \& F_4 \& (F_2 \vee \neg F_2) \vee F_1 \& F_2 \& F_3 \& \neg F_4;$
- $F = F_1 \& \neg F_2 \& F_3 \vee F_1 \& \neg F_2 \& \neg F_3 \vee F_1 \& F_2 \& \neg F_3 \& F_4 \vee F_1 \& \neg F_2 \& \neg F_3 \& F_4 \vee F_1 \& F_2 \& F_3 \& \neg F_4;$
- $F = F_1 \& \neg F_2 \& F_3 \& (F_4 \vee \neg F_4) \vee F_1 \& \neg F_2 \& \neg F_3 \& (F_4 \vee \neg F_4) \vee F_1 \& F_2 \& \neg F_3 \& F_4 \vee F_1 \& F_2 \& F_3 \& \neg F_4;$
- $F = (F_1 \& \neg F_2 \& F_3 \& F_4) \vee (F_1 \& \neg F_2 \& F_3 \& \neg F_4) \vee (F_1 \& \neg F_2 \& \neg F_3 \& F_4) \vee (F_1 \& \neg F_2 \& \neg F_3 \& \neg F_4) \vee (F_1 \& F_2 \& \neg F_3 \& F_4) \vee (F_1 \& F_2 \& F_3 \& \neg F_4);$

$(F_1 \& \neg F_2 \& \neg F_3 \& F_4) \vee$   
 $\vee (F_1 \& F_2 \& F_3 \& \neg F_4)$ . Так получена формула СДНФ.

**Алгоритм преобразования КНФ к виду СКНФ.**

**Шаг 1:** если в элементарную дизъюнкцию  $F$  не входит подформула  $F_i$  или  $\neg F_i$ , то дополнить элементарную дизъюнкцию высказыванием  $(F_i \& \neg F_i) = \text{л}$  и выполнить преобразование формулы по закону дистрибутивности  $F \vee (F_i \& \neg F_i) \equiv (F \vee F_i) \& (F \vee \neg F_i)$ ,

**Шаг 2:** если в элементарную конъюнкцию  $F$  не входит подформула  $F_j$  или  $\neg F_j$ , то повторить шаг 1, иначе – конец.

Пример 1.35. Дано  $F = (F_1 \vee F_2) \& (\neg F_1 \vee \neg F_2 \vee F_3 \vee F_4)$ .

Преобразовать формулу к виду СКНФ:

- $F = (F_1 \vee F_2 \vee F_3 \& \neg F_3) \& (\neg F_1 \vee \neg F_2 \vee F_3 \vee F_4)$ ;
- $F = (F_1 \vee F_2 \vee F_3) \& (F_1 \vee F_2 \vee \neg F_3) \& (\neg F_1 \vee \neg F_2 \vee F_3 \vee F_4)$ ;
- $F = (F_1 \vee F_2 \vee F_3 \vee F_4 \& \neg F_4) \& (F_1 \vee F_2 \vee \neg F_3 \vee F_4 \& \neg F_4) \&$   
 $\& (\neg F_1 \vee \neg F_2 \vee F_3 \vee F_4)$ ;

$F = (F_1 \vee F_2 \vee F_3 \vee F_4) \& (F_1 \vee F_2 \vee F_3 \vee \neg F_4) \& (F_1 \vee F_2 \vee \neg F_3 \vee F_4) \& (F_1 \vee F_2 \vee \neg F_3 \vee \neg F_4) \& (\neg F_1 \vee \neg F_2 \vee F_3 \vee F_4)$ . Так преобразования заданная формула к виду СКНФ.

Совершенные нормальные формы удобно записывать по таблицам истинности, используя значения пропозициональных переменных для всех значений, описываемых формулой.

Например, для формулы  $F = (A \rightarrow (B \vee C)) \& (B \rightarrow D) \& ((D \& A) \rightarrow \neg C)$  по таблице истинности 1.1 можно найти равносильные СДНФ  $F = (\neg A \& \neg B \& \neg C \& \neg D) \vee (\neg A \& \neg B \& \neg C \& D) \vee (\neg A \& \neg B \& C \&$

$$\begin{aligned} & \neg D) \vee (\neg A \& \neg B \& C \& D) \vee \\ & \vee (\neg A \& B \& \neg C \& D) \vee (\neg A \& B \& C \& D) \vee (A \& \neg B \& C \& \neg D) \vee (A \& \\ & B \& \neg C \& D) \text{ и СКНФ} \\ & F = (A \vee \neg B \vee C \vee D) \& (A \vee \neg B \vee \neg C \vee D) \& (\neg A \vee B \vee C \vee D) \& (\neg A \vee B \\ & \vee C \vee \neg D) \& (\neg A \vee B \vee \neg C \vee \\ & \vee \neg D) \& \\ & (\neg A \vee \neg B \vee C \vee D) \& (\neg A \vee \neg B \vee \neg C \vee D) \& (\neg A \vee \neg B \vee \neg C \vee \neg D). \end{aligned}$$

Элементарные конъюнкции формируются для значений формулы “и”. Число элементарных конъюнкций СДНФ равно числу истинных значений формулы. Пропозициональные переменные, элементарной конъюнкции записываются без изменений, если их значение равно “и” и с логической связкой “¬”, если их значение равно “л”.

Элементарные дизъюнкции СКНФ формируются для значения формулы, равной “л”. Число элементарных дизъюнкций равно числу ложных значений формулы. Пропозициональные переменные элементарной дизъюнкции, записываются без изменений, если их значение равно “л” и с логической связкой “¬”, если их значение равно “и”.

### 1.1.2. Исчисление высказываний

**Исчисление высказываний** – это формальная система, позволяющая формировать новые высказывания путем указания исходных высказываний, имеющих значение «истина», а также **аксиом** и **правил вывода**, каждое из которых описывает, как формировать новое высказывание из исходных и уже построенных высказываний. Поэтому изучение исчисления высказываний следует начинать с

определения аксиом и правил вывода, обеспечивающих доказательство истинности заключения.

**Аксиома\*** – это высказывание, имеющее значение истины при любых значениях пропозициональных переменных, входящих в это высказывание.

**Правилом вывода** называют такое отношение между высказываниями, которое позволяет из множества посылок и аксиом делать выводы об истинности заключения.

**Доказательством** или **выводом** называют такую линейно упорядоченную последовательность высказываний, каждое из которых является либо аксиомой, либо выводимо из одного или нескольких предыдущих высказываний.

Задание необходимого числа аксиом определяет **полноту исчисления**, а задание правил вывода – **метод исчисления**.

#### 1.1.2.1. Интерпретация формул

Определение истинного или ложного значений формулы высказывания для данного набора пропозициональных переменных называют **интерпретацией\*\***, а все множество формул – **полем интерпретации**.

Наиболее простой способ определения истинности или ложности формулы является использование таблиц истинности. Основным недостатком таблиц истинности состоит в том, что при большом числе пропозициональных переменных число строк таблицы равно  $2^n$ , а число столбцов не меньше  $(n+m)$ , где  $n$  – число пропозициональных переменных, а  $m$  – число логических операций.

---

\* ахіома (гр.) – отправное, исходное положение, лежащее в основе доказательств теорем.

\*\* interpretatio (лат.) – истолкование, разъяснение смысла чего-л.

Все множество формул можно разбить на три класса: тождественно истинные, тождественно ложные и выводимые.

**Тождественно истинные формулы** – это особый класс формул, которые принимают значение «истина» при любом значении пропозициональных переменных, входящих в формулу. Эти формулы играют роль аксиом и законов логики высказываний.

Например,  $F = ((A \rightarrow B) \& (A \rightarrow C) \rightarrow (A \rightarrow (B \& C))) = \text{и}$ .

Таблица 1.6

A	B	C	$A \rightarrow B$	$A \rightarrow C$	$B \& C$	$(A \rightarrow B) \& (A \rightarrow C)$	$(A \rightarrow (B \& C))$	
1	2	3	4	5	6	7	8	9
и	и	и	и	и	и	и	и	и
и	и	и	и	и	и	и	и	и
и	и	и	и	и	и	и	и	и
и	и	и	и	и	и	и	и	и
и	и	и	и	и	и	и	и	и
и	и	и	и	и	и	и	и	и
и	и	и	и	и	и	и	и	и
и	и	и	и	и	и	и	и	и

В девятом столбце таблицы 1.6 формула имеет значение истины для всех возможных

**Тождественно ложные формулы** - это особый класс формул, которые принимают значение «ложь» при любых значениях пропозициональных переменных, входящих в формулу.

Например,  $F = A \& (\neg B \vee \neg C) \& (A \rightarrow B) \& (A \rightarrow C) = \text{л}$ .

Таблица 1.7

A	B	C	$\neg B \vee \neg C$	$A \rightarrow B$	$A \rightarrow C$	$A \& (\neg B \vee \neg C)$	$(A \rightarrow B) \& (A \rightarrow C)$	
1	2	3	4	5	6	7	8	9
и	и	и	и	и	и	и	и	л
и	и	и	и	и	и	и	и	л
и	и	и	и	и	и	и	и	л
и	и	и	и	и	и	и	и	л
и	и	и	и	и	и	и	и	л
и	и	и	и	и	и	и	и	л
и	и	и	и	и	и	и	и	л
и	и	и	и	и	и	и	и	л
и	и	и	и	и	и	и	и	л

В девятом столбце таблицы 1.7 формула

и	п	и	и	и	п	п	п	п	п
и	и	п	и	и	и	и	п	и	п
и	и	и	п	п	и	и	п	п	п

**Выводимые формулы** - это класс формул, которые принимают значения истина или ложь в зависимости от значений пропозициональных переменных.

Например,  $F = (\neg A \rightarrow C \& \neg B) \& (\neg C \rightarrow B \& \neg A) \& (A \rightarrow \neg C)$ .

Таблица 1.8

A	B	C	3 & 4	1	2 & 3	1	5 & 7	9
1	2	3	4	5	6	7	8	9
п	п	п	п	п	п	п	и	п
п	п	и	и	и	п	и	и	и
п	и	п	п	п	и	и	и	п
п	и	и	п	п	и	и	и	п
и	п	п	п	и	п	п	и	п
и	п	и	и	и	п	и	п	п
и	и	п	п	и	п	п	и	п
и	и	и	п	и	п	и	п	п

В девятом столбце таблицы 1.8 формула имеет значение истины только для A=л, B=л и C=л.

1.1.2.2. Аксиомы исчисления высказываний

Среди множества тождественно истинных формул можно выделить подмножества, которые представляют аксиомы исчисления высказываний. В зависимости от числа логических связок могут быть найдены различные полные системы аксиом.

Например, известна полная система аксиом, опирающаяся на две логические связки « $\neg$ » и « $\rightarrow$ », которая содержит три аксиомы:

- $(F_1 \rightarrow (F_2 \rightarrow F_1))$ ,
- $(F_1 \rightarrow (F_2 \rightarrow F_3)) \rightarrow ((F_1 \rightarrow F_2) \rightarrow (F_1 \rightarrow F_3))$ ,
- $(\neg F_1 \rightarrow \neg F_2) \rightarrow ((\neg F_1 \rightarrow F_2) \rightarrow F_1)$ .

Возможны и другие полные системы аксиом, опирающиеся на другие сочетания логических связок. Так

система аксиом, опирающаяся на четыре логические связки  $\neg$ ,  $\&$ ,  $\vee$ ,  $\rightarrow$ , содержит двенадцать аксиом:

- A1.  $F_1 \rightarrow (F_2 \rightarrow F_1)$ ,
- A2.  $(F_1 \rightarrow F_2) \rightarrow ((F_2 \rightarrow F_3) \rightarrow (F_1 \rightarrow F_3))$ ,
- A3.  $(F_1 \& F_2) \rightarrow F_1$ ,
- A4.  $(F_1 \& F_2) \rightarrow F_2$ ,
- A5.  $F_1 \rightarrow (F_2 \rightarrow (F_1 \& F_2))$ ,
- A6.  $F_1 \rightarrow (F_1 \vee F_2)$ ,
- A7.  $F_2 \rightarrow (F_1 \vee F_2)$ ,
- A8.  $(F_1 \rightarrow F_3) \rightarrow ((F_2 \rightarrow F_3) \rightarrow ((F_1 \vee F_2) \rightarrow F_3))$ ,
- A9.  $(F_1 \rightarrow F_2) \rightarrow ((F_1 \rightarrow \neg F_2) \rightarrow \neg F_1)$ ,
- A10.  $(F_1 \rightarrow F_2) \rightarrow ((F_1 \& F_3) \rightarrow (F_2 \& F_3))$ ,
- A11.  $(F_1 \rightarrow F_2) \rightarrow ((F_1 \vee F_3) \rightarrow (F_2 \vee F_3))$ ,
- A12.  $\neg \neg F_1 \rightarrow F_1$ .

Приведенные полные системы аксиом равносильны между собой в том смысле, что порождают одно и то же множество формул. Их равносильность может быть доказана выводимостью аксиом первой системы из аксиом второй и наоборот.

Для проверки истинности аксиом достаточно рассмотреть аксиомы A2 и A8 (см. таблицы 1.9 и 1.10). В 9-м столбце каждой таблицы приведены значения этих формул.

Таблица  
1.9

$\neg$	$\vee$	$\&$	$\rightarrow$	$\rightarrow$	$\rightarrow$	$\rightarrow$	$\rightarrow$	$\rightarrow$
1	2	3	4	5	6	7	8	9

Таблица 1.10

$\neg$	$\vee$	$\&$	$\rightarrow$	$\rightarrow$	$\rightarrow$	$\rightarrow$	$\rightarrow$	$\rightarrow$
1	2	3	4	5	6	7	8	9

л	л	л	и	и	и	и	и	и	л	л	л	л	и	и	и	и	и
л	л	и	и	и	и	и	и	и	л	л	и	л	и	и	и	и	и
л	и	л	и	и	л	и	и	и	л	и	л	и	и	л	л	и	и
л	и	и	и	и	и	и	и	и	л	и	и	и	и	и	и	и	и
и	л	л	л	л	и	и	л	и	и	л	л	и	л	и	л	л	и
и	л	и	л	и	и	и	и	и	и	л	и	и	и	и	и	и	и
и	и	л	и	л	л	л	и	и	и	и	л	и	л	л	л	и	и
и	и	и	и	и	и	и	и	и	и	и	и	и	и	и	и	и	и

Для понимания механизма вывода заключения следует рассмотреть алгебраические правила введения и удаления логических связок. Эти правила – аналог аксиом исчисления высказываний - существенно облегчают преобразования сложных формул:

П1. Правило введения логической связки «&»: если формулы  $F_1$  и  $F_2$  истинны, то истинной является их конъюнкция, т. е.

$$\frac{F_1, F_2}{F_1 \& F_2} \text{ Это - аксиома A5.}$$

П2. Правило удаления логической связки «&»: если формула  $(F_1 \& F_2)$  истинна, то истинными являются формулы  $F_1$  и  $F_2$ , т. е.  $\frac{F_1 \& F_2}{F_1}$  и  $\frac{F_1 \& F_2}{F_2}$  Это - аксиомы A3 и A4.

П3. Правило введения логической связки « $\vee$ »: если хотя бы одна из формул  $F_1$  или  $F_2$  истинна, то истинной является их дизъюнкция, т. е.  $\frac{F_1}{F_1 \vee F_2}$  или  $\frac{F_2}{F_1 \vee F_2}$  Это - аксиомы

A6 и A7.

П4. Правило удаления логической связки « $\vee$ »: если формула  $(F_1 \vee F_2)$  истинна, то истинными могут быть фор-

мулы  $F_1$  или  $F_2$ , т. е.  $\frac{F_1 \vee F_2, \neg F_2}{F_1}$  или  $\frac{F_1 \vee F_2, \neg F_1}{F_2}$ .

П5. Правило введения логической связки « $\rightarrow$ »: если формула  $F_2$  истинна, то истинной является формула  $(F_1 \rightarrow F_2)$  при любом значении  $F_1$ , т. е.  $\frac{F_2}{F_1 \rightarrow F_2}$ . Это - аксиома

A1 («истина из чего угодно»).

П6. Правило *контрапозиции*: если формула  $(F_1 \rightarrow F_2)$  истинна, то истинной является формула  $(\Box F_2 \rightarrow \Box F_1)$ , т. е.  $\frac{F_1 \rightarrow F_2}{\neg F_2 \rightarrow \neg F_1}$ . Это - аксиома A9.

П7. Правило введения в формулу  $(F_1 \rightarrow F_2)$  логической связки « $\vee$ »: если формула  $(F_1 \rightarrow F_2)$  истинна, то истинной является формула  $((F_1 \vee F_3) \rightarrow (F_2 \vee F_3))$  при любом значении  $F_3$ , т. е.

$\frac{F_1 \rightarrow F_2}{(F_1 \vee F_3) \rightarrow (F_2 \vee F_3)}$  Это - аксиома A11.

П8. Правило введения в формулу  $(F_1 \rightarrow F_2)$  логической связки « $\&$ »: если формула  $(F_1 \rightarrow F_2)$  истинна, то истинной является формула  $((F_1 \& F_3) \rightarrow (F_2 \& F_3))$  при любом значении  $F_3$ , т. е.

$\frac{F_1 \rightarrow F_2}{(F_1 \& F_3) \rightarrow (F_2 \& F_3)}$  Это - аксиома A10.

П9. Правило *силлогизма*: если формулы  $(F_1 \rightarrow F_2)$  и  $(F_2 \rightarrow F_3)$  истинны, то истинной является формула  $(F_1 \rightarrow F_3)$ , т. е.

$\frac{(F_1 \rightarrow F_2), (F_2 \rightarrow F_3)}{(F_1 \rightarrow F_3)}$  Это - аксиома A2.

П10. Правило введения логической связки « $\leftrightarrow$ »: если формулы  $(F_1 \rightarrow F_2)$  и  $(F_2 \rightarrow F_1)$  истинны, то истинной является формула  $(F_1 \leftrightarrow F_2)$ ,

$$\text{т. е. } \frac{(F_1 \rightarrow F_2), (F_2 \rightarrow F_1)}{(F_1 \leftrightarrow F_2)}.$$

П11. Правило удаления логической связки « $\leftrightarrow$ »: если формула  $(F_1 \leftrightarrow F_2)$  истинна, то истинными являются формулы  $(F_1 \rightarrow F_2)$  и  $(F_2 \rightarrow F_1)$ , т.е.  $\frac{(F_1 \leftrightarrow F_2)}{(F_1 \rightarrow F_2)}$  или  $\frac{(F_1 \leftrightarrow F_2)}{(F_2 \rightarrow F_1)}$ .

П12. Правило объединения формул  $(F_1 \rightarrow F_3)$  и  $(F_2 \rightarrow F_3)$ : если формулы  $(F_1 \rightarrow F_3)$  и  $(F_2 \rightarrow F_3)$  истинны, то истинной является формула  $((F_1 \vee F_2) \rightarrow F_3)$ , т.е.  $\frac{(F_1 \rightarrow F_3), (F_2 \rightarrow F_3)}{(F_1 \vee F_2) \rightarrow F_3}$ .

Это - аксиома А8.

### 1.1.2.3. Метод дедуктивного вывода

**Выводом формулы**  $V$  из множества формул  $F_1, F_2, \dots, F_n$  называется такая последовательность формул, что любая  $F_i$  есть либо аксиома, либо выводима из множества предшествующих ей формул  $F_1, F_2, \dots, F_n$ .

В этом случае формулу  $V$  называют **заключением**, а последовательность формул  $F_1, F_2, \dots, F_n$ , - **схемой дедуктивного вывода**.

Схему дедуктивного вывода записывают так:

$$F_1, F_2, \dots, F_n \vdash V,$$

где « $\vdash$ » означает «верно, что  $V$  выводима из  $F_1, F_2, \dots, F_n$ ».

Известна другая форма записи этой схемы:

$$\frac{F_1, F_2, \dots, F_n}{V}$$

где над чертой записывают множество истинных посылок и аксиом  $F_1, F_2, \dots, F_n$ , а под чертой - заключение  $V$ .

На языке математики схема дедуктивного вывода\* есть доказательство теоремы (*теорема Эрбрана*):

$$\vdash F_1 \& F_2 \& \dots \& F_n \rightarrow B.$$

Истинность этой теоремы доказывается так: «если все  $F_i = \text{и}$ , то  $F_1 \& F_2 \& \dots \& F_n = \text{и}$ , а если  $B = \text{и}$ , то по таблице истинности импликации  $F_1 \& F_2 \& \dots \& F_n \rightarrow B = \text{и}$ ».

Используя правила эквивалентных преобразований теоремы, можно показать дедуктивный характер вывода заключения:

- $\vdash (F_1 \& F_2 \& \dots \& F_n \rightarrow B),$
- $\vdash (\neg(F_1 \& F_2 \& \dots \& F_n) \vee B),$
- $\vdash (\neg F_1 \vee \neg F_2 \vee \dots \vee \neg F_n \vee B),$
- $\vdash (\neg F_1 \vee \neg F_2 \vee \dots \vee \neg F_{n-1} \vee (F_n \rightarrow B)),$
- .....
- $\vdash (F_1 \rightarrow (F_2 \rightarrow \dots \rightarrow (F_{n-1} \rightarrow (F_n \rightarrow B)) \dots)).$

Таким образом, вывод заключения всегда логически следует из множества посылок и аксиом. Вывод заключения опирается на два основных правила:

а) если  $F_1$  и  $(F_1 \rightarrow F_2)$  выводимые формулы, то  $F_2$  также выводимая формула, т.е.  $\frac{F_1, F_1 \rightarrow F_2}{F_2}$ . Это правило называют

*modus ponens* (m. p.) или прямое доказательство.

б) если  $\neg F_2$  и  $(F_1 \rightarrow F_2)$  выводимые формулы, то  $\neg F_1$  также выводимая формула, т.е.  $\frac{\neg F_2, (F_1 \rightarrow F_2)}{\neg F_1}$ . Это правило на-

зывают *modus tollens* (m. t.) или доказательство «от противного».

Эти правила являются основными при выводе заключения, так как устанавливают отношение следования меж-

\* лат. deductio – выведение или логическое умозаключение.

ду двумя формулами.

Пример 1.36. «Всякое общественно опасное деяние (А) наказуемо (В). Преступление (С) есть общественно опасное деяние (А). Дача взятки (D) - преступление (С). Следовательно, дача взятки наказуема?» [9].

Формальная запись этого суждения:

$$\frac{(A \rightarrow B), (C \rightarrow A), (D \rightarrow C)}{(D \rightarrow B)}.$$

Дедуктивный вывод:

- $F_1 = A \rightarrow B$  – посылка,
- $F_2 = C \rightarrow A$  – посылка,
- $F_3 = D \rightarrow C$  – посылка,
- $F_4 = C \rightarrow B$  - заключение по формулам  $F_1$  и  $F_2$  и правилу П9,
- $F_5 = D \rightarrow B$  - заключение по формулам  $F_3$  и  $F_4$  и правилу П9, *ч.т.д.\**
- 

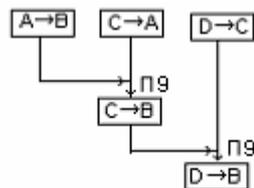


Рис. 1.1. Граф дедуктивного вывода ( $D \rightarrow B$ ).

Процесс дедуктивного вывода удобно демонстрировать с помощью графа (см. рис. 1.1), вершинами которого являются формулы, а дугами – отношения между формулами согласно аксиомам и правилам введения и удаления логических связок.

Пример 1.37. «Если Петров говорит неправду (А), то

\* - что и требовалось доказать.

он заблуждается (В) или сознательно вводит в заблуждение других (С). Петров говорит неправду и явно не заблуждается. Следовательно, он сознательно вводит в заблуждение других» [9].

Формальная запись этого суждения:

$$\frac{(A \rightarrow (B \vee C)), (A \& \neg B)}{C}$$

Дедуктивный вывод:

- $F_2 = A \& \neg B$  – посылка,
- $F_3 = A$  – заключение по формуле  $F_2$  и правилу П2,
- $F_4 = \neg B$  – заключение по формуле  $F_2$  и правилу П2,
- $F_5 = (B \vee C) \equiv (\neg B \rightarrow C)$  – заключение по  $F_1$ ,  $F_3$  и правилу т. р.;
- $F_6 = C$  – заключение по формулам  $F_4$ ,  $F_5$  и правилу т. р.,  
ч.т.д.

На рис. 1.2 показан граф вывода этой задачи.

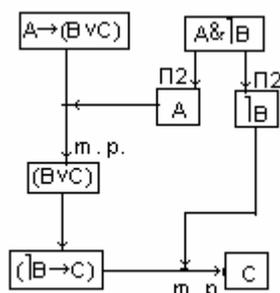


Рис. 1.2. Граф дедуктивного вывода С.

Пример 1.38. “Если Петров не трус (А), то он поступит в соответствие с собственными убеждениями (В). Если Петров честен (С), то он не трус (А). Если Петров не честен  $\neg(C)$ , то он не признает своей ошибки (D). Но Петров признает свои ошибки  $\neg(D)$ . Следовательно, он поступит со-

гласно собственным убеждениям (В)?"[9]

Формальная запись этого суждения:

$$\frac{(A \rightarrow B), (C \rightarrow A), (\neg C \rightarrow D), \neg D}{B}$$

Дедуктивный вывод:

- $F_1 = A \rightarrow B$  – посылка,
- $F_2 = C \rightarrow A$  – посылка,
- $F_3 = \neg C \rightarrow D$  – посылка,
- $F_4 = \neg D$  – посылка,
- $F_5 = C \rightarrow B$  – заключение по формулам  $F_1, F_2$  и правилу П9,
- $F_6 = \neg B \rightarrow \neg C$  – заключению по формуле  $F_5$  и правилу П6,
- $F_7 = \neg B \rightarrow D$  – заключение по формулам  $F_3$  и  $F_6$  и правилу П9,
- $F_8 = B$  – заключение по формулам  $F_4, F_7$  и правилу *m. t.*, *ч. т. д.*

На рис. 1.3 показан граф вывода этой задачи.

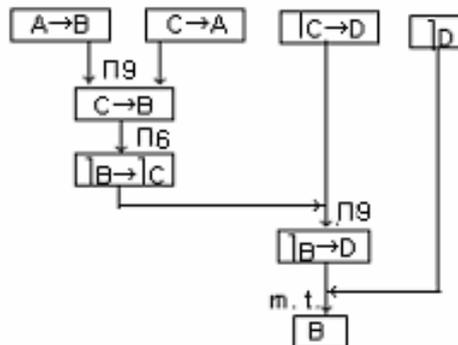


Рис. 1.3. Граф дедуктивного вывода В.

Пример 1.39. Доказать истинность заключения:

$$\frac{(A \vee B), (A \rightarrow C), (B \rightarrow D)}{(C \vee D)}.$$

Дедуктивный вывод:

- $F_1=(A \rightarrow C)$  - посылка,
- $F_2=(A \vee B) \rightarrow (C \vee B)$  - заключение по  $F_1$  и правилу П7,
- $F_3=(B \rightarrow D)$  - посылка,
- $F_4=(C \vee B) \rightarrow (C \vee D)$  - заключение по  $F_3$  и правилу П7,

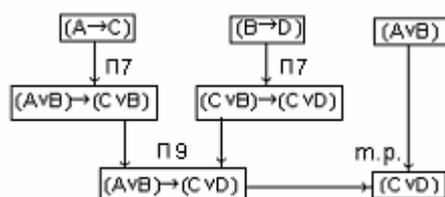


Рис. 1.4. Граф дедуктивного вывода  $(C \vee D)$ .

- $F_5=(A \vee B) \rightarrow (C \vee D)$  - заключение по  $F_2$  и  $F_4$  и правилу П9,
  - $F_6=(A \vee B)$  - посылка,
  - $F_7=(C \vee D)$  - заключение по  $F_5$  и  $F_6$  и правилу *m. p., ч. т. д.*
- На рис. 1.4 показан граф вывода этой задачи.

Пример 1.40. Доказать истинность заключения:

$$\frac{(A \rightarrow B) \& (C \rightarrow D), (D \& B \rightarrow E), \neg E}{(\neg C \vee \neg A)}.$$

Дедуктивный вывод:

- $F_1=(D \& B \rightarrow E)$  - посылка,
- $F_2=\neg E$  - посылка,
- $F_3=\neg(D \& B)$  - заключение по  $F_1$  и  $F_2$  и правилу *m. t.*,
- $F_4=(A \rightarrow B) \& (C \rightarrow D)$  - посылка,
- $F_5=(A \rightarrow B)$  - заключение по  $F_4$  и правилу П2,

- $F_6=(C \rightarrow D)$  - заключение по  $F_4$  и правилу П2,
- $F_7=(\neg B \rightarrow \neg A)$  - заключение по  $F_5$  и правилу П6,
- $F_8=(\neg D \vee \neg B)$  - заключение по  $F_3$  и закону де Моргана,
- $F_9=(D \rightarrow \neg B)$  - заключение по  $F_8$ ,
- $F_{10}=(D \rightarrow \neg A)$  - заключение по  $F_7$  и  $F_9$  и правилу П9,
- $F_{11}=(C \rightarrow \neg A)$  - заключение по  $F_6$  и  $F_{10}$  и правилу П9,
- $F_{12}=(\neg C \vee \neg A)$  - заключение по  $F_{11}$ , *ч.т.д.*

На рис. 1.5 показан граф вывода этой задачи.

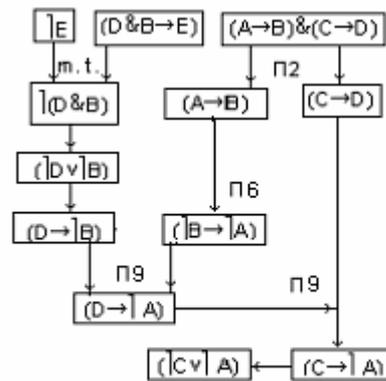


Рис.1.5. Граф дедуктивного вывода  $(C \vee A)$ .

Пример 1.41. Доказать истинность заключения:

$$\frac{((A \vee B) \rightarrow C), (C \rightarrow (D \vee M)), (M \rightarrow N), (\neg D \& \neg N)}{\neg A.}$$

Дедуктивный вывод:

- $F_1=(\neg D \& \neg N)$  - посылка,
- $F_2=\neg N$  - заключение по  $F_1$  и правилу П2,
- $F_3=(M \rightarrow N)$  - посылка,
- $F_4=\neg M$  - заключение по  $F_2$  и  $F_3$  и правилу m. t,
- $F_5=\neg D$  - заключение по  $F_1$  и правилу П2,

- $F_6 = (\neg D \& \neg M)$  - заключение по  $F_4$  и  $F_5$  и правилу П1,
- $F_7 = \neg(D \vee M)$  - заключение по  $F_6$  и закону де Моргана,
- $F_8 = ((A \vee B) \rightarrow C)$  - посылка,
- $F_9 = (C \rightarrow (D \vee M))$  - посылка,

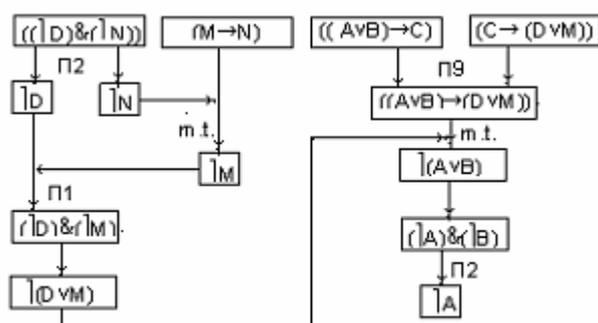


Рис.1.6. Граф дедуктивного вывода  $\neg A$ .

- $F_{10} = ((A \vee B) \rightarrow (D \vee M))$  - заключение по  $F_8$  и  $F_9$  и правилу П9,
- $F_{11} = \neg(A \vee B)$  - заключение по  $F_7$  и  $F_{10}$  и правилу m.t.,
- $F_{12} = (\neg A \& \neg B)$  - заключение по  $F_{11}$  и закону де Моргана,
- $F_{13} = \neg A$  - заключение по  $F_{12}$  и правилу П2, *ч.т.д.*

На рис. 1.6 показан граф вывода этой задачи.

Пример 1.42. “Если команда А выигрывает в футболе то город А’ торжествует, а если выигрывает команда В, то торжествовать будет город В’. Выигрывают или А или В. Однако, если выигрывают А, то город В’ не торжествует, а если выигрывают В, то не будет торжествовать город А’. Следовательно, город В’ будет торжествовать тогда и только тогда, когда не будет торжествовать город А”[3].

Формальная запись этого суждения:

$$\frac{(A \rightarrow A') \& (B \rightarrow B'), (A \vee B), (A \rightarrow \neg B') \& (B \rightarrow \neg A')}{B' \leftrightarrow \neg A'}$$

Дедуктивный вывод:

- $F_1 = (A \rightarrow A') \& (B \rightarrow B')$  – посылка,
- $F_2 = (A \rightarrow A')$  - заключение по формуле  $F_1$  и правилу П2,
- $F_3 = (B \rightarrow B')$  - заключение по формуле  $F_1$  и правилу П2,
- $F_4 = (A \rightarrow \neg B') \& (B \rightarrow \neg A')$  - посылка,
- $F_5 = (A \rightarrow \neg B')$  - заключение по формуле  $F_4$  и правилу П2,
- $F_6 = (B \rightarrow \neg A')$  - заключение по формуле  $F_4$  и правилу П2,
- $F_7 = (B' \rightarrow \neg A)$  - заключение по формуле  $F_5$  и правилу П6,
- $F_8 = (A' \rightarrow \neg B)$  - заключение по формуле  $F_6$  и правилу П6,
- $F_9 = (A \vee B)$  – посылка,
- $F_{10} = \neg A \rightarrow B$  - заключение по формуле  $F_9$ ,

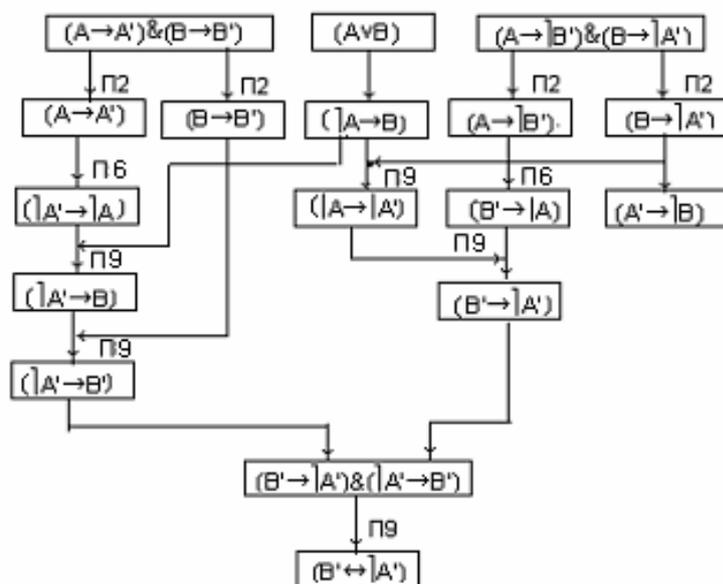


Рис. 1.7. Граф дедуктивного вывода  $(B' \leftrightarrow \neg A')$ .

- $F_{11} = \neg A \rightarrow \neg A'$  - заключение по формулам  $F_6$ ,  $F_{10}$  и правилу П9,
- $F_{12} = B' \rightarrow \neg A'$  - заключение по формулам  $F_7$ ,  $F_{11}$  и правилу П9,
- $F_{13} = \neg A' \rightarrow \neg A$  - заключение по формуле  $F_2$  и правилу П6,
- $F_{14} = \neg A' \rightarrow B$  - заключение по формулам  $F_{10}$ ,  $F_{13}$  и правилу П9,
- $F_{15} = \neg A' \rightarrow B'$  - заключение по формулам  $F_3$ ,  $F_{14}$  и правилу П9,
- $F_{16} = (B' \rightarrow \neg A') \& (\neg A' \rightarrow B') \equiv (B' \leftrightarrow \neg A')$  - заключение по формулам  $F_{12}$ ,  $F_{15}$  и правилу П1, *ч.т.д.*

На рис. 1.7 показан граф вывода этой задачи.

Примеры показывают, что правила обеспечивают последовательность в преобразовании формул, каждая из которых есть либо аксиома, либо посылка, либо промежуточный результат и, наконец, заключение.

#### 1.1.2.4. Метод резолюции

Существует эффективный алгоритм логического вывода - *принцип (или метод) резолюции* Дж. Робинсона. Он показал, что выводимость формулы  $B$  из множества посылок и аксиом  $F_1, F_2, F_3, \dots, F_n$  равносильна доказательству теоремы:

$$\begin{aligned} & \vdash (F_1 \& F_2 \& F_3 \& \dots \& F_n \rightarrow B), \\ & \vdash (\neg(F_1 \& F_2 \& F_3 \& \dots \& F_n) \vee B), \\ & \vdash \neg(F_1 \& F_2 \& F_3 \& \dots \& F_n \& \neg B). \end{aligned}$$

Из последней формулы следует, что заключение  $B$  истинно тогда и только тогда, когда формула  $(F_1 \& F_2 \& F_3 \& \dots \& F_n \& (\neg B)) = \text{л.}$  Так как  $(F_1 \& F_2 \& F_3 \& \dots \& F_n \& (\neg B))$  есть КНФ, то все  $F_i$  и  $\neg B$  должны быть приведены также к виду КНФ. Так может быть сформировано множество элементарных дизъюнктов  $K = \{D_1, D_2, \dots, D_m\}$ . Два дизъюнкта  $D_i$  и  $D_j$ , содержащие одинаковые пропозициональные переменные, но с противоположными знаками, объединяют в третий дизъюнкт  $D_k = (D_i \vee D_j)$  – *резольвенту*, из которой удаляются эти переменные. Пропозициональные переменные с противоположными знаками называют *контрарными атомами*. Если  $D_i = A$ ,  $D_j = \neg A$ , то  $D_k = (D_i \vee D_j) = (A \vee \neg A)$  есть пустая резольвента, которую обозначают символом  $\square$ . Многократно применяя процедуру объединения дизъюнктов множества  $K$  с контрарными атомами стремятся получить пустую ре-

зольвенту. Наличие пустой резольвенты есть решение  $\vdash \neg(F_1 \& F_2 \& F_3 \& \dots \& F_n \& \neg B)$ .

**Алгоритм вывода по методу резолюции:**

**Шаг 1:** принять отрицание заключения, т.е.  $\neg B$ ,

**Шаг 2:** привести все формулы посылок и отрицания заключения в конъюнктивную нормальную форму,

**Шаг 3:** выписать множество дизъюнктов всех посылок и отрицания заключения:  $K = \{D_1, D_2, \dots, D_k\}$ ,

**Шаг 4:** выполнить анализ пар множества  $K$  по правилу:

«если существуют дизъюнкты  $D_i$  и  $D_j$ , один из которых ( $D_i$ ) содержит атом  $A$ , а другой ( $D_j$ ) - атом  $\neg A$ , то соединить эту пару логической связкой дизъюнкции ( $D_i \vee D_j$ ) и сформировать новый дизъюнкт - резольвенту, исключив контрарные атомы  $A$  и  $\neg A$ , а резольвенту включить в множество  $K$ »,

**Шаг 5:** если в результате соединения дизъюнктов будет получена пустая резольвента (пустой дизъюнкт), то конец (доказательство подтвердило истинность заключения), иначе включить резольвенту в множество дизъюнктов  $K$  и перейти к шагу 4; по закону идемпотентности любой дизъюнкт и любую резольвенту можно использовать неоднократно, т.е. из множества  $K$  не следует удалять использованные в соединении дизъюнкты.

Пример 1.43. Доказать истинность заключения по методу резолюции:  $\frac{(A \& B \rightarrow C), (C \& D \rightarrow \neg M), (\neg N \rightarrow D \& M)}{(A \& B \rightarrow N)}$ .

Вывод по методу резолюции:

- $A \& B \rightarrow C \equiv \neg(A \& B) \vee C \equiv (\neg A \vee \neg B \vee C)$  – посылка содер-

жит один дизъюнкт,

- $C \& D \rightarrow \neg M \equiv \neg(C \& D) \vee \neg M \equiv (\neg C \vee \neg D \vee \neg M)$  – посылка содержит один дизъюнкт,

- $\neg N \rightarrow D \& M \equiv \neg \neg N \vee D \& M \equiv (N \vee D) \& (N \vee M)$  – посылка содержит два дизъюнкта,

- $\neg((A \& B) \rightarrow N) \equiv A \& B \& \neg N$  - отрицание заключения содержит три одно-литерных дизъюнкта,

- множество дизъюнктов:

$$K = \{(\neg A \vee \neg B \vee C), (\neg C \vee \neg D \vee \neg M), (N \vee D), (N \vee M), A, B, \neg N\},$$

- $(M \vee N) \vee \neg N \equiv M$  - резольвента,

- множество дизъюнктов:

$$K_1 = \{(\neg A \vee \neg B \vee C), (\neg C \vee \neg D \vee \neg M), (N \vee D), (N \vee M), A, B, M, \neg N\},$$

- $(\neg C \vee \neg D \vee \neg M) \vee M \equiv (\neg C \vee \neg D)$  - резольвента,

- множество дизъюнктов:

$$K_2 = \{(\neg A \vee \neg B \vee C), (\neg C \vee \neg D \vee \neg M), (N \vee D), (N \vee M), A, B, M, \neg N, (\neg C \vee \neg D)\},$$

- $(\neg A \vee \neg B \vee C) \vee (\neg C \vee \neg D) \equiv (\neg A \vee \neg B \vee \neg D)$  – резольвента,

- множество дизъюнктов:

$$K_3 = \{(\neg A \vee \neg B \vee C), (\neg C \vee \neg D \vee \neg M), (N \vee D), (N \vee M), A, B, M, \neg N, (\neg C \vee \neg D), (\neg A \vee \neg B \vee \neg D)\},$$

- $(\neg A \vee \neg B \vee \neg D) \vee A \equiv (\neg B \vee \neg D)$  - резольвента,

- множество дизъюнктов:

$$K_4 = \{(\neg A \vee \neg B \vee C), (\neg C \vee \neg D \vee \neg M), (N \vee D), (N \vee M), A, B, M, \neg N, D, (\neg C \vee \neg D), (\neg A \vee \neg B \vee \neg D)\}, (\neg B \vee \neg D) \},$$

- $(\neg B \vee \neg D) \vee B \equiv \neg D$  - резольвента,

- множество дизъюнктов:

$$K_5 = \{(\neg A \vee \neg B \vee C), (\neg C \vee \neg D \vee \neg M), (N \vee D), (N \vee M), A, B, M,$$

$\neg N, D, (\neg C \vee \neg D), (\neg A \vee \neg B \vee \neg D), (\neg B \vee \neg D), \neg D\}$ ,

•  $\neg D \vee (N \vee D) \equiv N$  - резольвента,

• множество дизъюнктов:

$K_6 = \{(\neg A \vee \neg B \vee C), (\neg C \vee \neg D \vee \neg M), (N \vee D), (N \vee M), A, B, M, \neg N, D, (\neg C \vee \neg D), (\neg A \vee \neg B \vee \neg D)\}, (\neg B \vee \neg D), \neg D, N\}$ ,

•  $N \vee \neg N \equiv \square$  - пустая резольвента, *ч.т.д.*

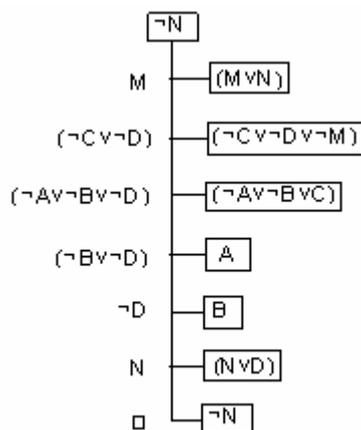


Рис. 1.8 Граф вывода  $(A \& B) \rightarrow N$  по принципу резолюции.

Для демонстрации удобно использовать граф типа дерева, корнем которого является один из дизъюнктов отрицания заключения, а листьями – дизъюнкты всех посылок и отрицания заключения. Тогда узлами графа являются резольвенты. На рис. 1.8 показан граф вывода этой задачи.

**Пример 1.44.** Доказать истинность заключения:

$$\frac{(A \rightarrow B) \& (C \rightarrow D), (D \& B \rightarrow M), \neg M}{(\neg A \vee \neg C)}.$$

Вывод по методу резолюции:

- $(A \rightarrow B) \& (C \rightarrow D) \equiv (\neg A \vee B) \& (\neg C \vee D)$  - посылка,
- $D \& B \rightarrow M \equiv \neg(D \& B) \vee M \equiv (\neg D \vee \neg B \vee M)$  - посылка,
- $\neg M$  - посылка,
- $\neg(\neg A \vee \neg C) \equiv A \& C$  - отрицание заключения,
- множество дизъюнктов:  
 $K = \{A, C, \neg M, (\neg A \vee B), (\neg C \vee D), (\neg D \vee \neg B \vee M)\}$ ,
- $A \vee (\neg A \vee B) \equiv B$  - резольвента,
- множество дизъюнктов:  
 $K = \{A, C, \neg M, (\neg A \vee B), (\neg C \vee D), (\neg D \vee \neg B \vee M), B\}$ ,
- $B \vee (\neg D \vee \neg B \vee M) \equiv (\neg D \vee M)$  - резольвента,
- множество дизъюнктов:  
 $K = \{A, C, \neg M, (\neg A \vee B), (\neg C \vee D), (\neg D \vee \neg B \vee M), B, (\neg D \vee M)\}$ ,
- $(\neg D \vee M) \vee (\neg C \vee D) \equiv (\neg C \vee M)$  - резольвента,
- множество дизъюнктов:  
 $K = \{A, C, \neg M, (\neg A \vee B), (\neg C \vee D), (\neg D \vee \neg B \vee M), B, (\neg D \vee M), (\neg C \vee M)\}$ ,
- $(\neg C \vee M) \vee \neg M \equiv \neg C$  - резольвента,
- $\neg C \vee C \equiv \square$  - пустая резольвента, *ч.т.д.*

На рис. 1.9 показан граф вывода этой задачи.

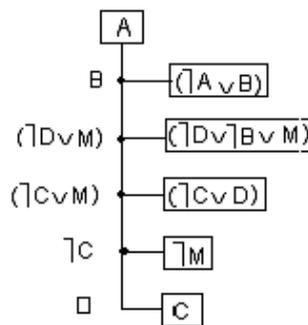


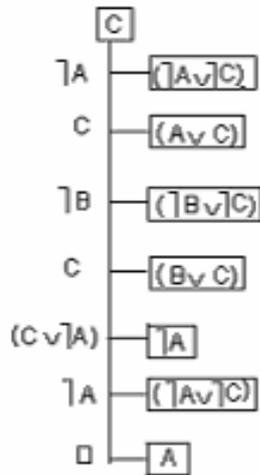
Рис. 1.9. Граф вывода  $(\neg A \vee C)$  по принципу резолюции.

Пример 1.45. Доказать истинность заключения:  

$$\frac{((\neg A \vee \neg B \vee \neg A \& \neg B) \rightarrow C), ((A \vee B \vee A \& B) \rightarrow \neg C)}{(C \rightarrow \neg A)}.$$

Вывод по методу резолюции:

- $((\neg A \vee \neg B \vee \neg A \& \neg B) \rightarrow C) \equiv (A \vee C) \& (B \vee C)$  - посылка,
- $(A \vee B \vee A \& B) \rightarrow \neg C \equiv (\neg A \vee \neg C) \& (\neg B \vee \neg C)$  - посылка,
- $\neg(C \rightarrow \neg A) \equiv C \& A$  - отрицание заключения,
- множество дизъюнктов:  
 $K = \{(A \vee C), (B \vee C), (\neg A \vee \neg C), (\neg B \vee \neg C), C, A\}$ ,
- $C \vee (\neg A \vee \neg C) = \neg A$  - резольвента,
- множество дизъюнктов:  
 $K_1 = \{(A \vee C), (B \vee C), (\neg A \vee \neg C), (\neg B \vee \neg C), C, A, \neg A\}$ ,
- $\neg A \vee (A \vee C) = C$  - резольвента,
- множество дизъюнктов:  
 $K_2 = \{(A \vee C), (B \vee C), (\neg A \vee \neg C), (\neg B \vee \neg C), C, A, \neg A\}$ ,
- $C \vee (\neg B \vee \neg C) = \neg B$  - резольвента,
- множество дизъюнктов:  
 $K_3 = \{(A \vee C), (B \vee C), (\neg A \vee \neg C), (\neg B \vee \neg C), C, A, \neg A, \neg B\}$ ,
- $\neg B \vee (B \vee C) = C$  - резольвента,
- множество дизъюнктов:  
 $K_4 = \{(A \vee C), (B \vee C), (\neg A \vee \neg C), (\neg B \vee \neg C), C, A, \neg A, \neg B\}$ ,
- $C \vee \neg A = (C \vee \neg A)$  - резольвента,
- множество дизъюнктов:  
 $K_5 = \{(A \vee C), (B \vee C), (\neg A \vee \neg C), (\neg B \vee \neg C), C, A, \neg A, \neg B, (C \vee \neg A)\}$ ,



**Рис. 1.10.** Граф вывода  $(\neg A \vee C)$  по принципу резолюции.

- $(C \vee \neg A) \vee (\neg A \vee \neg C) = \neg A$  – резольвента,
- множество дизъюнктов:  
 $K_5 = \{(A \vee C), (B \vee C), (\neg A \vee \neg C), (\neg B \vee \neg C), C, A, \neg A, \neg B, (C \vee \neg A)\}$ ,
- $(C \vee \neg A) \vee (\neg A \vee \neg C) = \neg A$  – резольвента,
- $\neg A \vee A = \square$  – пустая резольвента, *ч.т.д.*

На рис. 1.10 показан граф вывода этой задачи.

Обратите внимание, уже после формирования первой резольвенты  $(\neg A)$  из дизъюнктов второй посылки и отрицания заключения среди множества дизъюнктов появились два контрарных дизъюнкта  $(A$  и  $\neg A)$ , соединение которых формирует пустую резольвенту. Следовательно, первая посылка – излишняя в доказательстве истинности заключения. Это подтверждается истинностью дизъюнкта  $(\neg A \vee \neg C)$ , что равносильно истинности  $(C \rightarrow \neg A)$ . Однако, в настоящем выводе был продолжен поиск решения при наличии дизъюнктов первой посылки. Так как резольвен-

ты включаются в множество дизъюнктов, то возможно их многократное использование в процессе вывода. Это оправдано законом идемпотентности, т. е.  $D_i = D_i \& D_i \& \dots \& D_i$ . Достоинством принципа резолюции является то, что при доказательстве применяют только одно правило: поиск и удаление контрарных атомов дизъюнктов, а для этого механизм вывода использует правила подстановки и унификации дизъюнктов.

## Вопросы и задачи

## 1.1.1. Написать формулы суждений:

а) «подготовка специалистов высокой квалификации возможна лишь на базе всемерного развития вузовской науки, усиления связи вузовской, академической и отраслевой науки, обеспечения единства научной и учебной работы, широкого привлечения студентов к научным исследованиям»,

б) «хлеба уцелеют в различных климатических и погодных условиях тогда и только тогда, когда будут выполнены все мелиоративные работы, а если хлеба не уцелеют, то фермеры обанкротятся и оставят фермы» [21],

с) «если я поеду автобусом и автобус опоздает, то я опоздаю на работу, а если я опоздаю на работу, то я не сделаю в срок важную работу, а если я не сделаю в срок важную работу, то я стану огорчаться; следовательно, если я не поеду автобусом, то я сделаю в срок важную работу и не стану огорчаться» [3],

д) «обвиняемый может быть либо исполнителем, либо организатором совершенного преступления. Обвиняемый является организатором преступления. Следовательно, он не является исполнителем преступления»[9],

е) «если бы Цезарь был суверен, то он уступил бы просьбам Кальпурнии не идти в сенат. Если бы он был осторожен, он удалил бы Брута. Но Цезарь не уступил просьбам Кальпурии, не удалил Брута».

## 1.1.2. Доказать тождества:

а)  $(A \vee B) \& (A \vee \neg B) \equiv A$ ,

б)  $(A \vee B) \& (B \vee C) \& (C \vee A) \equiv (A \& B) \vee (B \& C) \vee (C \& A)$ ,

$$c) (A \vee B) \& (A \vee C) \& (B \vee D) \& (C \vee D) \equiv ((A \& D) \vee (B \& C)),$$

$$d) (A \vee B) \& (B \vee C) \& (C \vee A) \equiv ((A \& B) \vee (B \& C) \vee (C \& A)),$$

$$e) (A \vee B \vee C) \& (B \vee C \vee B) \& (C \vee D \vee A) \\ \equiv ((A \& B) \vee (A \& D) \vee (B \& D) \vee C),$$

$$f) (A \& B) \vee ((A \vee B) \& (\neg A \vee \neg B)) \equiv (A \vee B).$$

1.1.3. Привести формулу к виду ДНФ и КНФ:

$$a) (((A \rightarrow B) \rightarrow (C \rightarrow \neg A)) \rightarrow (\neg B \rightarrow \neg C)),$$

$$b) (((((A \rightarrow B) \rightarrow \neg A) \rightarrow \neg B) \rightarrow \neg C) \rightarrow C),$$

$$c) (A \rightarrow (B \rightarrow C)) \rightarrow (A \rightarrow \neg C) \rightarrow (A \rightarrow \neg B).$$

$$d) (\neg(A \& (B \vee C)) \rightarrow ((A \& B) \vee C)).$$

$$e) (C \rightarrow A) \rightarrow (\neg(B \vee C) \rightarrow A).$$

1.1.4. Выполнить подстановку:

$$\int_A^{B \& C} (A \vee B),$$

$$\int_{\neg B \rightarrow \neg A}^{B \vee C} \int_A^{\neg B \rightarrow \neg A} (A \rightarrow (B \vee C)),$$

$$c) \int_A^B (A \rightarrow B) \rightarrow (\neg B \rightarrow \neg A),$$

$$d) \int_C^B (A \vee B \vee C) \& (B \vee C \vee D) \& (A \vee C \vee D),$$

$$e) \int_{(C \rightarrow \neg A)}^{B \rightarrow C} (((A \rightarrow B) \rightarrow (C \rightarrow \neg A)) \rightarrow (\neg B \rightarrow \neg C))$$

1.1.5. Доказать истинность заключения по методу дедукции и принципу резолюции:

$$a) \frac{(A \vee B), (A \rightarrow C), (B \rightarrow D)}{(C \vee D)}.$$

$$b) \frac{((A \vee B) \rightarrow C), (C \rightarrow (D \vee E)), (E \rightarrow F), (\neg D \& \neg F)}{\neg A \& \neg C}.$$

$$c) \frac{(A \vee B), (A \rightarrow B), (B \rightarrow A)}{A \& B}.$$

$$\text{d) } \frac{((A \vee B) \rightarrow C \& D), ((D \vee B) \rightarrow F)}{(A \rightarrow F)}$$

$$\text{e) } \frac{(A \rightarrow B), (C \rightarrow D), (A \vee C), (A \rightarrow \neg D), (C \rightarrow \neg D)}{(D \leftrightarrow \neg B)}$$

### Расчетно-графическая работа

- составить таблицу истинности, число строк которой равно  $2^n$ , где  $n$  – число пропозициональных переменных, а число столбцов равно сумме числа пропозициональных переменных, посылок и заключения, а также столбец для конъюнкции всех посылок и столбец для импликации заключения из конъюнкции всех посылок; выделить штриховкой строки, в которых истинны все посылки и заключение; дать объяснения,
- доказать истинность заключения методом дедукции и нарисовать граф дедуктивного вывода,
- доказать истинность заключения методом резолюции и нарисовать граф вывода пустой резольвенты.

Вари-	Доказать истинность заключения
1	$(A \rightarrow B) (C \rightarrow D) (B \rightarrow \neg D) (A \& \neg D) \vdash \_$
2	$(A \rightarrow B) (C \rightarrow D) (B \rightarrow \neg D) C \vdash (\neg A \vee \neg D)$
3	$(A \rightarrow B) (C \rightarrow B) ((B \vee C) \rightarrow (C \vee D))$
4	$(A \rightarrow B) (C \rightarrow B) (B \rightarrow D) \neg D \vdash (\neg A \& \neg C)$
5	$(A \rightarrow B) (A \rightarrow (B \rightarrow C)) (B \rightarrow C) \neg C \vdash \_$
6	$(A \rightarrow B) (C \rightarrow \neg B) (\neg C \& D) \vdash A$
7	$(A \rightarrow B) (B \rightarrow A) (B \rightarrow (A \rightarrow C)) \neg C \vdash \_$
8	$(A \rightarrow B) (D \rightarrow C) \neg B \neg C \vdash \neg (A \vee D)$
9	$(A \rightarrow B) (C \rightarrow \neg B) (C \& \neg D) \vdash \neg (\neg A \rightarrow D)$
10	$(A \rightarrow B) (B \rightarrow C) (C \rightarrow D) A \& B \vdash B \& D$
11	$(A \rightarrow B) (B \rightarrow C) (A \rightarrow C) (A \vee B \vee D) \vdash \_$
12	$(A \rightarrow B) (D \rightarrow C) \neg (B \vee C) \vdash (\neg A \& \neg D)$
13	$(A \rightarrow B) (B \rightarrow C) (D \rightarrow C) (A \vee D) \vdash (C \vee D)$
14	$(A \rightarrow B) (B \& D \rightarrow C) (A \& D) \vdash C$
15	$(A \rightarrow B) (C \rightarrow B) (\neg B \vee D) \neg (D \vee C) \vdash \_$

16	$(A \rightarrow B) (A \rightarrow (B \rightarrow C)) (A \& D) \vdash C$
17	$(A \rightarrow B) (C \rightarrow B) (D \rightarrow (A \vee C)) D \vdash B$
18	$(A \rightarrow B) (C \rightarrow D) (B \rightarrow \neg D) (C \& \neg D) \vdash$
19	$(A \rightarrow B) (A \rightarrow (B \rightarrow C)) \neg(C \vee D) \vdash$
20	$(A \rightarrow B) (B \rightarrow C) (C \rightarrow D) A \& B \vdash B \& D$
21	$(A \rightarrow (B \rightarrow C)) (A \rightarrow B) \neg(C \vee D) \vdash$
22	$(A \rightarrow (B \rightarrow C)) (\neg D \vee A) B \vdash (\neg D \vee C)$
23	$(A \rightarrow (B \rightarrow C)) (\neg D \vee A) B D \vdash C$
24	$(A \rightarrow (B \rightarrow C)) (A \rightarrow B) A \& D \vdash C \& D$
25	$(A \rightarrow (B \rightarrow C)) (\neg D \vee A) B \vdash (D \vee C)$
26	$(A \rightarrow (B \rightarrow C)) (\neg D \vee A) B \vdash (D \rightarrow C)$
27	$(A \rightarrow (B \rightarrow C)) ((A \rightarrow C) \rightarrow D) \neg D \vdash \neg B$
28	$(B \rightarrow A) (B \rightarrow (\neg A \vee C)) (\neg C \vee \neg D) \vdash$
29	$(B \rightarrow A) (B \rightarrow (A \rightarrow C)) B \& D \vdash C$
30	$(B \rightarrow A) (B \rightarrow (A \rightarrow C)) \neg(C \vee D) \vdash$
31	$(B \rightarrow A) (B \rightarrow (\neg A \vee C)) (B \& D) \vdash C \& D$
32	$(B \rightarrow A) (C \rightarrow A) ((A \vee C) \rightarrow (C \vee D))$
33	$(B \rightarrow A) (A \rightarrow C) (D \rightarrow C) (B \vee D) \vdash (C \vee D)$
34	$(B \rightarrow (A \rightarrow C)) (B \rightarrow A) \neg C \& \neg D \vdash$
35	$(B \rightarrow (A \rightarrow C)) (B \rightarrow A) (C \rightarrow D) \neg D \vdash \neg B$
36	$(B \rightarrow (A \rightarrow C)) (B \rightarrow A) \neg C \vee \neg D \vdash$
37	$(B \rightarrow (A \rightarrow C)) (B \rightarrow A) \neg C \& \neg D \vdash$
38	$(\neg A \vee B) (C \vee \neg B) (\neg C \& \neg D) \vdash \neg(A \vee D)$
39	$(\neg A \vee \neg B) (C \rightarrow A) (B \vee \neg D) \vdash (\neg C \vee \neg D)$
40	$(\neg A \vee B) (C \vee \neg B) (A \vee D) \neg D \vdash C$
41	$(\neg A \vee \neg B) (C \rightarrow A) B \& D \vdash D \rightarrow C$
42	$(A \vee B) (A \rightarrow C) (B \rightarrow D) \neg C \vdash D$
43	$(\neg A \vee B) (C \rightarrow \neg B) \neg(C \rightarrow B) \vdash \neg(\neg A \rightarrow B)$
44	$(A \leftrightarrow B) (A \rightarrow (\neg B \vee C)) (D \vee B) \vdash (\neg C \rightarrow D)$
45	$(A \leftrightarrow B) (\neg B \rightarrow C) \rightarrow (\neg C \rightarrow D) (A \vee C) \vdash$
46	$(A \leftrightarrow B) (A \rightarrow (\neg B \vee C)) (D \vee B) \vdash (\neg C \rightarrow D)$
47	$(A \& B \rightarrow C) (\neg D \vee \neg C) D \vdash \neg A \vee \neg B$
48	$(A \& B \vee \neg A \& \neg B \vee C) \neg C \vdash (A \rightarrow B)$
49	$(A \vee (C \& D)) (D \vee A) B D \vdash C$

50	$(\neg A \vee (\neg C \vee B)) (\neg D \vee A) C D \vdash B$
----	--

## 1. 2. Логика предикатов

Логика предикатов – это расширение возможностей логики высказываний, позволяющее строить высказывания с учетом свойств объектов высказывания или отношений между ними.

Если неизвестны объекты высказывания или отношения между объектами высказывания, то такое высказывание называют **высказывательной функцией**. Высказывательную функцию иначе называют **предикатом**<sup>\*</sup> и обозначают символом  $P(\dots)$ . Аргументами предиката являются **предметные переменные** и **предметные постоянные**. Предметные переменные и предметные постоянные обозначают строчными буквами латинского алфавита  $\{x, y, z, \dots\}$  и  $\{a, b, c, \dots\}$  соответственно. Предикаты бывают одноместные  $P(\cdot)$ , когда его аргументом является одна предметная переменная, и многоместные  $P(\cdot, \cdot, \dots, \cdot)$ , когда его аргументами могут быть несколько предметных переменных и/или предметных постоянных. Одноместные предикаты выражают свойства предметных переменных, а многоместные – отношения между объектами высказывательной функции. Высказывательная функция приобретает значение высказывания только при замене предметных переменных предметными постоянными.

Множество, на котором даны предметные переменные и предметные постоянные, называют **областью опре-**

<sup>\*</sup> praedicatum - логическое сказуемое

**деления предиката** или **универсумом**. Множество, на котором предикат принимает значение «истины» называют **множеством истинности предиката**  $P(x)$  или  $P(x_1, x_2, \dots, x_n)$ .

Например,

а) если на множестве натуральных чисел задать функции:

$P_1(x) := \langle x \text{ - простое число} \rangle$ ,

$P_2(6, y) := \langle y \text{ меньше '6'} \rangle$ ,

$P_3(6, y, z) := \langle z \text{ есть частное от деления числа '6' на } y \rangle$ ,

где  $x, y, z$  есть предметные переменные, а '6' – предметная постоянная, то высказываниями будут

$P_1('3') = \text{и}, \quad P_1('4') = \text{л},$

$P_2('6', '2') = \text{и}, \quad P_2('6', '7') = \text{л},$

$P_3('6', '2', '3') = \text{и}, \quad P_3('6', '2', '5') = \text{л},$

б) если на множестве имен индивидов, учебных заведений и специальностей задать следующие предикаты:

$P_4(x) := \langle x \text{ - студент} \rangle$ ,

$P_5(\text{'Сидоров'}, \text{'КГТУ'}) = \text{л}$   $P_6(\text{'Петров'}, \text{'прикладная информатика'}) = \text{и}$ ,  $P_6(\text{'Сидоров'}, \text{'прикладная информатика'}) = \text{л}$ .

Студентов, учебных заведений и специальностей много. Для ограничения области определения предикатов на множестве предметных переменных вводят дополнительные логические операторы, которые называют **кванторами**.

Так, суждение, в котором утверждается или отрицается наличие каких-либо свойств или отношений у части предметных переменных, называют **частным суждением**. Для формального описания таких суждений используют

логический оператор « $\exists_x$ », который получил название **квантор существования**. Предикат или сложную высказывательную функцию записывают после квантора существования в круглых скобках  $\exists_x(P(x_1, x_2, \dots, x_i, \dots, x_n))$  или  $\exists_x(F(x_1, x_2, \dots, x_i, \dots, x_n))$ . На естественном языке эта запись означает «существуют такие предметные переменные  $x_i$ , для которых  $P(x_i)$  истинно». Иначе это выражают словами «один», «несколько» или «часть  $x_i$ » и т.п.

Пример 1.46. Если на множестве целых чисел  $N$  дана предметная переменная  $x \in N$  и дан предикат, определяющий свойство предметной переменной,  $P_1(x) :=$  «быть простым числом», то может быть записано частное суждение  $\exists_x(P_1(x)) :=$  «часть целых чисел являются простыми числами». Этот предикат на множестве  $N$  (область определения) выделяет подмножество простых чисел  $X = \{2, 3, 5, 7, 11, \dots, \}$ .

Пример 1.47. Если на множестве  $N$  даны предметные переменные  $x \in N$  и  $y \in N$  и дан предикат, устанавливающий отношение между ними,  $P_2(x, y) :=$  « $y$  меньше  $x$ », то при  $x = '6'$  может быть записано частное суждение  $\exists_y(P_2('6', y)) :=$  «есть целые числа, которые меньше '6'». Этот предикат на множестве  $N$  при заданной предметной постоянной выделяет подмножество  $Y = \{1, 2, 3, 4, 5\}$ .

Пример 1.48. Если на множестве  $N$  даны предметные переменные  $x, y, z \in N$  и предикат, устанавливающий отношение между ними:  $P_3(x, y, z) :=$  «существует такое  $z$ , которое является частным от деления  $x$  на  $y$ », то при  $x = '6'$  и  $y = '2'$  может быть записано частное суждение  $\exists_z(P_3('6', '2', z)) :=$  «существует такое  $z$ , которое является частным от де-

ления '6' на '2'». Этот предикат на множестве  $N$  выделяет единственное число  $z=3$ , для которого  $P_3('6', '2', '3')=и$ .

Пример 1.49. Если даны на множестве индивидов  $M$  предметная переменная  $x \in M$ , на множестве учебных заведений  $Y$  - предметная переменная  $y \in Y$ , на множестве специальностей  $Z$  - предметная переменная  $z \in Z$  и предикаты  $P_4(x) := \langle\langle x - \text{студент} \rangle\rangle$ ,  $P_5(x, y) := \langle\langle x \text{ обучается в университете } y \rangle\rangle$ ,  $P_6(x, z) := \langle\langle x \text{ обучается по специальности } z \rangle\rangle$  и  $P_7(x) := \langle\langle x \text{ имеет зачетную книжку} \rangle\rangle$ , то

$\exists_x(P_4(x)) := \langle\langle \text{существуют } x, \text{ которые являются студентами} \rangle\rangle$ ,

$\exists_y(P_5(x, y)) := \langle\langle \text{существуют } x, \text{ которые обучаются в университете} \rangle\rangle$ ,

$\exists_z(P_6(x, z)) := \langle\langle \text{существуют } x, \text{ которые обучаются по специальности } z \rangle\rangle$ ,

$\exists_x(P_4(x) \& \neg P_7(x)) := \langle\langle \text{существуют студенты, которые не имеют зачетной книжки} \rangle\rangle$ ,

$\exists_x \exists_y (P_4(x) \rightarrow P_5(x, y)) := \langle\langle \text{если } x - \text{студент, то он обучается в каком-то университете } y \rangle\rangle$ ,

$\exists_x \exists_y \exists_z (P_4(x) \rightarrow P_5(x, y) \& P_6(x, z)) := \langle\langle \text{если } x - \text{студент, то он обучается в каком-то университете } y \text{ по какой-то специальности} \rangle\rangle$

Частное суждение по нескольким переменным требует указания кванторов существования для каждой переменной перед предикатом. Например,  $\exists_x \exists_y \exists_z \dots (P_n(x, y, z, \dots))$ .

Пример 1.50. Частное суждение  $\exists_x \exists_y (P_4(x) \& \neg P_5(x, y) \& \neg P_7(x)) := \langle\langle \text{существуют студенты } (x) \text{ некоторых университетов } (y), \text{ которые не имеют зачетной книжки} \rangle\rangle$  и т.п. В данной формуле есть ограничение на множество уни-

верситетов и на множество студентов, которые не имеют зачетных книжек.

Суждение, в котором утверждается или отрицается наличие каких-либо свойств для всех предметных переменных области определения или отношений между ними, называют *общими суждениями*. Как правило, эти суждения в естественном языке отмечают словами «все», «каждый», «любой» и т.п. Для формального описания таких суждений используют логический оператор – « $\forall_x$ », который получил название *квантор всеобщности*. Предикат или сложную высказывательную функцию записывают после квантора всеобщности в круглых скобках  $\forall_x(P(x_1, x_2, \dots, x_n))$  или  $\forall_x(F(x_1, x_2, \dots, x_n))$ . На естественном языке эта формальная запись означает «для всех предметных переменных  $x_i$  значение  $P(x_i)$  истинно» или «для каждой предметной переменной  $x_i$  значение  $P_n(x_i)$  истинно»..

Если общее суждение распространяется на несколько предметных переменных, то перед предикатом или сложной высказывательной функцией записывают кванторы всеобщности для каждой предметной переменной, т.е.

$$\forall_x \forall_y \forall_z \dots (P(x, y, z, \dots, )).$$

Например,  $\forall_x(P_4(x) \& P_7(x)) :=$  «все студенты имеют зачетные книжки»,

$\forall_x \forall_y(P_4(x) \& P_5(x, y) \& P_7(x)) :=$  «все студенты всех университетов имеют зачетные книжки»,

$\forall_x(P_4(x) \& P_5(x, \text{'КГТУ'}) \& P_6(x, \text{'прикладная информатика'}) \& P_7(x)) :=$  «все студенты, обучающиеся в университете КГТУ по специальности «прикладная информатика», имеют зачетные книжки»,

$\forall_x \forall_z (P_4(x) \& P_5(x, \text{'КГТУ'}) \& P_6(x, z) \& P_7(x)) :=$  «все студенты университета КГТУ, обучающиеся на всех специальностях, имеют зачетные книжки»,

$\forall_x \forall_y \forall_z (P_4(x) \& P_5(x, y) \& P_6(x, z) \& P_7(x)) :=$  «все студенты всех университетов, обучающиеся на всех специальностях, имеют зачетные книжки».

Существуют предикаты, для которых область определения по различным предметным переменным ограничивают различными кванторами.

Например,  $\forall_x \exists_y (P_2(x, y)) :=$  «для любого целого числа  $x$  существует меньшее число  $y$ » или  $\exists_x \exists_y \exists_z (P_3(x, y, z)) :=$  «среди целых чисел существует число  $z$ , которое является частным от деления некоторого числа  $x$  на некоторое число  $y$ ».

Интересно отметить влияние на высказывание смены или перестановки кванторов.

$\exists_x \forall_y (P_2(x, y)) :=$  «существует целое число  $x$ , которое больше любого целого числа  $y$ ». Это высказывание явно ложное.

$\forall_x \forall_y (P_2(x, y)) :=$  « $x$  больше  $y$  для любых целых чисел». Это высказывание также ложное.

$\exists_x \exists_y (P_2(x, y)) :=$  «существуют целые числа  $x$ , для которых существуют меньшие число  $y$ ». Это высказывание истинное, но оно ограничивает область определения предиката.

$\exists_x \exists_y \forall_z (P_3(x, y, z)) :=$  «любое целое число  $z$  может быть частным от деления некоторого числа  $x$  на некоторое число  $y$ ». Это высказывание истинно для  $y=1$  и любого  $z$ .

Из рассмотренных примеров следует, что изменение кванторов или порядка их следования существенно изме-

няют смысл высказывания. Поэтому при исследовании текстов естественного языка следует очень аккуратно переходить к их формальному описанию.

Предметную переменную, если она находится под действием квантора, называют *связанной переменной*. И наоборот, если ее вхождение свободно от квантора, называют *свободной переменной*.

Например,  $\exists y(P_2(x, y))$  := «для любого целого числа существуют меньшие числа». В этом примере  $x$  – свободная, а  $y$  – связанная переменная квантором существования.

$\exists x \forall y (P_6(x, y) \& \neg P_7(x))$  := «во всех университетах существуют (есть) студенты, которые не имеют зачетной книжки». В этом примере  $x$  и  $y$  – связанные переменные кванторами существования и всеобщности соответственно.

$\exists x \forall y (P_1(x, y) \rightarrow \forall z (P_2(z)))$  все предметные переменные являются связанными.

$\forall z (P_1(z) \& \exists x (P_2(x, z)) \rightarrow (P_2(z, y) \vee (P_2(x, z)))$  предметные переменные  $x$  и  $z$  связанные, а  $y$  – свободная.

Свободная переменная может быть востребована на следующем шаге алгоритма, а связанная переменная служит для вычисления на данном шаге алгоритма значения предиката «и» или «л».

Между элементами области определения предиката могут быть заданы функциональные отношения. На это указывает функциональный символ  $f(x_1, x_2, \dots, x_n)$ .

Например, дату можно рассматривать как функцию, заданную на предметных переменных число, месяц, год. В этом случае функциональным символом является «дата», а аргументами <число>, <месяц> и <год>. Для предметных переменных структура этой функции есть «да-

та”” (“<число> ”, ”<месяц>”, ”<год>”)”. Для предметных постоянных она формирует высказывание. Например, «дата(1 января 2005 года)». Описание треугольника также можно рассматривать как некоторую структуру на предметных переменных, описывающих координаты вершин треугольника. Тогда функциональным символом является “треугольник”, а аргументами этой функции будут координаты вершин  $\langle x_1, y_1 \rangle$ ,  $\langle x_2, y_2 \rangle$ ,  $\langle x_3, y_3 \rangle$ . Для предметных переменных структура этой функции есть ”треугольник” (“< $x_1, y_1$ >”, ”< $x_2, y_2$ >”, ”< $x_3, y_3$ >”)”. При подстановке предметных постоянных будет дано описание конкретного треугольника.

### 1.2.1. Алгебра предикатов

Алгебры предикатов есть  $A_{\text{пред.}} = \langle T, F \rangle$ ,

где  $T = \{T_1, T_2, T_3, T_4\}$  - носитель алгебры, при этом  $T_1 = \{x, y, z, \dots\}$  - множество предметных переменных,  $T_2 = \{a, b, c, \dots\}$  - множество предметных постоянных,  $T_3 = \{f_1, f_2, f_3, \dots\}$  - множество функциональных символов и  $T_4 = \{P_1, P_2, P_3, \dots\}$  - множество предикатных символов,

$F = \{F_1, F_2, F_3\}$  - сигнатура алгебры, при этом  $F_1 = \{\neg, \&, \vee, \rightarrow, \leftrightarrow\}$  - множество логических связок,  $F_2 = \{\forall, \exists\}$  - множество кванторов,

$F_3 = \{\equiv\}$  - символ равносильности формул.

Любую предметную переменную и предметную постоянную предиката называют **термом** и обозначают символом  $t_i$ .

Если  $f_i$  - функциональный символ и  $t_1, t_2, \dots, t_n$  - его термы, то  $f(t_1, t_2, \dots, t_n)$  также есть терм, где  $n$  - число аргументов функции,  $i$  - индекс функции.

Никаких иных термов нет.

Если  $P_i$  – предикатный символ и  $t_1, t_2, \dots, t_n$  – термы, то

$F = P(t_1, t_2, \dots, t_n)$  – *элементарная формула*.

Если  $F_1$  и  $F_2$  – формулы, то  $\neg F_1, \neg F_2, (F_1 \& F_2), (F_1 \vee F_2), (F_1 \rightarrow F_2), (F_1 \leftrightarrow F_2)$  – также формулы.

Если  $F$  – формула, а  $x$  – предметная переменная, входящая в формулу  $F$ , то  $\forall_x(F)$  и  $\exists_x(F)$  – также формулы.

Никаких иных формул нет.

#### 1.2.1.1. Логические операции

Простейшими операциями в логике предикатов являются те же операции, что в алгебре высказываний.

**Отрицание** ( $\neg F(t_1, t_2, \dots, t_n)$ ) есть одноместная операция, посредством которой из данной формулы  $F(t_1, t_2, \dots, t_n)$  получают ее отрицание.

Пример 1.51. Если  $P(x, 'a') :=$  « $x$  находится на  $a$ », где ' $a$ ' = 'стол', то выводимы формулы:

а)  $\forall_x(\neg P(x, 'a')) :=$  «для всех  $x$  верно, что  $x$  не находится на столе».

б)  $\neg \forall_x(P(x, 'a')) :=$  «не для всех  $x$  верно, что  $x$  находится на столе»,

с)  $\neg \exists_x(P(x, 'a')) :=$  «нет таких  $x$ , для которых верно, что  $x$  находится на столе».

**Конъюнкция** ( $F_1(t_{11}, t_{12}, \dots, t_{1n}) \& F_2(t_{21}, t_{22}, \dots, t_{2n})$ ) есть двуместная операция, посредством которой из формул  $F_1$  и  $F_2$  получают новую формулу  $F(t_{11}, t_{12}, \dots, t_{1n}, t_{21}, t_{22}, \dots, t_{2n})$  с числом предметных переменных и постоянных, равным

объединению их в исходных формулах. Значение формулы истинно тогда и только тогда, когда истинны обе формулы  $F_1$  и  $F_2$ .

Пример 1.52. Если  $P_1(x)$  := «выдающийся музыкант» и  $P_2(x)$  := «талантливый писатель», то выводимы формулы:

а)  $\exists x(P_1(x)) \& \exists x(P_2(x))$  := «существуют выдающиеся музыканты и существуют талантливые писатели»,

б)  $\exists x(P_1(x) \& P_2(x))$  := «существуют лица, которые являются талантливыми писателями и выдающимися музыкантами».

с)  $\neg \forall x(P_1(x) \& P_2(x))$  := «не все талантливые писатели являются выдающимися музыкантами».

Пример 1.53. Если  $x$  - предметная переменная (индивид),  $a$  - предметная постоянная (например  $'a'$  := «Саша») и  $P_1(x, 'a')$  := « $x$  дружит с  $'a'$ »,  $P_2(x, 'a')$  := « $x$  встретил  $'a'$ », то выводимы формулы :

а)  $\exists x(P_1(x, 'a') \& P_2(x, 'a'))$  := «Саша встретил друга»,

б)  $\exists x(\neg P_1(x, 'a') \& P_2(x, 'a'))$  := «Саша встретил недруга»,

с)  $\neg \forall x(P_1(x, 'a') \& P_2(x, 'a'))$  := «не каждый встречный есть друг Саши»,

д)  $\exists x(P_1(x, 'a') \& (\neg P_2(x, 'a')))$  := «есть друзья, с которыми Саша не встречается».

Обратите внимание, как меняется смысл каждого высказывания.

**Дизъюнкция**  $(F_1(t_{11}, t_{12}, \dots, t_{1n}) \vee F_2(t_{21}, t_{22}, \dots, t_{2n}))$  есть двуместная операция, посредством которой из формул  $F_1$  и

$F_2$  получают новую формулу  $F(t_{11}, t_{12}, \dots, t_{1n}, t_{21}, t_{22}, \dots, t_{2n})$  с числом предметных переменных и постоянных, равным объединению их в исходных формулах. Значение формулы ложно тогда и только тогда, когда ложны обе формулы  $F_1$  и  $F_2$ .

Пример 1.54. Если предметные переменные  $x, y$  - города России и  $P_1(x, y) :=$  «транспортная связь  $x$  и  $y$  поездом»,  $P_2(x, y) :=$  «транспортная связь  $x$  и  $y$  самолетом»,  $P_3(x, y) :=$  «транспортная связь  $x$  и  $y$  автобусом», то выводимы формулы:

а)  $\forall x \forall y (P_1(x, y) \vee P_2(x, y) \vee P_3(x, y)) :=$  «для всех городов России ( $x$  и  $y$ ) возможна транспортная связь поездом, автобусом или самолетом»,

б)  $\neg \forall x \exists y (P_1(x, y) \vee \neg P_2(x, y) \vee \neg P_3(x, y))$  - «не для всех городов  $x$  существуют города  $y$ , между которыми нет транспортной связи автобусом или самолетом, но есть поездом».

**Импликация** ( $F_1(t_{11}, t_{12}, \dots, t_{1n}) \rightarrow F_2(t_{21}, t_{22}, \dots, t_{2n})$ ) есть двухместная операция, посредством которой из формул  $F_1$  и  $F_2$  получают новую формулу  $F(t_{11}, t_{12}, \dots, t_{1n}, t_{21}, t_{22}, \dots, t_{2n})$  с числом предметных переменных и постоянных, равным объединению их в исходных формулах. Значение формулы ложно тогда и только тогда, когда  $F_1$  истинно, а  $F_2$  - ложно.

Пример 1.55. Если  $x$  - индивид,  $P_1(x) :=$  «быть судьей»,  $P_2(x) :=$  «быть юристом», то выводимы формулы:

а)  $\forall x (P_1(x) \rightarrow P_2(x)) :=$  «все судьи – юристы»,

б)  $\neg \forall x (P_2(x) \rightarrow P_1(x)) :=$  «неверно, что все юристы – су-

дья»,

Пример 1.56. Если  $x$  - предметная переменная для индивида,  $P_1(x)$  := « $x$  принадлежит к большинству»,  $P_2(x)$  := « $x$  стремится к миру», то выводима формула:

а)  $\forall x(P_1(x) \& (P_1(x) \rightarrow P_2(x)))$  := «большинство стремится к миру».

б)  $\exists x(\neg P_1(x) \& (P_1(x) \rightarrow P_2(x)))$  := «существуют люди, не принадлежащие к большинству, которые не стремятся к миру».

**Эквивалентность** ( $F_1(t_{11}, t_{12}, \dots, t_{1n}) \leftrightarrow F_2(t_{21}, t_{22}, \dots, t_{2n})$ ) есть двуместная операция, посредством которой из формул  $F_1$  и  $F_2$  получают новую формулу  $F(t_{11}, t_{12}, \dots, t_{1n}, t_{21}, t_{22}, \dots, t_{2n})$  с числом предметных переменных и постоянных, равным объединению их в исходных формулах. Значение формулы истинно тогда и только тогда, когда обе формулы  $F_1$  и  $F_2$  имеют одно и то же значение истины или лжи.

Пример 1.57. Если  $x$  - предметная переменная,  $P(x)$  - предикат, то выводимы формула:

а)  $\exists x(P(x)) \leftrightarrow \neg \forall x(\neg P(x))$  := «существует переменная  $x$ , для которой  $P(x)$  истинно эквивалентна не для всех  $x$   $P(x)$  ложно».

Рассмотренные логические операции позволяют формализовать с помощью термов, предикатов и кванторов внутреннюю структуру предложения и формировать сложные суждения.

### 1.2.1.2. Правила записи сложных формул

Пример 1.58. Дано «некоторые действительные числа являются рациональными». В этом суждении есть два предиката  $P_1(x)$  := « $x$  - действительное число» и  $P_2(x)$  := « $x$  - рациональное число».

Формула этого суждения есть  $F = \exists x(P_1(x) \& P_2(x))$ .

Ошибочными являются формулы  $F = \exists x(P_1(x) \rightarrow P_2(x))$  := «некоторые числа, если они действительные, то они рациональные», и  $F = \exists x(\neg P_1(x) \vee P_2(x))$  := «некоторые числа не являются действительными или являются рациональными».

Пример 1.59. «Некоторые социал-демократы уважают всех демократов. Ни один социал-демократ не уважает ни одного коммуниста. Следовательно, ни один демократ не является коммунистом». В этом суждении три одноместных предиката  $P_1(x)$  := «быть социал-демократом»,  $P_2(x)$  := «быть демократом»,  $P_3(x)$  := «быть коммунистом» и один двухместный предикат  $P_4(x, y)$  := « $x$  уважает  $y$ ».

Формула сложного суждения имеет вид:

$$\frac{\exists x(P_1(x) \& \forall y(P_2(y) \rightarrow P_4(x, y))), \neg \forall x(P_1(x) \rightarrow \forall y(P_3(y) \rightarrow \neg P_4(x, y)))}{\forall x(P_2(x) \rightarrow \neg P_3(x))}$$

Пример 1.60. «Ни один торговец наркотиками не является наркоманом. Некоторые наркоманы привлекались к ответственности. Следовательно, некоторые люди, привлекавшиеся к ответственности, не являются торговцами наркотиков»[15].

В этом суждении три предиката  $P_1(x)$  := «быть торговцем наркотиков»,  $P_2(x)$  := «быть наркоманом»,  $P_3(x)$  := «привлекаться к ответственности».

Формула этого суждения имеет вид:

$$\frac{\forall_x (P_1(x) \rightarrow P_2(x)), \exists_x (P_2(x) \& P_3(y))}{\exists_x (P_3(x) \& \neg P_1(x))}.$$

Примеры позволяют сформулировать некоторые правила:

- каждое вхождение логической связки « $\rightarrow$ » относится к формуле, следующей непосредственно за логической связкой справа,
- каждое вхождение логических связок « $\&$ » или « $\vee$ » после расстановки скобок связывает формулы, непосредственно окружающие логическую связку,
- логические связки по силе упорядочены так:  $\neg$ ,  $\&$ ,  $\vee$ ,  $\rightarrow$ ,  $\leftrightarrow$ .
- за квантором общности чаще всего следует импликация, а за квантором существования - конъюнкции,
- если формула содержит подформулу, то последняя не должна содержать кванторов, связывающих ту же переменную, что и квантор формулы,
- значения всех предметных переменных и постоянных должны принадлежать одной области определения предиката или функции,
- если в одной формуле есть кванторы всеобщности и существования, то при формализации суждений следует стремиться поставить квантор существования слева от всей формулы.

### 1.2.1.3. Законы алгебры предикатов

Основные законы алгебры предикатов приведены в таблице 1.11.

Равносильные формулы алгебры высказываний верны в алгебре предикатов, если в них вместо высказываний подставить предикаты и термы. Но при наличии в формуле кванторов усложняется исполнение некоторых алгебраических операций.

Таблица 1.11.

Имя закона	Равносильные формулы $F_i \equiv F_j$
Коммутативности	$\forall_x \forall_y (F(x, y)) \equiv \forall_y \forall_x (F(x, y))^*$ , $\exists_x \exists_y (F(x, y)) \equiv \exists_y \exists_x (F(x, y))^*$ . * - только для одноименных кванторов.
Дистрибутивности	$\forall_x (F_1(x)) \& \forall_x (F_2(x)) \equiv \forall_x (F_1(x) \& F_2(x))^*$ , $\exists_x (F_1(x)) \vee \exists_x (F_2(x)) \equiv \exists_x (F_1(x) \vee F_2(x))^{**}$ , * - только для формул с кванторами $\forall_x$ по одной переменной $x$ . ** - только для формул с кванторами $\exists_x$ по одной переменной $x$ .
Идемпотентности	$\mathfrak{R}_x(F(x)) \vee \mathfrak{R}_x(F(x)) \equiv \mathfrak{R}_x(F(x))$ , $\mathfrak{R}_x(F(x)) \& \mathfrak{R}_x(F(x)) \equiv \mathfrak{R}_x(F(x))$ для $\mathfrak{R} \in \{\forall, \exists\}$ .
Исключенного третьего	$\mathfrak{R}_x(F(x)) \vee \neg \mathfrak{R}_x(F(x)) =$ и для $\mathfrak{R} \in \{\forall, \exists\}$
Противоречия	$\mathfrak{R}_x(F(x)) \& \neg \mathfrak{R}_x(F(x)) =$ л для $\mathfrak{R} \in \{\forall, \exists\}$

Отрицание кванторов	$\forall_x(\neg F(x)) \equiv \neg \exists_x(F(x)), \exists_x(\neg F(x)) \equiv \neg \forall_x(F(x)),$ $\forall_x(F(x)) \equiv \neg \exists_x(\neg F(x)), \exists_x(F(x)) \equiv \neg \forall_x(\neg F(x)),$
Дополнения	$\neg(\neg \mathfrak{R}_x(F(x))) \equiv \mathfrak{R}_x(F(x)),$ для $\mathfrak{R} \in \{\forall, \exists\}$

Две формулы  $F_i(t_1, t_2, \dots, t_n)$  и  $F_j(t_1, t_2, \dots, t_n)$  называются **равносильными**, если они при одинаковых наборах термов имеют одинаковое значение. Равносильность формул будем обозначать символом « $\equiv$ », т.е.  $(F_1 \equiv F_2)$ . Если две формулы равносильны  $F_1(t_1, t_2, \dots, t_n) \equiv F_2(t_1, t_2, \dots, t_n)$ , то они эквивалентны. Эквивалентность формул будем обозначать символом « $\leftrightarrow$ », т.е.  $(F_i(t_1, t_2, \dots, t_n) \leftrightarrow F_j(t_1, t_2, \dots, t_n))$ . И наоборот, если две формулы эквивалентны  $(F_i \leftrightarrow F_j)$ , то они равносильны  $(F_i \equiv F_j)$ .

Если в составе формулы  $F$  есть формула  $F_i$ , т.е.  $F(t_1, t_2, \dots, F_i, \dots, t_n)$ , и для  $F_i$  существует эквивалентная формула  $F_j$ , т.е.  $(F_i \leftrightarrow F_j)$ , то возможна подстановка формулы  $F_j$  в формулу  $F$  без нарушения истинности формулы, т.е.

$$F(t_1, t_2, \dots, F_i, \dots, t_n) \equiv F(t_1, t_2, \dots, F_j, \dots, t_n).$$

Докажем равносильность некоторых законов.

• Дано  $\forall_x(F_1(x)) \& \forall_x(F_2(x)) \equiv \forall_x(F_1(x) \& F_2(x))$ .

Если предикаты  $F_1(x)=и$ ,  $F_2(x)=и$ ,  $F_1(x) \& F_2(x)=и$ , то истинны  $\forall_x(F_1(x))=и$ ,  $\forall_x(F_2(x))=и$ ,  $\forall_x(F_1(x)) \& \forall_x(F_2(x))=и$ ,  $\forall_x(F_1(x) \& F_2(x))=и$ .

Если предикаты  $F_1(x)=л$ ,  $F_2(x)=и$ ,  $F_1(x)\&F_2(x)=л$ , то ложны  $\forall_x(F_1(x))=л$ ,  $\forall_x(F_1(x))\&\forall_x(F_2(x))=л$ ,  $\forall_x(F_1(x)\&F_2(x))=л$ .

Следовательно,  $\forall_x(F_1(x))\&\forall_x(F_2(x))\equiv\forall_x(F_1(x)\&F_2(x))$ .

• Дано  $\exists_x(F_1(x))\vee\exists_x(F_2(x))\equiv\exists_x(F_1(x)\vee F_2(x))$ .

Если предикаты  $F_1(x)=и$ ,  $F_2(x)=и$ ,  $F_1(x)\vee F_2(x)=и$ , то истинны  $\exists_x(F_1(x))=и$ ,  $\exists_x(F_2(x))=и$ ,  $\exists_x(F_1(x))\vee\exists_x(F_2(x))=и$ ,  $\exists_x(F_1(x)\vee F_2(x))=и$ .

Если предикаты  $F_1(x)=л$ ,  $F_2(x)=и$ ,  $F_1(x)\vee F_2(x)=л$ , то ложны  $\exists_x(F_1(x))=л$ ,  $\exists_x(F_1(x))\vee\exists_x(F_2(x))=л$ ,  $\exists_x(F_1(x)\vee F_2(x))=л$ .

Следовательно,  $\exists_x(F_1(x))\vee\exists_x(F_2(x))\equiv\exists_x(F_1(x)\vee F_2(x))$ .

• Дано  $\forall_x(\neg F(x))\equiv\neg\exists_x(F(x))$ .

Если предикат  $\neg F(x)=и$ , а  $F(x)=л$ , то  $\forall_x(\neg F(x))=и$ ,  $\exists_x(F(x))=л$ ,  $\neg\exists_x(F(x))=и$ .

Если предикат  $\neg F(x)=л$ , а  $F(x)=и$ , то  $\forall_x(\neg F(x))=л$ ,  $\exists_x(F(x))=и$ ,  $\neg\exists_x(F(x))=л$ .

Следовательно,  $\forall_x(\neg F(x))\equiv\neg\exists_x(F(x))$ .

Если в формуле  $F$ , содержащей свободную переменную  $x$ , выполнить всюду подстановку вместо переменной  $x$  терма  $t$ , то получим формулу  $F(t)$ . Этот факт записывают

так:  $\frac{\int_x F(x)}{F(t)}$

Подстановка называется **правильной**, если в формуле всюду вместо свободной переменной  $x$  выполнена подстановка терма  $t$ .

Например,

$$\frac{\int_{x_1}^{x_2} \exists x_3 (P_1(x_1, x_3) \rightarrow P_2(x_2))}{\exists x_3 (P_1(x_2, x_3) \rightarrow P_2(x_2))} .$$

Это - правильная подстановка, так как вместо свободной переменной можно подставить другую переменную.

$$\frac{\int_{x_1}^{f(x_2, t)} \exists x_3 (P_1(x_1, x_3) \rightarrow P_2(x_2))}{\exists x_3 (P_1(f(x_2, t), x_3) \rightarrow P_2(x_2))} .$$

Это - правильная подстановка, так как вместо свободной переменной можно подставить функцию.

Подстановка называется **неправильной**, если в результате подстановки свободная переменная окажется в области действия квантора или связанная переменная будет заменена термом.

$$\frac{\int_{x_3}^{x_2} \exists x_3 (P_1(x_1, x_3) \rightarrow P_2(x_2))}{\exists x_3 (P_1(x_1, x_2) \rightarrow P_2(x_2))} .$$

Это - неправильная подстановка, так как связанная  $x_3$  заменена свободной переменной.

$$\frac{\int_{x_2}^{x_3} \exists x_3 (P_1(x_1, x_3) \rightarrow P_2(x_2))}{\exists x_3 (P_1(x_1, x_3) \rightarrow P_2(x_3))} .$$

Это - неправильная подстановка, так как предикат  $P_2(\cdot)$  попадает в область действия квантора  $\exists x_3$ .

#### 1.2.1.4. Эквивалентные преобразования формул

Пример. 1.61.  $F = \neg \exists x_1 \forall x_2 (P_1(x_1) \rightarrow \forall x_3 (P_2(x_1, x_3) \vee P_3(x_2, x_3)))$ .

Упростить алгебраическое выражение.

- Выполнить операцию отрицания:

$$F = \forall_{x_1} \neg \forall_{x_2} (P_1(x_1) \rightarrow \forall_{x_3} (P_2(x_1, x_3) \vee P_3(x_2, x_3))),$$

$$F = \forall_{x_1} \exists_{x_2} (\neg (P_1(x_1) \rightarrow \forall_{x_3} (P_2(x_1, x_3) \vee P_3(x_2, x_3)))).$$

- Удалить логическую связку « $\rightarrow$ »:

$$F = \forall_{x_1} \exists_{x_2} \neg (\neg P_1(x_1) \vee \forall_{x_3} (P_2(x_1, x_3) \vee P_3(x_2, x_3))).$$

- Выполнить операцию отрицания:

$$F = \forall_{x_1} \exists_{x_2} (P_1(x_1) \& \neg \forall_{x_3} (P_2(x_1, x_3) \vee P_3(x_2, x_3))),$$

$$F = \forall_{x_1} \exists_{x_2} (P_1(x_1) \& \exists_{x_3} (\neg (P_2(x_1, x_3) \vee P_3(x_2, x_3)))),$$

$$F = \forall_{x_1} \exists_{x_2} (P_1(x_1) \& \exists_{x_3} (\neg P_2(x_1, x_3) \& \neg P_3(x_2, x_3))).$$

- Перенести квантор  $\exists_{x_3}$  влево:

$$F = \forall_{x_1} \exists_{x_2} \exists_{x_3} (P_1(x_1) \& \neg P_2(x_1, x_3) \& \neg P_3(x_2, x_3)).$$

Пример 1.62.  $F = \forall_x (P_1(x) \rightarrow \neg P_2(x)) \rightarrow \neg (\exists_x (P_1(x))$

$\& \forall_x (P_2(x)))$ .

Упростить алгебраическое выражение.

- Удалить логическую связку « $\rightarrow$ »:

$$F = \neg (\forall_x (\neg P_1(x) \vee \neg P_2(x))) \vee \neg (\exists_x (P_1(x)) \& \forall_x (P_2(x))).$$

- Выполнить операцию отрицания:

$$F = \exists_x (\neg (\neg P_1(x) \vee \neg P_2(x))) \vee \forall_x (\neg P_1(x)) \vee \exists_x (\neg P_2(x)),$$

$$F = \exists_x (P_1(x) \& P_2(x)) \vee \forall_x (\neg P_1(x)) \vee \exists_x (\neg P_2(x)).$$

- Применить закон дистрибутивности по квантору  $\exists_x$ :

$$F = \exists_x (P_1(x) \& P_2(x) \vee \neg P_2(x)) \vee \forall_x (\neg P_1(x)),$$

$$F = \exists_x ((P_1(x) \vee \neg P_2(x)) \& (P_2(x) \vee \neg P_2(x))) \vee \forall_x (\neg P_1(x)).$$

- Применить закон исключенного третьего:

$$F = \exists_x ((P_1(x) \vee \neg P_2(x)) \vee \forall_x (\neg P_1(x))).$$

- Применить закон де Моргана:

$$F = \exists_x ((P_1(x) \vee \neg P_2(x)) \vee \neg \exists_x (P_1(x))).$$

- Применить закон дистрибутивности по квантору  $\exists_x$ :

$$F = \exists_x(P_1(x)) \vee \exists_x(\neg P_2(x)) \vee \neg \exists_x(P_1(x)).$$

- Применить закон исключенного третьего:

$$F = \exists_x(\neg P_2(x)) \vee \text{и} = \text{и}.$$

Следовательно,  $\forall_x(P_1(x) \rightarrow \neg P_2(x)) \rightarrow \neg(\exists_x(P_1(x)) \& \& \forall_x(P_2(x)))$  - аксиома.

### 1.2.1.2. Предварённая нормальная форма

Для удобства анализа сложных формул рекомендуется преобразовывать ее к нормальной форме. Если в алгебре высказываний приняты две нормальные формы ДНФ и КНФ, то в алгебре предикатов – кроме ДНФ и КНФ есть **предварённая нормальная форма** (ПНФ), суть которой сводится к разделению формулы на две части: **префикс** и **матрицу**. Для этого все кванторы выносятся влево по правилам логики предикатов, формируя префикс, а логические связки соединяют предикаты формулы, формируя матрицу. В результате будет получена формула:  $\mathfrak{R}_{x_1} \mathfrak{R}_{x_2} \dots \mathfrak{R}_{x_n}(M)$ , где  $\mathfrak{R}_{x_1} \mathfrak{R}_{x_2} \dots \mathfrak{R}_{x_n}$  - префикс формулы при  $\mathfrak{R} \in \{\forall, \exists\}$ , а  $M$  – матрица формулы. Затем матрицу формулы преобразуют к виду КНФ для определения истинности заключения по правилам 1.1.1.5.

#### Алгоритм приведения формулы к виду ПНФ:

**Шаг 1:** исключить в формуле всюду логические связки  $\leftrightarrow$  и  $\rightarrow$ :

$$(F_1 \leftrightarrow F_2) = (F_1 \rightarrow F_2) \& (F_2 \rightarrow F_1) = (\neg F_1 \vee F_2) \& (\neg F_2 \vee F_1),$$

$$(F_1 \rightarrow F_2) = (\neg F_1 \vee F_2),$$

**Шаг 2:** продвинуть отрицание до элементарной формулы:

$$\neg \forall x(F) = \exists x(\neg F), \quad \neg(F_1 \vee F_2) = (\neg F_1 \& \neg F_2), \quad \neg \exists x(F) = \forall x(\neg F),$$

$$\neg(F_1 \& F_2) = (\neg F_1 \vee \neg F_2),$$

**Шаг 3:** переименовать связанные переменные по правилу: «найти самое левое вхождение предметной переменной – такое, что это вхождение связано некоторым квантором, но существует еще одно вхождение этой же переменной, затем сделать замену связанного вхождения на вхождение новой переменной. Операцию повторять пока возможна замена связанных переменных»,

**Шаг 4:** вынести кванторы новых связанных переменных влево, не нарушая их последовательности.

**Шаг 5:** преобразовать бескванторную матрицу к виду КНФ, т. е.  $M = D_1 \& D_2 \& D_3 \& \dots$ , где  $D_i = (F_i \vee F_j \vee F_k \vee \dots)$ . Алгоритм приведения матрицы формулы к виду КНФ приведен в 1.1.1.5.

**Пример 1.63.** Дано  $F = \neg \exists x_1 \forall x_2 (P_1(x_1) \rightarrow \forall x_3 (P_2(x_1, x_3) \vee P_3(x_2, x_3)))$ . Привести формулу к виду ПНФ.

- Выполнить операцию отрицания:

$$F = \forall x_1 (\neg \forall x_2 (P_1(x_1) \rightarrow \forall x_3 (P_2(x_1, x_3) \vee P_3(x_2, x_3))))),$$

$$F = \forall x_1 \exists x_2 (\neg (P_1(x_1) \rightarrow \forall x_3 (P_2(x_1, x_3) \vee P_3(x_2, x_3)))).$$

- Удалить логическую связку « $\rightarrow$ »:

$$F = \forall x_1 \exists x_2 (\neg (\neg P_1(x_1) \vee \forall x_3 (P_2(x_1, x_3) \vee P_3(x_2, x_3)))).$$

- Выполнить отрицание формулы:

$$F = \forall x_1 \exists x_2 (P_1(x_1) \& \neg \forall x_3 (P_2(x_1, x_3) \vee P_3(x_2, x_3))),$$

$$F = \forall x_1 \exists x_2 (P_1(x_1) \& \exists x_3 (\neg (P_2(x_1, x_3) \vee P_3(x_2, x_3)))),$$

$$F = \forall x_1 \exists x_2 (P_1(x_1) \& \exists x_3 (\neg P_2(x_1, x_3) \& \neg P_3(x_2, x_3))).$$

- Перенести квантор  $\exists x_3$  влево:

$$F = \forall x_1 \exists x_2 \exists x_3 (P_1(x_1) \& \neg P_2(x_1, x_3) \& \neg P_3(x_2, x_3)).$$

Матрица ПНФ содержит три элементарных дизъюнкта:

$$K = \{P_1(x_1), \neg P_2(x_1, x_3), \neg P_3(x_2, x_3)\}.$$

Пример 1.64. Дано  $F = (\forall_x(P_1(x) \rightarrow \forall_y(P_2(y) \rightarrow P_3(z)))) \& (\neg \forall_y(P_4(x, y) \rightarrow P_5(z)))$ . Привести формулу к виду ПНФ.

- Удалить логические связки « $\rightarrow$ »:

$$F = \forall_x(\neg P_1(x) \vee \forall_y(\neg P_2(y) \vee P_3(z))) \& \neg \forall_y(\neg P_4(x, y) \vee P_5(z)),$$

- Применить закон  $\neg \forall_x(F(x)) \equiv \exists_x(\neg F(x))$ :

$$F = \forall_x(\neg P_1(x) \vee \forall_y(\neg P_2(y) \vee P_3(z))) \& \exists_y(\neg(\neg P_4(x, y) \vee P_5(z))).$$

- Применить закон  $\neg(F_1 \vee F_2) \equiv (\neg F_1 \& \neg F_2)$ :

$$F = \forall_x(\neg P_1(x) \vee \forall_y(\neg P_2(y) \vee P_3(z))) \& \exists_y(P_4(x, y) \& \neg P_5(z)).$$

- Переименовать связанную переменную левого квантора  $x=w$ :

$$F = \forall_w(\neg P_1(w) \vee \forall_y(\neg P_2(y) \vee P_3(z))) \& \exists_y(P_4(x, y) \& \neg P_5(z)).$$

- Переименовать связанную переменную левого квантора  $y=v$ :

$$F = \forall_w(\neg P_1(w) \vee \forall_v(\neg P_2(v) \vee P_3(z))) \& \exists_y(P_4(x, y) \& \neg P_5(z)).$$

- Вынести квантор  $\forall_v$  в префикс:

$$F = \forall_w \forall_v(\neg P_1(w) \vee \neg P_2(v) \vee P_3(z)) \& \exists_y(P_4(x, y) \& \neg P_5(z)).$$

- Вынести квантор  $\exists_y$  в префикс:

$$F = \forall_w \forall_v \exists_y((\neg P_1(w) \vee \neg P_2(v) \vee P_3(z)) \& P_4(x, y) \& \neg P_5(z)).$$

Матрица ПНФ содержит три элементарных дизъюнкта:

$$K = \{(\neg P_1(w) \vee \neg P_2(v) \vee P_3(z)), P_4(x, y), \neg P_5(z)\}.$$

Пример 1.65. Дано  $F = \forall_x(P_1(x) \leftrightarrow \exists_y(P_2(y))) \rightarrow \forall_z(P_3(z))$ .

Привести формулу к виду ПНФ.

- Удалить логические связки « $\leftrightarrow$ » и « $\rightarrow$ »:

$$F = \neg \forall_x(P_1(x)) \& \exists_y(P_2(y)) \vee (\neg(P_1(x)) \& \neg \exists_y(P_2(y))) \vee \forall_z(P_3(z)).$$

- Применить закон де Моргана:

$$F = \exists_x((\neg P_1(x) \vee \forall_y(\neg P_2(y))) \& (P_1(x) \vee \exists_y(P_2(y)))) \vee \forall_z(P_3(z)).$$

- Применить закон дистрибутивности:

$$F = \exists_x(\neg P_1(x) \& P_1(x) \vee \neg P_1(x) \& \exists_y(P_2(y)) \vee P_1(x) \& \forall_y(\neg P_2(y)) \vee \forall_y(\neg P_3(y)) \& \exists_y(P_2(y))) \vee \forall_z(P_3(z)) = \\ \exists_x(\neg P_1(x) \& \exists_y(P_2(y)) \vee P_1(x) \& \forall_y(\neg P_2(y))) \vee \forall_z(P_3(z)).$$

- Переименовать связанную переменную левого квантора  $y=v$ :

$$F = \exists_x(\neg P_1(x) \& \exists_v(P_2(v)) \vee P_1(x) \& \forall_y(\neg P_2(y))) \vee \forall_z(P_3(z)).$$

- Вынести кванторы  $\exists_v$ , и  $\forall_z$  в префикс:

$$F = \exists_x \exists_v \forall_y (\neg P_1(x) \& P_2(v) \vee P_1(x) \& \neg P_2(y) \vee P_3(z)).$$

- Преобразовать матрицу к виду КНФ:

$$F = \exists_x \exists_v \forall_y ((\neg P_1(x) \vee P_1(x) \vee P_3(z)) \& (\neg P_1(x) \vee \neg P_2(y) \vee P_3(z)) \& \\ (P_2(v) \vee P_1(x) \vee P_3(z)) \& (P_2(v) \vee \neg P_2(y) \vee P_3(z))).$$

Матрица ПНФ содержит три элементарных дизъюнкта:

$$K = \{(\neg P_1(x) \vee \neg P_2(y) \vee P_3(z)), (P_1(x) \vee P_2(v) \vee P_3(z)), (P_2(v) \vee \neg P_2(y) \vee P_3(z)).$$

Пример 1.66. Дано  $F = (\forall_x(P_1(x) \rightarrow \forall_y(P_2(y) \rightarrow P_3(z)))) \& (\forall_y(P_4(x, y) \rightarrow P_5(z)))$ . Привести формулу к виду ПНФ.

- Удалить логические связки « $\rightarrow$ »:

$$F = (\forall_x(\neg P_1(x) \vee \forall_y(\neg P_2(y) \vee P_3(z)))) \& (\forall_y(\neg P_4(x, y) \vee P_5(z))).$$

- Применить закон  $\neg \forall_x(F(x)) \equiv \exists_x(\neg F(x))$ :

$$F = (\forall_x(\neg P_1(x) \vee \forall_y(\neg P_2(y) \vee P_3(z)))) \& (\exists_y(\neg(\neg P_4(x, y) \vee P_5(z)))).$$

- Применить закон  $\neg(F_1 \vee F_2) \equiv (\neg F_1 \& \neg F_2)$ :

$$F = \forall_x(\neg P_1(x) \vee \forall_y(\neg P_2(y) \vee P_3(z))) \& (\exists_y(P_4(x, y) \& (\neg P_5(z)))).$$

- Переименовать связанную переменную  $x=w$ :

$$F = \forall_w(\neg P_1(w) \vee \forall_y(\neg P_2(y) \vee P_3(z))) \& (\exists_y(P_4(x, y) \& (\neg P_5(z)))).$$

- Переименовать связанную переменную  $y=v$ :

$$F = \forall_w(\neg P_1(w) \vee \forall_v(\neg P_2(v) \vee P_3(z))) \& (\exists_y(P_4(x, y) \& (\neg P_5(z)))).$$

- Вынести квантор  $\forall_v$  влево:

$$F = \forall_w \forall_v (\neg P_1(w) \vee \neg P_2(v) \vee P_3(z)) \& (\exists_y (P_4(x, y) \& (\neg P_5(z)))).$$

• Вынести квантор  $\exists_y$  влево:

$$F = \forall_w \forall_v \exists_y (\neg P_1(w) \vee \neg P_2(v) \vee P_3(z)) \& P_4(x, y) \& \neg P_5(z).$$

Матрица ПНФ содержит три элементарных дизъюнкта:

$$K = \{(\neg P_1(w) \vee \neg P_2(v) \vee P_3(z)), P_4(x, y), \neg P_5(z)\}.$$

Пример 1.67. Дано  $F = \forall_x \exists_y (P_1(x, y) \& \neg (\exists_x \forall_y (P_2(x, y))))$ .

Привести формулу к виду ПНФ.

• Применить закон  $\neg \exists_x (F(x)) \equiv \forall_x (\neg F(x))$ :

$$F = \forall_x \exists_y (P_1(x, y) \& (\forall_x \neg (\forall_y (P_2(x, y)))).$$

• Применить закон  $\neg \forall_x (F(x)) \equiv \exists_x (\neg F(x))$ :

$$F = \forall_x \exists_y (P_1(x, y) \& (\forall_x \exists_y (\neg (P_2(x, y)))).$$

• Вынести квантор  $\forall_x$  влево:

$$F = \forall_x (\exists_y (P_1(x, y) \& \exists_y (\neg (P_2(x, y)))).$$

• Переименовать связанную переменную  $y=v$ :

$$F = \forall_x (\exists_z (P_1(x, z) \& \exists_y (\neg (P_2(x, y)))).$$

• Вынести кванторы  $\exists_z$  и  $\exists_y$  влево:

$$\forall_x \exists_z \exists_y (P_1(x, z) \& \neg P_2(x, y)).$$

Матрица ПНФ содержит два элементарных дизъюнкта:

$$K = \{P_1(x, z), \neg P_2(x, y)\}.$$

Пример 1.68. Дана матрица

$M = P_1(z) \& \neg P_2(w) \vee P_2(x) \& \neg P_1(z) \vee P_3(y)$ . Привести формулу матрицы к виду КНФ.

• Применить закон дистрибутивности:

$$M = P_1(z) \& \neg P_2(w) \vee (P_2(x) \vee P_3(y)) \& (\neg P_1(z) \vee (P_3(y))).$$

$$M = (P_1(z) \& \neg P_2(w) \vee P_2(x) \vee P_3(y)) \& (P_1(z) \& \neg P_2(w) \vee \neg P_1(z) \vee P_3(y)).$$

$$M = (P_1(z) \vee P_2(x) \vee P_3(y)) \& (\neg P_2(w) \vee P_2(x) \vee P_3(y)) \& (\neg P_1(z) \vee \neg P_1(z) \vee P_3(y)) \& (\neg P_2(w) \vee \neg P_1(z) \vee P_3(y)).$$

• Применить закон исключенного третьего:

$$M = (P_1(z) \vee P_2(x) \vee P_3(y)) \& (\neg P_2(w) \vee P_2(x) \vee P_3(y)) \& (\neg P_2(w) \vee \neg P_1(z) \vee P_3(y)).$$

Матрица содержит три элементарных дизъюнкта  $K = \{(P_1(z) \vee P_2(x) \vee P_3(y)), (\neg P_2(w) \vee P_2(x) \vee P_3(y)), (\neg P_2(w) \vee \neg P_1(z) \vee P_3(y))\}$ .

Дизъюнкты матрицы содержат контрарные атомы  $P_1(z)$  и  $\neg P_1(z)$ ,  $P_2(x)$  и  $\neg P_2(w)$ . Если имена переменных различны (например,  $x$  и  $w$ ), то выполняют унификацию дизъюнктов подстановкой вместо одной переменной другой или промежуточных термов (подробнее см. 1.2.2.3).

### 1.2.1.3. Сколемовская стандартная форма

Наличие разноименных кванторов в префиксе не позволяет осуществлять вывод заключения, опираясь только на матрицу. Однако есть эффективный алгоритм Сколема, удаляющий из префиксной части кванторы существования и преобразующий формулу к виду  $F = \forall_{x_1} \forall_{x_2} \dots \forall_{x_n} (M)$ . В этом случае вывод заключения возможен только по формуле матрицы. Для устранения в префиксе кванторов существования вводится *сколемовская функция* от предметных переменных кванторов всеобщности, которая замещает в матрице связанную квантором существования предметную переменную.

**Алгоритм приведения формулы к виду ССФ:**

**Шаг 1:** представить формулу  $F$  в виде ПНФ, т. е.

$F = \mathcal{R}_{x_1} \mathcal{R}_{x_2} \dots \mathcal{R}_{x_n} (M)$ , где  $\mathcal{R}_i \in \{\forall, \exists\}$ , а

$M=D_1 \& D_2 \& D_3 \& \dots,$

**Шаг 2:** найти в префиксе самый левый квантор существования и заменить его по правилу:

а) «если квантор существования находится на первом месте префикса, то вместо переменной, связанной этим квантором, подставить в матрице всюду предметную постоянную  $a$ , отличную от встречающихся постоянных, а квантор существования удалить»,

б) «если квантор существования находится на  $i$ -м месте префикса, т.е.  $\forall_{x_1} \forall_{x_2} \dots \forall_{x_{i-1}} \exists_{x_i} \dots$ , то выбрать  $(i-1)$ -местную функцию  $f(x_1, x_2, \dots, x_{i-1})$ , отличную от функций матрицы  $M$  и выполнить замену предметной переменной  $x_i$ , связанной квантором существования, на функцию  $f(x_1, x_2, \dots, x_{i-1})$  и квантор существования из префикса удалить».

**Шаг 3:** найти в префиксе следующий слева квантор существования и перейти к исполнению шага 2, иначе конец.

Формулу ПНФ, полученную в результате введения сколемовских функций называют **сколемовской стандартной формой (ССФ)**.

Преобразованная таким образом матрица может быть допущена к анализу истинности суждения по принципу резолюции.

**Пример 1.69.** Дано  $F = \exists_{x_1} \forall_{x_2} \forall_{x_3} \exists_{x_4} \forall_{x_5} \exists_{x_6} ((P_1(x_1, x_2) \vee \neg P_2(x_3, x_4, x_5)) \& \& P_3(x_4, x_6))$ . Преобразовать формулу к виду ССФ.

• Принять  $x_1 = a$  и удалить  $\exists_{x_1}$ :

$F = \forall_{x_2} \forall_{x_3} \exists_{x_4} \forall_{x_5} \exists_{x_6} ((P_1(a, x_2) \vee \neg P_2(x_3, x_4, x_5)) \& P_3(x_4, x_6))$ ,

• принять  $x_4 = f_1(x_2, x_3)$  и удалить  $\exists_{x_4}$ :

$$F = \forall_{x_2} \forall_{x_3} \forall_{x_5} \exists_{x_6} ((P_1(a, x_2) \vee \neg P_2(x_3, f_1(x_2, x_3), x_5)) \& P_3(f_1(x_2, x_3), x_6)),$$

- принять  $x_6 = f_2(x_2, x_3, x_5)$  и удалить  $\exists_{x_6}$ :

$$F = \forall_{x_2} \forall_{x_3} \forall_{x_5} ((P_1(a, x_2) \vee \neg P_2(x_3, f_1(x_2, x_3), x_5)) \& P_3(f_1(x_2, x_3), f_2(x_2, x_3, x_5))).$$

Множество дизъюнктов матрицы  $K = \{(P_1(a, x_2) \vee \neg P_2(x_3, f_1(x_2, x_3), x_5)), P_3(f_1(x_2, x_3), f_2(x_2, x_3, x_5))\}$ .

Пример 1.70.

Дано  $F = \exists_z \forall_w \exists_x \forall_y ((P_1$

$$z) \vee P_2(x) \vee P_3(y)) \& (\neg P_2(w) \vee$$

$\vee P_2(x) \vee P_3(y)) \& (\neg P_2(w) \vee \neg P_1(z) \vee P_3(y)))$ . Преобразовать формулу к виду ССФ.

- Принять  $z = a$  и удалить  $\exists_z$ :

$$F = \forall_w \exists_x \forall_y ((P_1(a) \vee P_2(x) \vee P_3(y)) \& (\neg P_2(w) \vee P_2(x) \vee P_3(y)) \& (\neg P_2(w) \vee \neg P_1(a) \vee P_3(y))),$$

- принять  $x = f(w)$  и удалить квантор  $\exists_x$ :

$$F = \forall_w \forall_y ((P_1(a) \vee P_2(f(w)) \vee P_3(y)) \& (\neg P_2(w) \vee P_2(f(w)) \vee P_3(y)) \& (\neg P_2(w) \vee \neg P_1(a) \vee P_3(y))).$$

Множество дизъюнктов матрицы  $K = \{(P_1(a) \vee P_2(f(w)) \vee P_3(y)), (\neg P_2(w) \vee P_2(f(w)) \vee P_3(y)), (\neg P_2(w) \vee \neg P_1(a) \vee P_3(y))\}$ .

Пример 1.71.

Дано  $F = \exists_z \forall_w \exists_x \forall_y ((P_1$

$$z) \vee P_2(x) \vee P_3(y)) \& (\neg P_2(w) \vee$$

$\vee P_2(x) \vee P_3(y)) \& (\neg P_2(w) \vee \neg P_1(z) \vee P_3(y)))$ . Преобразовать формулу к виду ССФ.

- Принять  $z = a$  и удалить  $\exists_z$ :

$$F = \forall_w \exists_x \forall_y ((P_1(a) \vee P_2(x) \vee P_3(y)) \& (\neg P_2(w) \vee P_2(x) \vee P_3(y)) \& (\neg P_2(w) \vee \neg P_1(a) \vee P_3(y))),$$

- принять  $x = f(w)$  и удалить  $\exists x$ :

$$F = \forall_w \forall_y ((P_1(a) \vee P_2(f(w)) \vee P_3(y)) \& (\neg P_2(w) \vee P_2(f(w)) \vee P_3(y)) \& (\neg P_2(w) \vee \neg P_1(a) \vee P_3(y))).$$

Множество дизъюнктов матрицы  $K = \{(P_1(a) \vee P_2(f(w)) \vee P_3(y)), (\neg P_2(w) \vee P_2(f(w)) \vee P_3(y)), (\neg P_2(w) \vee \neg P_1(a) \vee P_3(y))\}$ .

Обратите внимание, что в одном из дизъюнктов есть контрарные атомы, но один из них содержит предметную переменную  $w$ , а другой – сколемовскую функцию  $f(w)$ . Это -  $\neg P_2(w)$  и  $P_2(f(w))$ . В двух других дизъюнктах также есть контрарные атомы, но содержащие предметную постоянную 'а'. Это -  $P_1(a)$  и  $\neg P_1(a)$ . При исполнении алгоритма по принципу резолюции выполняется поиск и унификация таких атомов.

### 1.2.2. Исчисление предикатов

Если выполнить подстановку вместо пропозициональных переменных исчисления высказываний формулы алгебры предикатов, то каждая схема доказательства теоремы и каждая схема вывода заключения сохраняются в исчислении предикатов. Однако для кванторных операций следует дать интерпретацию формул логики предикатов и выделить правила и аксиомы, позволяющие управлять кванторами и выполнять унификацию предикатов.

#### 1.2.2.1. Интерпретация формул

Под *интерпретацией формул* следует понимать систему, состоящую из непустого множества  $V$ , называемого *универсумом*, и отображения предиката  $P(t_1, t_2, \dots, t_n)$  на

двухэлементное множество  $\{и, л\}$ . Предметные постоянные  $t_i$  и функциональные символы  $f_i(t_1, t_2, \dots)$  есть элементы  $t \in V$ , а предметные переменные есть обычные переменные, заданные на универсуме. Символы логических и кванторных операций приобретают также обычный смысл. Тогда каждому предикату  $P(t_1, t_2, \dots, t_n)$ , принявшему значение «и» или «л», ставится в соответствие  $n$ -местное отношение на множестве  $V$ .

Например, если  $V$  - множество целых чисел и дано частное суждение  $\exists_x \exists_y \exists_z (P(x, f_+(y, z))) :=$  «существует число  $x$ , которое меньше суммы чисел  $y$  и  $z$ », то при  $x=5, y=1, z=3$  имеем двухместную операцию  $f_+(1, 3)$  и двухместное отношение между числом 5 и значением операции  $f_+(1,3)=4$ . Отображение  $P(5,4)$  на двухэлементное множество дает значение «л». Это же суждение при  $x=5, y=2, z=4$  имеет  $f_+(2,4)=6$  и  $P(5, 6)=$ «и» (см. рис. 1.10.)

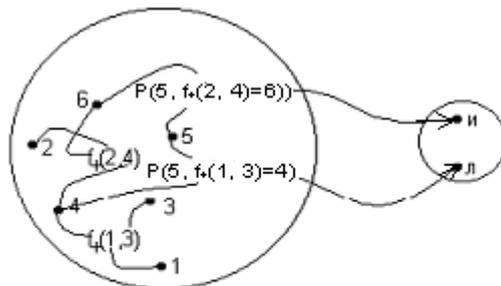


Рис. 1.10. Интерпретация формулы  
 $\exists_x \exists_y \exists_z (P(x, f_+(y, z))) :=$  « $x$  меньше  $(y+z)$ »  
 для  $x=5, y=1$  и  $z=3$  или  $x=5, y=2, z=4$ .

Другими словами, интерпретация функциональных символов определяется по значениям функции на универсуме, а интерпретация предикатных символов по отображению на множество  $\{и, л\}$ .

Выделяют три класса формул: тождественно истинные, тождественно ложные и выводимые.

**Тождественно истинные формулы** - это особый класс формул, которые при выполнении логических и кванторных операций принимают значение «истины» для всех интерпретаций предметных постоянных, функциональных и предикатных символов. Большинство из них - аксиомы исчисления предикатов.

Например, формула  $\exists x(F(x)) \leftrightarrow \neg \forall x(\neg F(x))$  := «существуют  $x$ , для которых  $F(x)=и$ , эквивалентна формуле не для всех  $x$   $F(x)=л$ ». Это - тождественно истинная формула.

**Тождественно ложные формулы** - это особый класс формул исчисления предикатов, которые при выполнении логических и кванторных операций принимают значение «ложь» для всех интерпретаций предметных постоянных, функциональных и предикатных символов.

Например, формула  $\exists x(F(x)) \& \forall x(\neg F(x))$  := «существуют  $x$ , для которых  $F(x)=и$ , и для всех  $x$   $F(x)=л$ » является тождественно ложной.

**Выводимые формулы** - это особый класс формул исчисления предикатов, которые при выполнении логических и кванторных операций принимают значение «истина» не для всех интерпретаций предметных постоянных, функциональных и предикатных символов.

Например, формула  $\exists x(F(x)) \rightarrow \neg \forall x(F(x))$  := «если существует  $x$ , для которого  $F(x)=и$ , то не для всех  $x$   $F(x)=и$ ».

Вывод заключения из множества посылок есть:

$$F_1, F_2, \dots, F_n \vdash B,$$

где слева от знака “ $\vdash$ ” записывают множество посылок и необходимых аксиом  $F_1, F_2, \dots, F_n$ , а справа – заключение  $B$ . На естественном языке эта запись читается так: «верно, что  $B$  выводимо из посылок и аксиом  $F_1, F_2, \dots, F_n$ ».

Другая форма вывода заключения:

$$\frac{F_1, F_2, \dots, F_n}{B},$$

где над чертой - множество посылок и аксиом  $F_1, F_2, \dots, F_n$ , а под чертой - заключение  $B$ .

Вывод заключения на языке математической логики это доказательство теоремы  $\vdash F_1 \& F_2 \& \dots \& F_n \rightarrow B$ . В процессе доказательства последовательно используют аксиомы и законы логики предикатов, правила введения и удаления кванторов, правила подстановки и унификации предикатов. При выводе используют правила *modus ponens* и *modus tollens*.

#### 1.2.2.2. Аксиомы исчисления предикатов

Аксиомы исчисления предикатов опираются на двенадцать аксиом исчисления высказываний, сохраняя при этом особенности исполнения кванторных операций. Поэтому для понимания механизма вывода в исчислении предикатов следует знать правила введения и удаления кванторов. Эти правила существенно облегчают преобразования сложных логических формул.

П1. Удаление квантора всеобщности: если выводима формула  $\forall x(F(x))$ , то, заменив предметную переменную  $x$  на терм  $t$ , можно удалить квантор всеобщности и получить выводимую формулу

$$\frac{\forall_x(F(x))}{F(t)}$$

П2. Введение квантора всеобщности:

а) если выводима формула  $F(t)$ , то, заменив терм  $t$  на предметную переменную  $x$ , можно ввести квантор всеобщности и получить выводимую формулу

$$\frac{F(t)}{\forall_x F(x)}$$

б) если выводима формула  $(F_1(t) \rightarrow F_2(x))$  и  $F_1(t)$  не содержит свободной переменной  $x$ , то выводима формула

$$\frac{F_1(t) \rightarrow F_2(x)}{F_1(t) \rightarrow \forall_x(F_2(x))}$$

П3. Удаление квантора существования: если выводима формула  $\exists_x(F(x))$ , то, заменив предметную переменную  $x$  на предметную постоянную 'а', можно удалить квантор существования и получить выводимую формулу

$$\frac{\exists_x(F(x))}{F(a)}$$

П4. Введение квантора существования:

а) если выводима формула  $F(t)$ , то, заменив терм  $t$  на предметную переменную  $x$  в заданной области интерпретации, можно ввести квантор существования и получить выводимую формулу

$$\frac{F(t)}{\exists_x F(x)}$$

б) если выводима формула  $F_1(x) \rightarrow F_2(t)$  и  $F_2$  не содержит свободной переменной  $x$ , то выводима формула

$$\frac{F_1(x) \rightarrow F_2(t)}{\exists_x(F_1(x)) \rightarrow F_2(t)}$$

П5. Формирование ПНФ формулы:

а) если при выполнении логических операций один из предикатов  $F_i$  не содержит переменной  $x$ , связанной в

другом предикате  $F_j$ , и формулы выводимы, то выводима формула:

$$\frac{\mathfrak{R}_x(F_1(x)) \vee F_2(t)}{\mathfrak{R}_x(F_1(x) \vee F_2(t))}, \quad \frac{\mathfrak{R}_x(F_1(x)) \& F_2(t)}{\mathfrak{R}_x(F_1(x) \& F_2(t))}, \quad \frac{F_1(t) \rightarrow \mathfrak{R}_x(F_2(x))}{\mathfrak{R}_x(F_1(t) \rightarrow F_2(x))}, \quad \text{где } \mathfrak{R}_x \in \{\forall_x, \exists_x\}.$$

б) если выводимы формулы  $\forall_x(F_1(x)) \vee \forall_x(F_2(x))$  и  $\exists_x(F_1(x)) \& \exists_x(F_2(x))$ , то при смене в левой формуле имени переменной получим также выводимые формулы:

$$\frac{\forall_x(F_1(x)) \vee \forall_x(F_2(x))}{\forall_y \forall_x(F_1(y) \vee F_2(x))}, \quad \frac{\exists_x(F_1(x)) \& \exists_x(F_2(x))}{\exists_y \exists_x(F_1(y) \vee F_2(x))}.$$

### 1.2.2.3. Правила унификации предикатов

Если матрица ССФ не содержит сколемовских функций, то для вывода заключения применим принцип резолюции исчисления высказываний. Однако, если аргументы атомов дизъюнктов ССФ содержат сколемовские функции, то для поиска контрарных атомов необходимо выполнять унификацию самих дизъюнктов. Множество подстановок при унификации дизъюнктов нужно выполнять последовательно, просматривая каждый раз только одну предметную переменную.

Например, если даны два дизъюнкта  $(P_1(x) \vee \neg P_2(x))$  и  $(\neg P_1(f(x)) \vee P_3(y))$ , то для получения контрарной пары атомов необходима подстановка:

$$\frac{\int_x^{f(x)} (P_1(x) \vee \neg P_2(x))}{(P_1(f(x)) \vee \neg P_2(f(x)))}.$$

В результате соединения дизъюнктов логической связкой « $\vee$ » может быть получена резольвента:  $(P_1(f(x)) \vee \neg P_2(f(x))) \vee (\neg P_1(f(x)) \vee P_3(y)) = (\neg P_2(f(x)) \vee P_3(y))$ .

Если пара дизъюнктов имеет вид  $(P_1(f_1(x)) \vee \neg P_2(x))$  и

$(\neg P_1(f_2(x)) \vee P_3(y))$ , то никакая подстановка не позволит вычислить контрарные атомы.

Пример 1.72. Даны  $P_3(a, x, f(q(y)))$  и  $\neg P_3(z, f(z), f(u))$ .  
Выполнить унификацию контрарных атомов.

$$\frac{\int_z^a \neg P_3(z, f(z), f(u))}{\neg P_3(a, f(a), f(u))}, \quad \frac{\int_x^{f(a)} P_3(a, x, f(q(y)))}{P_3(a, f(a), f(q(y)))}, \quad \frac{\int_u^{q(y)} \neg P_3(a, f(a), f(u))}{\neg P_3(a, f(a), f(q(y)))}.$$

В результате получены два контрарных атома  $P_3(a, f(a), f(q(y)))$  и  $\neg P_3(a, f(a), f(q(y)))$ .

Пример 1.73. Даны  $P_3(x, a, f(q(a)))$  и  $\neg P_3(z, y, f(u))$ .  
Выполнить унификацию контрарных атомов.

$$\frac{\int_x^b P_3(x, a, f(q(a)))}{P_3(b, a, f(q(a)))}, \quad \frac{\int_y^a \neg P_3(z, y, f(u))}{\neg P_3(z, a, f(u))}, \quad \frac{\int_z^b \neg P_3(z, a, f(u))}{\neg P_3(b, a, f(u))}, \quad \frac{\int_u^{q(a)} \neg P_3(b, a, f(u))}{\neg P_3(b, a, f(q(a)))}.$$

В результате получены две контрарных формулы:

$$P_3(b, a, f(q(a))) \text{ и } \neg P_3(b, a, f(q(a))).$$

Обратите внимание, всюду выполнялась замена только предметной переменной на постоянную или функциональный символ.

#### 1.2.2.4. Метод дедуктивного вывода

В логике предикатов вывод выполняется также, как в исчислении высказываний. Все правила логики высказываний (ИВ) включены в множество правил логики предикатов (ИП).

**Пример 1.74.** Доказать истинность заключения

$$\frac{\forall_x(P_1(x) \rightarrow \neg P_2(x)), \forall_x(P_3(x) \rightarrow P_1(x))}{\forall_x(P_3(x) \rightarrow \neg P_2(x))}$$

Доказательство:

- $F_1 = \forall_x(P_1(x) \rightarrow \neg P_2(x))$  – посылка,
- $F_2 = \forall_x(P_3(x) \rightarrow P_1(x))$  – посылка,
- $F_3 = (P_1(t) \rightarrow \neg P_2(t))$  - заключение по  $F_1$  и правилу П1 и.п.,
- $F_4 = (P_3(t) \rightarrow P_1(t))$  - заключение по  $F_2$  и правилу П1 и.п.,
- $F_5 = (P_3(t) \rightarrow (\neg P_2(t)))$  - заключение по  $F_3, F_4$  и правилу П9,
- $F_6 = \forall_x(P_3(x) \rightarrow (\neg P_2(x)))$  - заключение по  $F_5$  и правилу П2, *ч.т.д.*

На рис. 1.11. изображен граф вывода  $\forall_x(P_3(x) \rightarrow \neg P_2(x))$ .

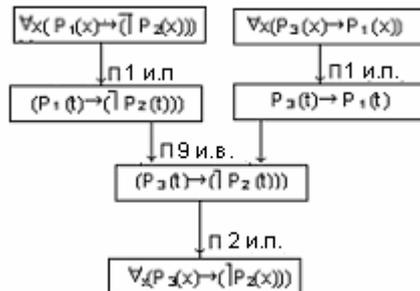


Рис.1.11. Граф дедуктивного вывода.

**Пример 1.75.** Доказать истинность заключения

$$\frac{\forall_x(\exists_y(P_1(x, y) \& P_2(y)) \rightarrow \exists_y(P_3(y) \& P_4(x, y))), \neg \exists_x(P_3(x))}{\neg \exists_x(P_3(x)) \rightarrow \forall_x \forall_y (P_1(x, y) \rightarrow \neg P_2(y))}$$

Доказательство:

- $F_1 = \forall_x(\exists_y(P_1(x, y) \& P_2(y)) \rightarrow \exists_y((P_3(y) \& P_4(x, y))))$  - посылка,
- $F_2 = \neg \exists_x(P_3(x))$  - посылка,

- $F_3 = \forall x(\neg P_3(x))$  - заключение по  $F_2$  и закону отрицания формулы,
- $F_4 = \neg P_3(t)$  - заключение по  $F_3$  и правилу П1 и.п.,
- $F_5 = \neg P_3(t) \vee \neg P_4(x, t)$  - заключение по  $F_4$  и правилу ПЗ и.в.,
- $F_6 = \forall y(\neg P_3(y) \vee (\neg P_4(x, y)))$  - заключение по  $F_5$  и правилу П2 и.п.,
- $F_7 = \neg \exists y(P_3(y) \& P_4(x, y))$  - заключение по  $F_6$ ,
- $F_8 = \exists y(P_1(t, y) \& P_2(y)) \rightarrow \exists y(P_3(y) \& P_4(t, y))$  - заключение по  $F_1$  и правилу П1 и.п.,
- $F_9 = \neg \exists y(P_1(t, y) \& P_2(y))$  - заключение по  $F_7$  и  $F_8$  и правилу м. т.,
- $F_{10} = \forall y(\neg P_1(t, y) \vee \neg P_2(y))$  - заключение по  $F_9$ ,
- $F_{11} = \forall y(P_1(t, y) \rightarrow \neg(P_2(y)))$  - заключение по  $F_{10}$ ,
- $F_{12} = \forall x \forall y(P_1(x, y) \rightarrow \neg(P_2(y)))$  - заключение по  $F_{11}$  и правилу П2 и.п.,
- $F_{13} = \neg \exists x(P_3(x)) \rightarrow \forall x \forall y(P_1(x, y) \rightarrow (\neg P_2(y)))$  - заключение по  $F_2$  и  $F_{12}$ .

*ч.т.д.*

На рис. 1.12. изображен граф вывода  $\neg \exists x(P_3(x)) \rightarrow \forall x \forall y(P_1(x, y) \rightarrow (\neg P_2(y)))$ .

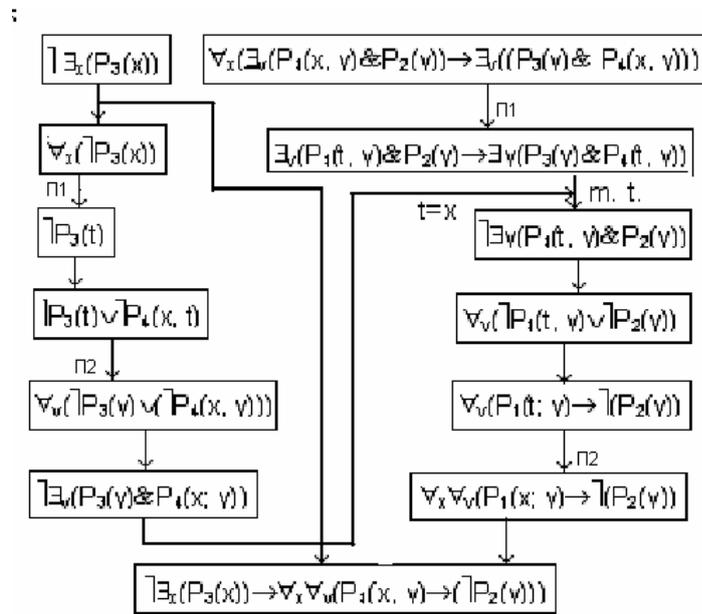


Рис. 1.12. Граф дедуктивного вывода.

Пример 1.76. Доказать истинность заключения  
 $\frac{\exists_x(P_1(x) \& \forall_y(P_2(y) \rightarrow P_4(x, y))), \forall_x(P_1(x) \rightarrow \forall_y(P_3(y) \rightarrow \neg P_4(x, y)))}{\forall_x(P_2(x) \rightarrow \neg P_3(x))}$

Доказательство:

- $F_1 = \exists_x(P_1(x) \& \forall_y(P_2(y) \rightarrow P_4(x, y)))$  – посылка,
- $F_2 = \forall_x(P_1(x) \rightarrow \forall_y(P_3(y) \rightarrow \neg P_4(x, y)))$  - посылка,
- $F_3 = P_1(a) \& \forall_y(P_2(y) \rightarrow P_4(a, y))$  - заключение по  $F_1$ ,
- $F_4 = P_1(a)$  - заключение по  $F_3$  и правилу П2 и.в.,
- $F_5 = \forall_y(P_2(y) \rightarrow P_4(a, y))$  - заключение по  $F_3$  и правилу П2 и.в.,
- $F_6 = P_2(t) \rightarrow P_4(a, t)$  - заключение по  $F_5$  и правилу П1 и.п.,
- $F_7 = P_1(t) \rightarrow \forall_y(P_3(y) \rightarrow \neg P_4(t, y))$  - заключение по  $F_2$  и правилу П1 и.п.,

- $F_8 = \forall y(P_3(y) \rightarrow \neg P_4(a, y))$  - заключение по  $F_3$  и  $F_7$  при  $t=a$  и *m. p.*,
- $F_9 = P_3(t) \rightarrow \neg P_4(a, t)$  - заключение по  $F_8$  и правилу П1 и.п.,
- $F_{10} = P_4(a, t) \rightarrow \neg P_3(t)$  - заключение по  $F_9$  и правилу П6 и.в.,
- $F_{11} = P_2(t) \rightarrow \neg P_3(t)$  - заключение по  $F_6$  и  $F_{10}$  и правилу П9 и.в.,
- $F_{12} = \forall x(P_2(x) \rightarrow \neg P_3(x))$  - заключение по  $F_{11}$  и правилу П2 и.п. *ч.т.д.*

На рис. 1.13. изображен граф вывода  $\forall x(P_2(x) \rightarrow \neg P_3(x))$ .

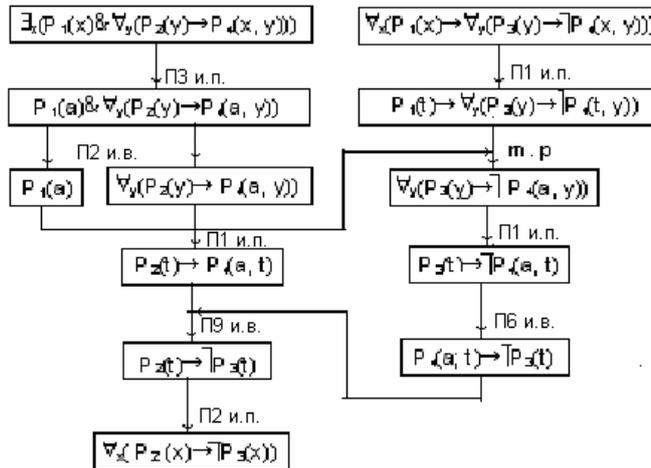


Рис.1.13. Граф дедуктивного вывода.

Пример 1.77. «Таможенные чиновники обыскивают каждого, кто въезжает в страну, кроме высокопоставленных лиц. Если некоторые люди способствуют провозу наркотиков, то на внутреннем рынке есть наркотик. Никто из высокопоставленных лиц не способствует провозу наркотиков. Следовательно, некоторые из таможенников способствуют провозу наркотиков?»[15].

Пусть даны предикаты  $P_1(x)$ : « $x$  - таможенный чиновник»,  $P_2(x,y)$ : « $x$  обыскивает  $y$ »,  $P_3(y)$ : « $y$  въезжает в страну»,  $P_4(y)$ : « $y$  - высокопоставленное лицо»,  $P_5(y)$ : « $y$  способствует провозу наркотиков». Тогда формальная запись суждения имеет вид:

$$\forall_y (P_3(y) \& \neg P_4(y) \rightarrow \forall_x (P_1(x) \& P_2(x,y))),$$

$$\exists_y (P_3(y) \& P_5(y)),$$

$$\forall_y (P_3(y) \& P_4(y) \rightarrow \neg P_5(y))$$

---


$$\exists_x (P_1(x) \& P_5(x)).$$

Доказательство:

- $F_1 = \exists_y (P_3(y) \& P_5(y))$  - посылка,
- $F_2 = P_3(a) \& P_5(a)$  - заключение по  $F_1$  и правилу ПЗ и.п.,
- $F_3 = P_3(a)$  - заключение по  $F_2$  и правилу П2 и.в.,
- $F_4 = P_5(a)$  - заключение по  $F_2$  и правилу П2 и.в.,
- $F_5 = \forall_y (P_3(y) \& P_4(y) \rightarrow \neg P_5(y))$  - посылка,
- $F_6 = P_3(t) \& P_4(t) \rightarrow \neg P_5(t)$  - заключение по  $F_5$  и правилу П1 и.п.,
- $F_7 = \neg P_3(t) \vee \neg P_4(t) \vee \neg P_5(t)$  - заключение по  $F_6$ ,
- $F_8 = \neg P_4(a)$  - заключение по  $F_7$  при  $t=a$ , когда  $\neg P_3(a) = \text{л}$  и  $\neg P_5(a) = \text{л}$ ,
- $F_9 = \forall_y (P_3(y) \& \neg P_4(y) \rightarrow \forall_x (P_1(x) \& P_2(x,y)))$  - посылка,
- $F_{10} = \forall_y \forall_x (P_3(y) \& \neg P_4(y) \rightarrow (P_1(x) \& P_2(x,y)))$  - заключение по  $F_9$  и правилу П5 и.п.,
- $F_{11} = P_3(a) \& \neg P_4(a) \rightarrow P_1(t) \& P_2(t,a)$  - заключение по  $F_{10}$  при  $y=a$  и правилу П1 и.п.,
- $F_{12} = P_3(a) \& \neg P_4(a)$  - заключение по  $F_3$  и  $F_8$  и правилу П1 и.в.,
- $F_{13} = (P_1(t) \& P_2(t,a))$  - заключение по  $F_{11}$  и  $F_{12}$  и правилу  $m.p.$ ,
- $F_{14} = P_1(t)$  - заключение по  $F_{13}$  и правилу П2 и.в.,

- $F_{15} = P_1(t) \& P_5(t)$  - заключение по  $F_4, F_{14}$  при  $t=a$  и правилу П1 и.в.,
- $F_{16} = \exists x(P_1(x) \& P_5(x))$  - заключение по  $F_{15}$  и правилу П3 и.п. ч.т.д.

На рис. 1.14. изображен граф вывода  $\exists x(P_1(x) \& P_5(x))$

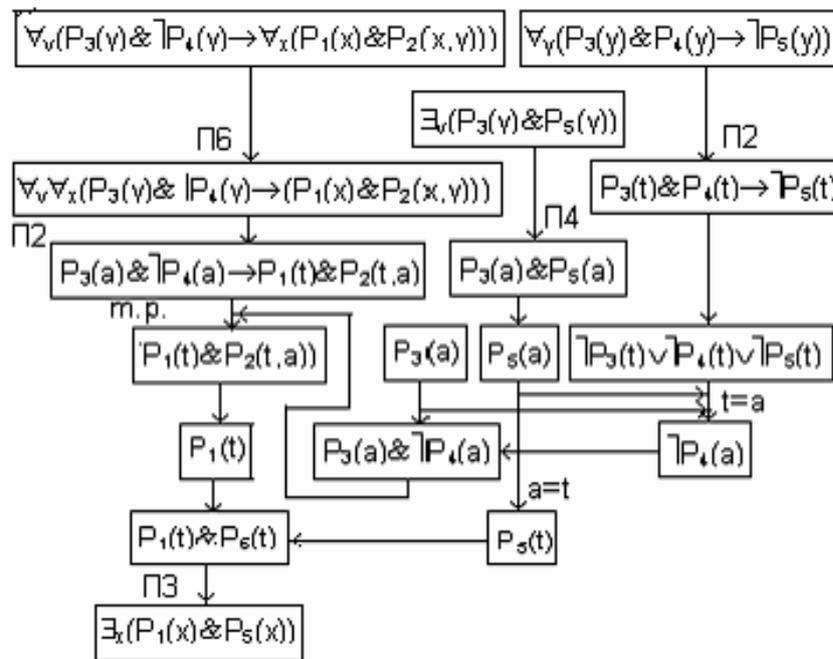


Рис. 1.14. Граф дедуктивного вывода  $\exists x(P_1(x) \& P_5(x))$ .

### 1.2.2.5. Метод резолюции

Если матрица формулы в результате приведения ее к виду ПНФ не будет содержать сколемовских функций, то для вывода заключения полностью применим алгоритм метода резолюции исчисления высказываний. Однако, ес-

ли в результате приведения формулы к виду ССФ аргументы атомов будут содержать сколемовские функции, то для поиска контрарных атомов необходимо выполнить подстановку вместо предметных переменных аргумента атома новых предметных переменных или постоянных или функциональных символов, т.е. термов, и получить новые формулы дизъюнктов, которые уже возможно допускают унификацию контрарных пар.

Множество подстановок нужно выполнять последовательно, просматривая каждый раз только одну предметную переменную.

Принцип резолюции логики предикатов может быть усилен использованием информации о резольвируемых атомах.

*Упорядоченным дизъюнктом* называется дизъюнкт с заданной последовательностью атомов. Атом  $P_j$  старше атома  $P_i$  в упорядоченном дизъюнкте тогда и только тогда, когда  $j > i$ . Например, в упорядоченном дизъюнкте  $(P_1 \vee P_2 \vee P_3 \vee P_4)$  самым младшим считается атом  $P_1$ , а самым старшим -  $P_4$ . При наличии в упорядоченном дизъюнкте двух одинаковых атомов, по закону идемпотенции, следует удалить старший атом.

Например,

$$(P_1(x) \vee P_2(y) \vee P_1(x) \vee P_3(z)) \equiv (P_1(x) \vee P_2(y) \vee P_3(z)).$$

При соединении дизъюнктов, содержащих контрарные атомы, и формировании резольвенты нужно удалить из пары старший контрарный атом, а младший сохранить в резольвенте, выделив его рамкой. Например,

$$(P_1(x) \vee P_2(y)) \vee (\neg P_1(x) \vee P_3(z)) \equiv (\boxed{P_1(x)} \vee P_2(y) \vee P_3(z)).$$

При этом следует соблюдать правила:

а) если в резольvente за обрамленным атомом справа не следует никакой другой атом, то обрамленный атом удаляют ,

б) если в резольvente за обрамленным атомом справа следует какой-либо атом без рамки, то обрамленный атом оставить для последующего анализа,

с) если резольвента представляет собой один обрамленный атом и он унифицируется с контрарным одноатомным дизъюнктом, то формируется пустая резольвента, что свидетельствует об истинности выводимого заключения. \*

Пример 1.78. «Существуют студенты, которые любят всех преподавателей. Ни один из студентов не любит невежд. Следовательно, ни один преподаватель не является невеждой»[3].

Введем предикаты:  $P_1(x)$ := «x – студент»,  $P_2(y)$ := «y – преподаватель»,  $P_3(x, y)$ := «x любит y»,  $P_4(y)$ := «y – невежда».

Формула этого суждение имеют вид:

$$\frac{\exists_x (P_1(x) \& \forall_y (P_2(y) \rightarrow P_3(x, y))), \forall_x (P_1(x) \rightarrow \forall_y (P_4(y) \rightarrow \neg P_3(x, y)))}{\forall_y (P_2(y) \rightarrow \neg P_4(y))}.$$

Вывод заключения:

- преобразовать посылки и отрицание заключения к виду

ССФ:

$$F_1 = \exists_x (P_1(x) \& \forall_y (P_2(y) \rightarrow P_3(x, y))) = \forall_y (P_1(a) \& (\neg P_2(y) \vee P_3(a, y))),$$

$$F_2 = \forall_x (P_1(x) \rightarrow \forall_y (P_4(y) \rightarrow \neg P_3(x, y))) = \forall_x \forall_y (\neg P_1(x) \vee \neg P_4(y) \vee \neg P_3(x, y)),$$

---

\* по материалам [3].

$$F_3 = \neg \forall y (P_2(y) \rightarrow \neg P_4(y)) = P_2(b) \& P_4(b),$$

• выписать множество дизъюнктов посылок и отрицания заключения:  $K = \{P_1(a), (\neg P_2(y) \vee P_3(a, y)), (\neg P_1(x) \vee \neg P_4(y) \vee \neg P_3(x, y)), P_2(b), P_4(b)\}$ .

Процесс вывода заключения по принципу резолюции представлен графом на рис. 1.15.

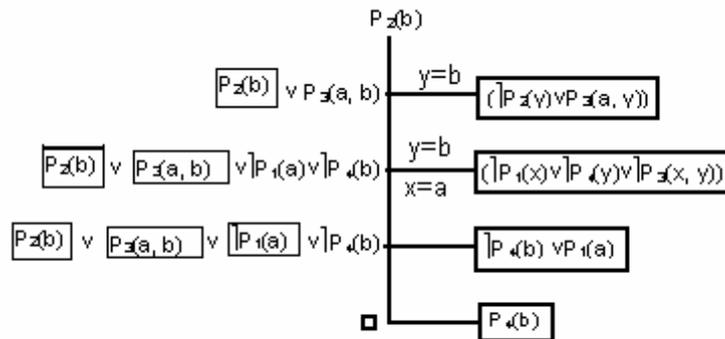


Рис. 1.15. Граф вывода по принципу резолюции.

• Выполнить подстановку и унификацию дизъюнктов:

$$P_2(b) \vee \int_y^b (P_2(y) \vee P_3(a, y)) = P_2(b) \vee \cancel{P_2(b)} \vee P_3(a, b) =$$

$$\boxed{P_2(b)} \vee P_3(a, b) - \text{резольвента,}$$

$$\boxed{P_2(b)} \vee P_3(a, b) \vee \int_y^b (\neg P_1(x) \vee \neg P_4(y) \vee \neg P_3(x, y)) =$$

$$\boxed{P_2(b)} \vee P_3(a, b) \vee \neg P_1(x) \vee \neg P_4(b) \vee \neg P_3(x, b) =$$

$$\begin{aligned}
& \boxed{P_2(b)} \vee P_3(a,b) \vee \int_x^a (\neg P_1(x) \vee \neg P_4(b) \vee \neg P_3(x,b)) = \\
& \boxed{P_2(b)} \vee P_3(a,b) \vee \neg P_1(a) \vee \neg P_4(b) \vee \cancel{\neg P_3(a,b)} = \\
& \boxed{P_2(b)} \vee \boxed{P_3(a,b)} \vee \neg P_1(a) \vee \neg P_4(b) \text{—резольвента,} \\
& \boxed{P_2(b)} \vee \boxed{P_3(a,b)} \vee \neg P_1(a) \vee \neg P_4(b) \vee \cancel{P_1(a)} = \\
& \boxed{P_2(b)} \vee \boxed{P_3(a,b)} \vee \boxed{\neg P_1(a)} \vee \neg P_4(b) \text{—резольвента,} \\
& \boxed{P_2(b)} \vee \boxed{P_3(a,b)} \vee \boxed{\neg P_1(a)} \vee \neg P_4(b) \vee \cancel{P_4(b)} = \\
& \boxed{P_2(b)} \vee \boxed{P_3(a,b)} \vee \boxed{\neg P_1(a)} \vee \boxed{\neg P_4(b)} \equiv 1.
\end{aligned}$$

**Пример 1.79.** «Преподаватели принимали зачеты у всех студентов, не являющихся отличниками. Некоторые аспиранты и студенты сдавали зачеты только аспирантам. Ни один из аспирантов не был отличником. Следовательно, некоторые преподаватели были аспирантами» [3].

Пусть  $P_1(x)$  := « $x$  – отличник»,  $P_2(x)$  := « $x$  – аспирант»,  $P_3(x)$  := « $x$  – студент»,  $P_4(x, y)$  := « $x$  сдает зачет  $y$ »,  $P_5(x)$  := « $x$  – преподаватель».

Формулы этого суждения имеют вид:

$$\begin{array}{l}
\forall_x (P_3(x) \& \neg P_1(x) \rightarrow \exists_y (P_5(y) \& P_4(x, y))), \\
\exists_x (P_2(x) \& P_3(x) \& \forall_y (P_4(x, y) \rightarrow P_2(y)), \\
\forall_x (P_2(x) \rightarrow \neg P_1(x)). \\
\hline
\exists_x (P_5(x) \& P_2(x)).
\end{array}$$

Вывод заключения (на рис. 1.16 изображен граф вывода):

- преобразовать посылки и отрицание заключения в ССФ:

$$F_1 = \forall_x (P_3(x) \& \neg P_1(x) \rightarrow \exists_y (P_5(y) \& P_4(x, y))) = \forall_x (\neg P_3(x) \vee P_1(x) \vee P_5(f(x))) \&$$

$$\& (\neg P_3(x) \vee P_1(x) \vee P_4(x, f(x))),$$

$$F_2 = \exists_x (P_2(x) \& P_3(x) \& \forall_y (P_4(x, y) \rightarrow P_2(y))) = \forall_y (P_2(a) \& P_3(a) \& \& (\neg P_4(a, y) \vee P_2(y))),$$

$$F_3 = \forall_x (P_2(x) \rightarrow \neg P_1(x)) = \forall_x (\neg P_2(x) \vee \neg P_1(x)),$$

$$F_4 = \neg \exists_x (P_5(x) \& P_2(x)) = \forall_x (\neg P_5(x) \vee \neg P_2(x)),$$

- ВЫПИСАТЬ МНОЖЕСТВО

$$\text{ДИЗЬЮНКТОВ: } K = \{(\neg P_3(x) \vee P_1(x) \vee P_5(f(x))), (\neg P_3(x) \vee P_1(x) \vee P_4(x, f(x))), P_2(a), P_3(a), (\neg P_4(a, y) \vee P_2(y)), (\neg P_1(x) \vee \neg P_2(x)), (\neg P_5(x) \vee \neg P_2(x))\},$$

- ВЫПОЛНИТЬ ПОДСТАНОВКУ И УНИФИКАЦИЮ ДИЗЬЮНКТОВ:

$$\int_x^a (\neg P_1(x) \vee \neg P_2(x) \vee P_2(x)) = \neg P_1(a) \vee \boxed{\neg P_2(a)} \vee \cancel{P_2(a)} = \neg P_1(a) -$$

-резольвента,

$$\neg P_1(a) \vee \int_x^a (\neg P_3(x) \vee \neg P_1(x) \vee P_4(x, f(x))) = \boxed{\neg P_1(a)} \vee \neg P_3(a) \vee$$

$$\vee \cancel{\neg P_1(a)} \vee P_4(a, f(a)) = \boxed{\neg P_1(a)} \vee \neg P_3(a) \vee P_4(a, f(a)) - \text{резольвента,}$$

$$\boxed{\neg P_1(a)} \vee \neg P_3(a) \vee P_4(a, f(a)) \vee \int_y^{f(a)} (\neg P_4(a, y) \vee P_2(y)) = \boxed{\neg P_1(a)} \vee$$

$$\vee \neg P_3(a) \vee \boxed{P_4(a, f(a))} \vee \cancel{\neg P_4(a, f(a))} \vee P_2(f(a)) = \boxed{\neg P_1(a)} \vee \neg P_3(a) \vee$$

$$\vee \boxed{P_4(a, f(a))} \vee P_2(f(a)) - \text{резольвента,}$$

$$\boxed{\neg P_1(a)} \vee \neg P_3(a) \vee \boxed{P_4(a, f(a))} \vee P_2(f(a)) \vee \int_x^{f(a)} (\neg P_5(x) \vee \neg P_2(x)) =$$

$$\boxed{\neg P_1(a)} \vee \neg P_3(a) \vee \boxed{P_4(a, f(a))} \vee \boxed{P_2(f(a))} \vee \neg P_5(f(a)) \vee \cancel{\neg P_2(f(a))} =$$

$$\boxed{\neg P_1(a)} \vee \neg P_3(a) \vee \boxed{P_4(a, f(a))} \vee \boxed{P_2(f(a))} \vee \neg P_5(f(a)) - \text{резольвента,}$$

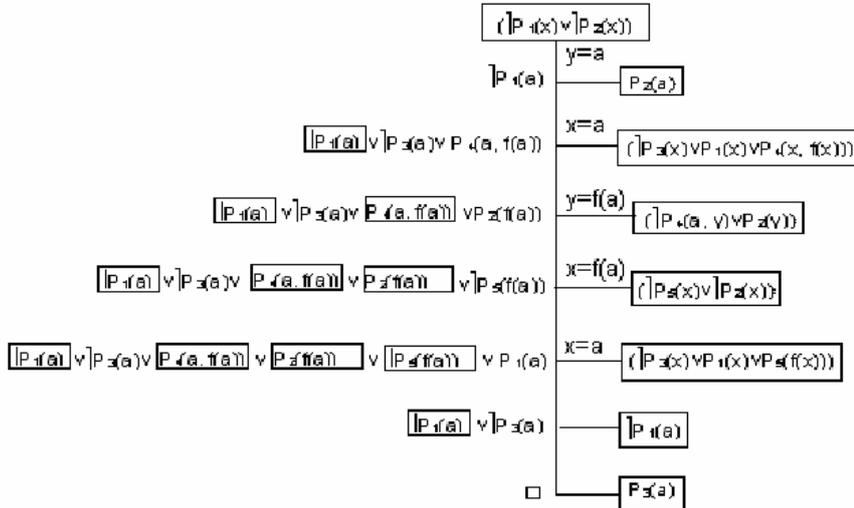


Рис. 1.16. Граф вывода заключения по принципу резолюции.

$$\begin{aligned} & \neg P_1(a) \vee \neg P_3(a) \vee P_4(a, f(a)) \vee P_2(f(a)) \vee \neg P_5(f(a)) \vee \\ & \vee \int_x (\neg P_3(x) \vee P_1(x) \vee P_5(f(x))) = \neg P_1(a) \vee \neg P_3(a) \vee P_4(a, f(a)) \vee \\ & \vee P_2(f(a)) \vee \neg P_5(f(a)) \vee \neg P_3(a) \vee P_1(a) \vee P_5(f(a)) = \neg P_1(a) \vee \neg P_3(a) \vee \\ & \vee P_4(a, f(a)) \vee P_2(f(a)) \vee \neg P_5(f(a)) \vee P_1(a) - \text{резольвента,} \\ & \neg P_1(a) \vee \neg P_3(a) \vee P_4(a, f(a)) \vee P_2(f(a)) \vee \neg P_5(f(a)) \vee P_1(a) \vee \\ & \vee \neg P_3(a) = \neg P_1(a) \vee \neg P_3(a) - \text{резольвента,} \\ & \neg P_1(a) \vee \neg P_3(a) \vee P_3(a) = \square. \end{aligned}$$

Приведенные примеры показывают эффективность вывода заключения по принципу резолюции, так как применяются только два правила: подстановки и унификации. Эти процедуры достаточно просто реализуются на компьютере, что позволило разработать формальные языки для логических задач.

### 1.2.3. Логическое программирование

Языки логического программирования не определяют жесткой последовательности действий, а допускают многовариантность решения задачи, используя правила подстановки термов и механизм унификации дизъюнктов. Такие языки называют *декларативными*. Типичным представителем языков логического программирования является Prolog. Этот язык появился в начале 70-х годов. Само название Prolog есть сокращение, означающее *программирование в терминах логики*.

#### 1.2.3.1. Основы логического программирования\*

В математической логике была доказана теорема  $\vdash F_1 \& F_2 \& \dots \& F_n \rightarrow B$ , где каждое  $F_i$  есть посылка или аксиома, а  $B$  – заключение. При истинности всех посылок и аксиом  $F_i$  эта теорема будет истинной только при  $B=i$ .

Для удобства программирования в языке Prolog принята *обратная импликация*, т.е.  $\vdash B \leftarrow F_1 \& F_2 \& \dots \& F_n$ . При этом оператор конъюнкции замещают знаком «&», а оператор импликации символом «:-», т.е.  $B :- F_1, F_2, \dots, F_n$ .

Поэтому синтаксическая конструкция теоремы есть

<заключение>«:-»<условие>{«&»<условие>}«.».

Такую форму записи на языке Prolog называют *правилом*.

Множество правил есть *интенциональная\*\** часть дедуктивной системы, а множество посылок или фактов, формирующих условия для множества правил, – её *экстенциональная\*\*\** часть. Левую часть правила называют

\* по материалам [1].

\*\* лат. intensio – усиление организации вывода множеством правил.

\*\*\* лат. extensus – расширяющий вывод на множестве посылок и фактов.

*заголовком*, а правую – *телом*. Предметные переменные в правилах начинаются с прописной буквы, а предметные постоянные – со строчной. Множество правил представляет механизм достижения цели при заданных условиях.

Если заключение В содержит несколько предметных переменных  $X_1, X_2, \dots, X_n$  то на языке математической логики это есть указание приписать в теореме кванторы всеобщности по каждой переменной, т.е.  $\forall_{x_1} \forall_{x_2} \dots \forall_{x_r} (B(X_1, X_2, \dots, X_r))$ .

При формировании условий на языке математической логики это - указание приписать в теореме кванторы существования по каждой переменной, т.е.  $\exists_{x_1} \exists_{x_2} \dots \exists_{x_r} (F_i(X_1, X_2, \dots, X_r))$ .

В этом случае теорема имеет вид:

$$\begin{aligned} & \vdash \forall_{x_1} \forall_{x_2} \dots \forall_{x_r} (B(X_1, X_2, \dots, X_r)) :- \\ & \quad \exists_{x_1} \exists_{x_2} \dots \exists_{x_r} (F_1(X_1, X_2, \dots, X_r), F_2(X_1, X_2, \dots, X_r), \dots, F_n(X_1, X_2, \dots, X_r)). \end{aligned}$$

Истинность правила определяется условием «если истинны посылки и аксиомы, то истинно и заключение».

Если правая часть правила пустая, то левая часть, т.е.  $\forall_{x_1} \forall_{x_2} \dots \forall_{x_r} (B(X_1, X_2, \dots, X_r))$  описывает только **факты**. Факты есть повествовательные предложения (или высказывания). Структура такого высказывания описывается предикатом, все аргументы которого есть предметные постоянные:  $v(a_1, a_2, \dots, a_r)$ . Например, если дан предикат  $P_1(X) := \langle X - \text{простое число} \rangle$ , то высказывание  $P_1(3)$  описывает факт. Или если дан предикат  $P_3(X, Y, Z) := \langle Z \text{ есть частное от деления числа } X \text{ на } Y \rangle$ , то высказывание  $P_3(6, 2, 3)$  также описывает факт.

Множество фактов есть экстенциональная часть дедуктивной системы и представляет «базу данных». Хотя в

современной реализации Visual Prolog работа с базами данных организуется отдельно.

Если заголовок пуст, то правая часть правила, т.е.

$:\exists x_1 \exists x_2 \dots \exists x_r (F_1(x_1, x_2, \dots, x_r), F_2(x_1, x_2, \dots, x_r), \dots, F_n(x_1, x_2, \dots, x_r))$  описывает запрос или *цель*.

Цель представляет собой утверждение, которое нужно доказать, исходя из множества фактов и правил программы.

Простейшая Prolog-программа включает 3 или 4 раздела:

Раздел DOMAINS содержит описание типов предикатных аргументов, определяемых пользователем. Если используются только стандартные типы, то этот раздел может отсутствовать.

Раздел PREDICATES содержит перечень всех предикатов с указаниями на типы аргументов и уровни детерминизма самих предикатов.

Раздел CLAUSES содержит формально записанные факты и правила («клаузы», «хорновские дизъюнкты»). Это основной раздел Prolog-программы.

Раздел GOAL содержит цель (запрос или вопрос) – формулировку теоремы, которую нужно доказать, исходя из множества фактов и правил, имеющихся в предыдущем разделе.

Принцип резолюции при истинности всех фактов требует доказательства  $\neg(\forall(x_1, x_2, \dots, x_r))$ , т.е. организовать вывод противоречия или тождественно ложной формулы из множества тождественно истинных аксиом и посылок, т.е.  $false \leftarrow true$ . Это известный в математике способ доказательства «от противного»: найти такие значения предметных

переменных  $x_1, x_2, \dots, x_r$ , при которых правило имеет ложное значение.

Множество фактов и правил вместе с целью представляют *логическую программу* для *интерпретатора*, который анализирует эти факты и правила и исполняет каждое отдельное предписание. Механизм процесса интерпретации использует процедуру унификации.

Рассмотрим действие интерпретатора на примере следующей логической программы [6]:

$R(a, b)$ . – факт 1,

$Q(b, g(c))$ . – факт 2,

$R(X, f(Y))$ :-  $R(X, Z), Q(Z, f(Y))$ . – правило 1,

$R(X, f(Y))$ :-  $R(X, Z), Q(Z, g(Y))$ . – правило 2,

$R(X, Z)$ :-  $Q(f(X), g(Z))$ . – правило 3.

Здесь  $a, b, c$  – предметные постоянные,  $X, Y, Z$  – предметные переменные,  $P, R, Q$  – предикаты,  $f, g$  – сколемовские функции.

Пусть целевой запрос есть формула  $\neg P(U, f(V))$ , где  $U$  и  $V$  – предметные переменные. При исчислении ответа интерпретатор пытается унифицировать цель с фактами. В нашем случае  $P(U, f(V))$  не унифицируется ни с одним из фактов. Тогда интерпретатор пытается унифицировать  $P(U, f(V))$  с заголовком одного из правил. Это возможно с заголовком первого и второго правил при выполнении подстановки  $\{X/U, Y/V\}$ . Но надо проверить истинность правила.

Интерпретатор формирует запрос  $R(X, Z), Q(Z, f(Y))$  для правой части первого правила. Цель  $R(X, Z)$  этого запроса достижима за счет унификации с первым фактом при подстановке  $\{a/X, b/Z\}$ . Цель  $Q(Z, f(Y))$  – не достижима.

ма, поскольку не унифицируется со вторым фактом (разные сколемовские функции).

Интерпретатор возвращается к цели  $P(U, f(V))$  и обращается ко второму правилу. Интерпретатор формирует запрос  $R(X, Z), Q(Z, g(Y))$  для правой части второго правила. Цель  $R(X, Z)$  запроса также достижима за счет унификации с первым фактом при подстановке  $\{a/X, b/Z\}$ . Цель  $Q(Z, g(Y))$  достижима за счет унификации со вторым фактом при подстановке  $\{b/Z, c/Y\}$ . Так подтверждается истинность правила для данных фактов.

Следовательно, исходная цель  $P(U, f(V))$  достижима при подстановке  $\{a/U, c/V\}$ .

Рассмотрим второй пример на естественном языке по родословной князей нашего государства в X веке [19].

*«Игорь – отец Святослава».*

*«Святослав – отец Владимира».*

*«Владимир – отец Бориса и Глеба».*

*«Дед кого-либо – это отец отца кого-либо».*

*«Два лица являются братьями, если у них общий отец».*

Три первых предложения описывают факты в форме предикатов:

*отец(игорь, святослав).*

*отец(святослав, владимир).*

*отец(владимир, борис).*

*отец(владимир, глеб).*

Здесь *отец* – имя предиката, *игорь, святослав, владимир, борис, глеб* – индивидуальные (предметные) постоянные.

Четвертое предложение задает правило:

*«Для любых  $X, Y, Z$ , если  $отец(X, Z)$  и  $отец(Z, Y)$ , то из*

этого следует, что  $\text{дед}(X, Y)$ ».

Запишем это правило на языке математической логики, используя кванторы всеобщности:

$$\forall_x \forall_y \forall_z (\text{дед}(X, Y) \leftarrow \text{отец}(X, Z) \& \text{отец}(Z, Y))$$

Удалим кванторы:  $\text{дед}(X, Y) \leftarrow \text{отец}(X, Z) \& \text{отец}(Z, Y)$ .

Пятое предложение задает правило:

«Для любых  $X, Y, Z$ , если  $\text{отец}(Z, X)$  и  $\text{отец}(Z, Y)$ , то из этого следует, что  $\text{брат}(X, Y)$ »:

$$\forall_x \forall_y \forall_z (\text{брат}(X, Y) \leftarrow \text{отец}(Z, X) \& \text{отец}(Z, Y)).$$

Удалим кванторы:  $\text{брат}(X, Y) \leftarrow \text{отец}(Z, X) \& \text{отец}(Z, Y)$ .

Так получено следующее множество фактов и правил:

$\text{отец}(\text{игорь}, \text{святослав})$ .

$\text{отец}(\text{святослав}, \text{vlадимир})$ .

$\text{отец}(\text{vlадимир}, \text{борис})$ .

$\text{отец}(\text{vlадимир}, \text{глеб})$ .

$\text{дед}(X, Y) \leftarrow \text{отец}(X, Z), \text{отец}(Z, Y)$ .

$\text{брат}(X, Y) \leftarrow \text{отец}(Z, X), \text{отец}(Z, Y)$ .

Запрос к множеству фактов возможен такой: «Святослав является дедом  $Y$ ». Формула этого запроса:  $\leftarrow \text{дед}(\text{святослав}, Y)$ .

Такой запрос равносильен клаузе:  $\neg \text{дед}(\text{святослав}, Y)$ , «Святослав не является дедом  $Y$ », которую можно добавить к множеству фактов. Если формула запроса  $\leftarrow \text{дед}(\text{святослав}, Y)$  логически следует из исходного множества клауз программы, то множество клауз с включенным в него новым фактом должно быть несовместимо.

Рассмотрим механизм логического вывода, являющийся основой большинства систем логического программирования (*SLD-резолюция*).

Выполним проверку запроса  $\leftarrow \text{дед}(\text{святослав}, Y)$  по правилу:

$$\begin{aligned} \leftarrow \text{дед}(\text{святослав}, Y) &\equiv \neg \text{дед}(\text{святослав}, Y) \\ \text{дед}(X, Y) \leftarrow \text{отец}(X, Z), \text{отец}(Z, Y) &\equiv \neg \text{дед}(X, Y) \rightarrow \neg(\text{отец}(X, Z) \& \\ &\& \text{отец}(Z, Y)) \equiv \forall_x \forall_y \forall_z (\neg \text{дед}(X, Y) \rightarrow \neg(\text{отец}(X, Z) \& \text{отец}(Z, Y))) \end{aligned}$$

По правилам вывода при подстановке  $X = \text{святослав}$ :

$$\frac{\forall_x \forall_y \forall_z (\neg \text{дед}(X, Y) \rightarrow \neg(\text{отец}(X, Z) \& \text{отец}(Z, Y)))}{\neg \text{дед}(\text{святослав}, Y) \rightarrow \neg(\text{отец}(\text{святослав}, Z) \& \text{отец}(Z, Y))}$$

По правилу *т.р.* выводим:

$$\frac{\begin{aligned} &\neg \text{дед}(\text{святослав}, Y), \\ &\neg \text{дед}(\text{святослав}, Y) \rightarrow \neg(\text{отец}(\text{святослав}, Z) \& \text{отец}(Z, Y)) \end{aligned}}{\neg(\text{отец}(\text{святослав}, Z) \& \text{отец}(Z, Y))}$$

и получаем новый запрос,

$$\neg \exists_z \exists_y (\text{отец}(\text{святослав}, Z) \& \text{отец}(Z, Y)),$$

который утверждает, что не существует таких индивидуумов  $Z$  и  $Y$ , что Святослав является отцом для  $Z$ , а  $Z$  является отцом для  $Y$ .

$$\begin{aligned} &\neg \exists_z \exists_y (\text{отец}(\text{святослав}, Z) \& \text{отец}(Z, Y)) \equiv \\ &\equiv \forall_z \forall_y \neg(\text{отец}(\text{святослав}, Z) \& \text{отец}(Z, Y)) \equiv \\ &\equiv \neg \text{отец}(\text{святослав}, Z) \vee \neg \text{отец}(Z, Y) \equiv \\ &\equiv \text{отец}(\text{святослав}, Z) \rightarrow \neg \text{отец}(Z, Y). \end{aligned}$$

Унифицируя левую часть импликации при подстановке  $Z = \text{владимир}$  и снова используя правило заключения, получим:

$$\frac{\text{отец}(\text{святослав}, \text{владимир}), \text{отец}(\text{святослав}, Z) \rightarrow \neg \text{отец}(Z, Y)}{\neg \text{отец}(\text{владимир}, Y)}$$

Таким образом, получаем новый запрос:

$$\begin{aligned} \neg \text{отец}(\text{владимир}, Y) &\equiv \leftarrow \text{отец}(\text{владимир}, Y) \equiv \\ &\equiv \text{отец}(\text{владимир}, Y) \rightarrow \text{false}. \end{aligned}$$

Унифицируя левую часть импликации при подстановке  $Y=\text{борис}$  и снова используя правило заключения, получим:

$$\frac{\text{отец}(\text{владимир}, \text{борис}), \text{отец}(\text{владимир}, Y) \rightarrow \text{false}}{\text{false}}$$

Унифицируя левую часть этой же импликации при подстановке  $Y=\text{глеб}$  и снова используя правило заключения, получим:

$$\frac{\text{отец}(\text{владимир}, \text{глеб}), \text{отец}(\text{владимир}, Y) \rightarrow \text{false}}{\text{false}}$$

Если рассматривать этот процесс менее формально, то он выглядит следующим образом. По запросу ищется самое первое правило или факт, которые могут быть унифицированы с запросом.

Запрос  $\leftarrow \text{дед}(\text{святослав}, Y)$  может быть унифицирован с правилом  $\text{дед}(X, Y) \leftarrow \text{отец}(X, Z), \text{отец}(Z, Y)$  при подстановке  $X=\text{святослав}$ , в результате получим запрос  $\leftarrow \text{отец}(\text{святослав}, Z), \text{отец}(Z, Y)$ ,

который, в свою очередь, может быть унифицирован с фактом:

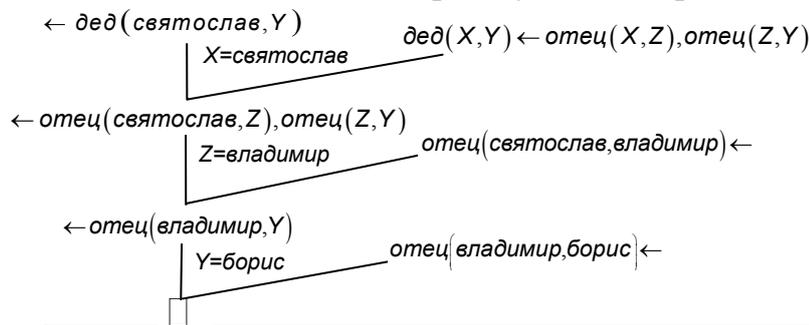
$\text{отец}(\text{святослав}, \text{владимир}) \leftarrow$ . При подстановке  $Z=\text{владимир}$  получится новый запрос  $\leftarrow \text{отец}(\text{владимир}, Y)$ . Он может быть унифицирован с любым из двух фактов:  $\text{отец}(\text{владимир}, \text{борис}) \leftarrow$ ,  $\text{отец}(\text{владимир}, \text{глеб}) \leftarrow$ .

При подстановках  $Y=\text{борис}$  или  $Y=\text{глеб}$  получатся новые запросы:

$\leftarrow \text{отец}(\text{владимир}, \text{борис})$ ,  
 $\leftarrow \text{отец}(\text{владимир}, \text{глеб})$ ,  
 упоминаемыми фактами.

Процесс, называемый *опровержением*, изображают в виде дерева (рис. 1.17).

Ответом на данный запрос будет  $Y=\text{борис}$  и  $Y=\text{глеб}$ .



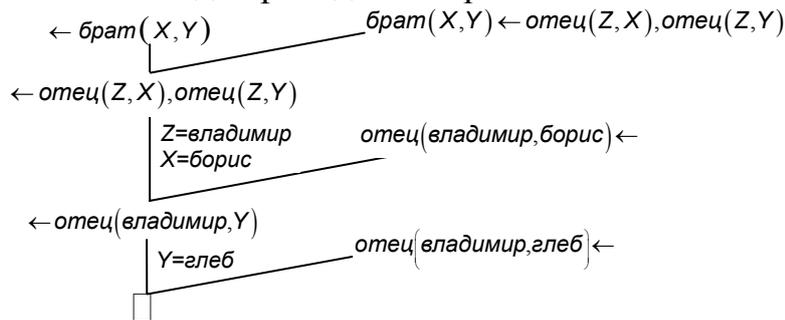
**Рис. 1.17.** Дерево «Дедом кого является Святослав?»

Для установления второй личности дерево строится повторно.

Также можно выполнить проверку запроса

$\leftarrow \text{брат}(X, Y)$ .

Ответом на запрос будет  $X=\text{борис}$ ,  $Y=\text{глеб}$ , дерево логического вывода приведено на рис. 1.18.

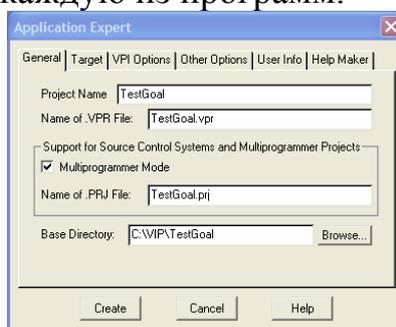


**Рис. 1.18.** Дерево "опровержения", реализующее запрос «Какие два человека являются братьями?»

### 1.2.3.2. Подготовка среды Visual Prolog для работы

Для изучения примеров предлагается использовать визуальную среду разработки Visual Prolog 5.2. На момент написания данных материалов последней версией Visual Prolog была 6.1, но она плохо подходит для проверки учебных примеров, так как не имеет встроенной утилиты Test Goal, позволяющей не акцентировать внимание на организации поиска с возвратом и вывода результатов выполнения программ. Test Goal находит все возможные решения GOAL и показывает значения всех переменных, используемых в GOAL (раздел запросов). При отсутствии этой утилиты необходима модификация программ (подробнее – в литературе по программированию на Прологе, например, [1]).

Перед началом работы с примерами необходимо создать специальный проект, в который впоследствии можно будет загружать каждую из программ.



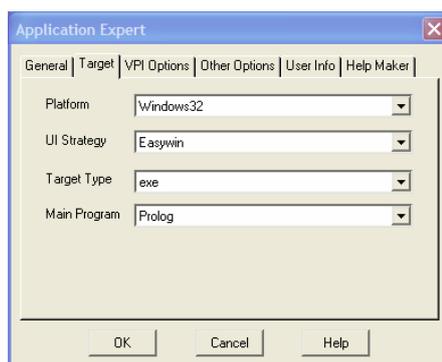
**Рис. 1.19.** Общие установки диалогового окна **Application Expert**.

Выберите команду главного меню **Project | New Project**, активизируется диалоговое окно **Application Expert**.

Переключитесь на вкладку **General**. В поле **Base Directory** задайте имя каталога для хранения программ, например, C:\VIP\TestGoal. Определите имя в поле **Project Name** - введите TestGoal. Установите флажок **Multiprogrammer Mode**. Щелкните мышью внутри поля **Name of .PRJ File** (в нем появится имя файла проекта). В результате окно должно выглядеть, как на рис. 1.19.

Переключитесь на вкладку **Target**. В поле **Platform** выберите Windows32, в поле **UI Strategy** – Easywin, в поле **Target Type** – exe, в поле **Main Program** – Prolog. В результате окно должно выглядеть, как на рис. 1.20.

Нажмите кнопку Create, чтобы создать файлы проекта



по умолчанию.

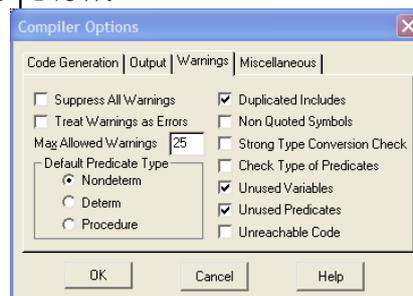
**Рис. 1.20.** Установки на вкладке **Target** диалогового окна **Application Expert**.

Выберите команду главного меню **Options | Project | Compiler Options**. Откройте вкладку **Warnings**. Установите переключатель **Nondeterm** (в этом случае компилятор будет принимать все определенные пользователем предикаты как недетерминированные по умолчанию, т. е. могут

порождать более одного решения). Снимите флажки **Non Quoted Symbols**, **Strong Type Conversion Check**, **Check Type of Predicates** (для подавления некоторых возможных предупреждений компилятора). В результате окно должно выглядеть, как на рис. 1.21. Нажмите кнопку **ОК**.

Сохраните настроенный проект: **Project | Save Project**.

Теперь для работы с примерами достаточно открыть этот TestGoal-проект командой меню **Project | Open Project**, загрузить текст программы командой меню **File | Open** или создать текст новой программы, используя команду меню **File | New**.



**Рис.1.21.** Установки опций компилятора.

Для тестирования программы *необходимо* использовать команду меню **Project | Test Goal** (<Ctrl>+<G>, имеется кнопка на стандартной панели инструментов).

### 1.2.3.3. Описание логических задач на языке Prolog

Рассмотрим описание примеров логики высказываний и предикатов на языке Prolog (нумерация примеров повторяет примеры, рассмотренные выше).

Примеры 1.1 и 1.2. Даны простые и сложные высказывания.

1. если  $A_1 := \langle 3 \text{ - простое число} \rangle$ , то  $A_1 = \text{и}$ ,
2. если  $A_2 := \langle 3 \text{ - вещественное число} \rangle$ , то  $A_2 = \text{и}$ ,
3. если  $A_3 := \langle 3 \text{ - целое число} \rangle$ , то  $A_3 = \text{и}$ ,
4. если  $B_1 := \langle 3, 14 \dots \text{- рациональное число} \rangle$ , то  $B_1 = \text{л}$ ,
5. если  $B_2 := \langle 3, 14 \dots \text{- не рациональное число} \rangle$ , то  $B_2 = \text{и}$ ,
6. если  $C := \langle \text{Колумб открыл Америку} \rangle$ , то  $C = \text{и}$ ,
7. если  $D := \langle \text{Киев - столица Узбекистана} \rangle$ , то  $D = \text{л}$ ,
8. если  $E := \langle \text{Число 6 делится на 1, 2 и 3} \rangle$ , то  $E = \text{и}$ ,
9. если  $G := \langle \text{Число 6 есть сумма чисел 1, 2, 3} \rangle$ , то  $G = \text{и}$ .
10. если высказывание:  $\langle 3 \text{ - вещественное и целое число} \rangle$ , то формула  $(A_2 \& A_3) = \text{и}$ ,
11. если высказывание:  $\langle 3, 14 \dots \text{- рациональное число} \rangle$ , то формулы  $B_1 = \text{л}$  или  $\neg B_1 = \text{и}$ ,
12. если высказывание:  $\langle \text{число 6 делится на 1, 2, 3 и представляет сумму делителей 1, 2, 3} \rangle$ , то формула  $(E \& G) = \text{и}$ ,
13. для высказывания:  $\langle \text{если 3 - целое число, то оно вещественное} \rangle$ , справедлива формула  $(A_3 \rightarrow A_2) = \text{и}$ ,
14. для высказывания  $\langle \text{если 3 - простое число, то оно целое} \rangle$ , справедлива формула  $(A_1 \rightarrow A_3) = \text{и}$ ,
15. для высказывания  $\langle 3 \text{ - простое число тогда и только тогда, когда оно целое} \rangle$ , справедлива формула  $(A_1 \leftrightarrow A_3) = \text{л}$ .

Программа на языке Prolog:

```

predicates
    a1()
    a2()
    a3()

```

```
b2()
c()
e()
g()
a2_and_a3()
e_and_g()
a3_implies_a2()
a1_implies_a3()
clauses
a1.
a2.
a3.
b2.
c.
e.
g.
a2_and_a3:-a2,a3.
e_and_g:-e,g.
a3_implies_a2:-not(a3);a2.
a1_implies_a3:-not(a1);a3.
goal
a2_and_a3.
%e_and_g.
%a3_implies_a2.
%a1_implies_a3.
```

Пояснения к программе: имена безаргументных предикатов a1, a2, a3, b2, c, e, g соответствуют простым высказываниям 1, 2, 3, 5, 6, 8, 9. Имена предикатов a2\_and\_a3, e\_and\_g, a3\_implies\_a2, a1\_implies\_a3 соответствуют сложным высказываниям 10, 12, 13, 14. Предикаты

b1, d для простых высказываний 4 и 7, а также предикат для сложного высказывания 15 отсутствуют, так как они ложны. А ложные высказывания не могут быть фактами.

Простые высказывания перечислены как **факты**, а сложные высказывания представлены **правилами**, благодаря чему можно судить об истинности или ложности этих высказываний на основании истинности или ложности простых высказываний, входящих в формулы более сложных. Здесь следует отметить, что импликации  $(A_3 \rightarrow A_2)$  и  $(A_1 \rightarrow A_3)$  заменены их эквивалентными формулами  $(\neg A_3 \vee A_2)$  и  $(\neg A_1 \vee A_3)$ . Not – это встроенный предикат Prolog, аргументом которого может быть только другой предикат. Он принимает значение истины только в том случае, если *нельзя вывести* истинность аргумента.

При запуске программы в режиме Test Goal получим сообщение «yes» в ответ на запрос a2\_and\_a3, что означает, что теорема, предъявленная в запросе, выводима из множества фактов и правил программы, т. е. в данном случае мы получили информацию об истинности высказывания a2\_and\_a3.

Поочередно снимая знак % с последующих запросов и устанавливая его на уже проверенных (знак процента перед строкой превращает ее в комментарий, не обрабатываемый компилятором), получим тот же ответ для остальных трех запросов.

Пример 1.12. Дана формула высказывания:

$$F = (A \rightarrow (B \vee C)) \& (B \rightarrow D) \& ((D \& A) \rightarrow \neg C)$$

При каких значениях истинности A, B, C, D формула имеет значение «и»?

Программа:

```
domains
    res=true,false;disj(res,res);conj(res,res);
    ne(res)
predicates
    imp(res,res)
    conj(res,res)
    disj(res,res)
    ne(res)
    f(res,res,res,res)
    mogetbyty(res)
clauses
    mogetbyty(true).mogetbyty(false).
    imp(true,true).
    imp(false,_).
    imp(true,disj(X,Y)):-disj(X,Y).
    imp(conj(X,Y),ne(Z)):-
        conj(X,Y),ne(Z);
        not(conj(X,Y)).
    conj(true,true).
    disj(true,true).
    disj(true,false).
    disj(false,true).
    ne(false).
    f(A,B,C,D):-
        imp(A,disj(B,C)),
        imp(B,D),
        imp(conj(D,A),ne(C)).
goal
    mogetbyty(A),
```

mogetbyty(B),  
mogetbyty(C),  
mogetbyty(D),  
f(A,B,C,D).

Пояснения к программе: предикаты `imp`, `conj`, `disj`, `ne` означают, соответственно, операции импликации, конъюнкции, дизъюнкции и отрицания. Четырехместный предикат `f` используется для записи конечной формулы. Предикат `mogetbyty` принимает значение истины как для значения аргумента `true`, так и для `false`. Это нужно для того, чтобы компилятор Prolog не возвращал в качестве результатов значений элементарных высказываний знак подчеркивания, означающий, что на его месте может быть любое из двух значений истинности. Сами значения истинности представлены словами `true` и `false` в разделе `domains`. Помимо них, значениями аргументов предикатов могут быть значения дизъюнкции, конъюнкции и отрицания. Поэтому в том же разделе должны быть описаны *составные домены*. Составные домены есть способ описания многоуровневых данных. В предложении

`imp(false,_)`.

знак подчеркивания в качестве второго аргумента означает, что безразлично, какое значение может там быть (вспомним: импликация всегда истинна, если слева от знака импликации стоит ложное высказывание). В программе все предложения, относящиеся к одному предикату – факты и правила – должны быть собраны вместе (иначе получим сообщение компилятора об ошибке). Просмотр последовательности фактов и правил идет сверху вниз, и это обстоятельство важно, когда выбор основания для заключе-

ния должен производиться с учетом приоритета предложений.

Запуск программы в режиме Test Goal дает следующие результаты:

A=true, B=true, C=false, D=true  
 A=true, B=false, C=true, D=false  
 A=false, B=true, C=true, D=true  
 A=false, B=true, C=false, D=true  
 A=false, B=false, C=true, D=true  
 A=false, B=false, C=true, D=false  
 A=false, B=false, C=false, D=true  
 A=false, B=false, C=false, D=false  
 8 Solutions

Пример 1.13. Дано суждение:

$$\frac{A \rightarrow B, A \vee C, C \rightarrow \neg D, D}{B}$$

При каких значениях А, В, С, D истинны посылки и заключение? Для истинности посылок и ожидаемой истинности заключения задачу следует преобразовать в формулу:

$$F = (A \rightarrow B) \& (A \vee C) \& (C \rightarrow \neg D) \& D \& B$$

Программа:

domains

res=true;false;ne(res)

predicates

imp(res,res)

disj(res,res)

ne(res)

vyskaz(res)

f(res,res,res,res)

```
    mogetbyty(res)
clauses
    mogetbyty(true).mogetbyty(false).
    imp(true,true).
    imp(false,_).
    imp(true,ne(Y)):-ne(Y).
    disj(true,true).
    disj(true,false).
    disj(false,true).
    ne(false).
    vyskaz(true).
    f(A,B,C,D):-
        imp(A,B),
        disj(A,C),
        imp(C,ne(D)),
        vyskaz(D),
        vyskaz(B).
```

```
goal
    mogetbyty(A),
    mogetbyty(B),
    mogetbyty(C),
    mogetbyty(D),
    f(A,B,C,D).
```

Запуск программы с помощью Test Goal дает единственное решение:

A=true, B=true, C=false, D=true

1 Solution

Пример 1.15. Дано суждение:

$$\frac{(A \vee B) \rightarrow (C \vee D), C \leftrightarrow (A \& \neg D), B \rightarrow (\neg C \vee \neg A)}{(\neg D \rightarrow (\neg A \& \neg C))}$$

Как и в предыдущем примере, преобразуем исходное описание в формулу:

$$F = ((A \vee B) \rightarrow (C \vee D)) \& (C \leftrightarrow (A \& \neg D)) \& \\ \& (B \rightarrow (\neg C \vee \neg A)) \& (\neg D \rightarrow (\neg A \& \neg C))$$

Программа:

domains

```
res=true;false;disj1(res,res);
disj2(ne_res,ne_res);conj1(res,ne_res);
conj2(ne_res,ne_res);ne(res)
ne_res=ne(res)
```

predicates

```
imp(res,res)
equiv(res,res)
conj1(res,res)
conj2(res,res)
disj1(res,res)
disj2(res,res)
ne(res)
f(res,res,res,res)
mogetbyty(res)
```

clauses

```
mogetbyty(true).mogetbyty(false).
imp(true,true).
imp(false,_).
imp(disj1(A,B),disj1(C,D)):-
    disj1(A,B),disj1(C,D);
    not(disj1(A,B)).
imp(true,disj2(ne(C),ne(A))):-disj2(ne(C),ne(A)).
```

```

imp(ne(D),conj2(ne(A),ne(C))):-
    ne(D),conj2(ne(A),ne(C));
    not(ne(D)).
equiv(true,true).
equiv(false,false).
equiv(true,conj1(A,ne(D))):-conj1(A,ne(D)).
equiv(false,conj1(A,ne(D))):-not(conj1(A,ne(D))).
conj1(true,ne(D)):-ne(D).
conj2(ne(A),ne(C)):-ne(A),ne(C).
disj1(true,true).
disj1(true,false).
disj1(false,true).
disj2(ne(C),ne(A)):-
    ne(C);
    ne(A).
ne(false).
f(A,B,C,D):-
    imp(disj1(A,B),disj1(C,D)),
    equiv(C,conj1(A,ne(D))),
    imp(B,disj2(ne(C),ne(A))),
    imp(ne(D),conj2(ne(A),ne(C))).

```

goal

```

mogetbyty(A),
mogetbyty(B),
mogetbyty(C),
mogetbyty(D),
f(A,B,C,D).

```

Пояснения к программе: в данном примере используются двухуровневые составные домены (disj2, conj1, conj2). Декларация домена объявляет только один уровень

дерева, а не целое дерево, поэтому объект `ne_res` объявлен как отдельный одноуровневый составной домен.

В программе присутствует описание операции эквивалентности (предикат `equiv`). Предикаты для операций конъюнкции и дизъюнкции разделены на два типа для каждой операции (основанием для деления послужили различные сочетания типов аргументов операций). Можно обойтись и одним предикатом для дизъюнкции и одним для конъюнкции, но при объявлении доменов все составные объекты должны иметь различные функторы (т. е. имена перед скобкой). Для лучшего восприятия программы тот же принцип применен и к соответствующим предикатам.

Запуск программы с помощью `Test Goal` дает следующие результаты:

```
A=true, B=true, C=false, D=true  
A=true, B=false, C=false, D=true  
A=false, B=true, C=false, D=true  
A=false, B=true, C=false, D=true  
A=false, B=false, C=false, D=true  
A=false, B=false, C=false, D=false  
6 Solutions
```

В примере было предложено 5 удовлетворительных наборов значений пропозициональных переменных, а здесь получилось 6. Обратите внимание, третья и четвертая строки в результате, выданном программой, совпадают. Это объясняется спецификой работы механизма поиска с возвратом (информация об этом – в любой литературе по Prolog), за счет чего возможно порождение избыточных решений. Это не является ошибкой.

Пример (см. таблицу 1.6). Дана формула:

$$F = (A \rightarrow B) \& (A \rightarrow C) \rightarrow (A \rightarrow (B \& C))$$

Проверить ее тождественную истинность.

Программа:

domains

res=true;false;conj1(imp,imp);imp1(res,conj);

imp(res,res);conj(res,res)

imp=imp(res,res)

conj=conj(res,res)

predicates

imp(res,res)

conj(res,res)

f(res,res,res)

mogetbyty(res)

clauses

mogetbyty(true).mogetbyty(false).

imp(true,true).

imp(false,\_).

imp(true,conj(B,C)):-conj(B,C).

imp(conj(imp(A,B),imp(A,C)),imp(A,conj(B,C))):-

conj(imp(A,B),imp(A,C)),imp(A,conj(B,C));

not(conj(imp(A,B),imp(A,C))).

conj(true,true).

f(A,B,C):-

imp(conj(imp(A,B),imp(A,C)),imp(A,conj(B,C))).

goal

mogetbyty(A),

mogetbyty(B),

mogetbyty(C),

$f(A,B,C)$ .

Запуск программы с помощью Test Goal дает следующие результаты:

A=true, B=true, C=true

A=true, B=true, C=false

A=true, B=false, C=true

A=true, B=false, C=false

A=false, B=true, C=true

A=false, B=true, C=false

A=false, B=false, C=true

A=false, B=false, C=false

8 Solutions

Таким образом, формула истинна во всех восьми интерпретациях, т. е. является тождественно истинной.

Пример (см. таблицу 1.7). Дана формула:

$$F = A \& (\neg B \vee \neg C) \& (A \rightarrow B) \& (A \rightarrow C)$$

Проверить ее тождественную ложность.

Программа:

domains

res=true;false;ne(res)

predicates

imp(res,res)

disj(res,res)

vyskaz(res)

ne(res)

f(res,res,res)

mogetbyty(res)

clauses

mogetbyty(true).mogetbyty(false).

```

imp(true,true).
imp(false,_).
disj(ne(X),ne(Y)):-ne(X);ne(Y).
vyskaz(true).
ne(false).
f(A,B,C):-
    vyskaz(A),
    disj(ne(B),ne(C)),
    imp(A,B),
    imp(A,C).

```

goal

```

mogetbyty(A),
mogetbyty(B),
mogetbyty(C),
f(A,B,C).

```

Запуск программы с помощью Test Goal дает результат No Solution, что говорит об отсутствии такой интерпретации, в которой формула истинна. Следовательно, она тождественно ложна.

Пример (см. таблицу 1.8). Дана формула:

$$F = (\neg A \rightarrow C \ \& \ \neg B) \ \& \ (\neg C \rightarrow B \ \& \ \neg A) \ \& \ (A \rightarrow \neg C)$$

При каких значениях А, В, С формула имеет значение истины?

Программа:

domains

```
res=true;false;ne(res);conj(res,ne_res)
```

```
ne_res=ne(res)
```

predicates

```
imp(res,res)
```

```
conj(res,res)
ne(res)
f(res,res,res)
mogetbyty(res)
clauses
mogetbyty(true).mogetbyty(false).
imp(true,true).
imp(false,_).
imp(ne(X),conj(Y,ne(Z))):-
    ne(X),conj(Y,ne(Z));
    not(ne(X)).
imp(true,ne(X)):-ne(X).
conj(true,true).
conj(true,ne(X)):-ne(X).
ne(false).
f(A,B,C):-
    imp(ne(A),conj(C,ne(B))),
    imp(ne(C),conj(B,ne(A))),
    imp(A,ne(C)).
goal
mogetbyty(A),
mogetbyty(B),
mogetbyty(C),
f(A,B,C).
```

Запуск программы с помощью Test Goal дает следующий результат:

A=false, B=false, C=true

1 Solution

Пример 1.40.

Доказать истинность заключения:

$$\frac{(A \rightarrow B) \& (C \rightarrow D), (D \& B \rightarrow E), \neg E}{(\neg C \vee \neg A)}$$

- посылка  $(A \rightarrow B)$  - правило,
  - посылка  $(C \rightarrow D)$  - правило,
  - посылка  $(D \& B \rightarrow E)$  - правило,
  - отрицание заключения  $\neg(\neg C \vee \neg A) \equiv (C \& A)$  - два факта:  $C$  и  $A$ ,
  - целью удобно поставить отрицание посылки  $\neg(\neg E) \equiv E$ .
- Имеет смысл еще раз обратить внимание на то, что цель представляет собой выражение  $(false \leftarrow E)$ , записываемое без слова *false* и знака обратной импликации, т. е. просто  $E$ . В процессе же унификации дизъюнктов участвует эквивалентное выражение  $(\neg E \leftarrow true)$ , представляющее, по своей сути, однолитерный дизъюнкт  $\neg E$ .

Программа:

```

predicates
  a()
  b()
  c()
  d()
  e()
clauses
  b:-a.
  d:-c.
  e:-d,b.
  c.
  a.
goal

```

е.

Запуск программы с помощью Test Goal сопровождается результатом yes, что говорит о том, что  $E$  логически следует из множества фактов и правил программы, следовательно, множество посылок и отрицание заключения несовместимы. Это эквивалентно получению пустой резольвенты по классическому методу резолюции.

Пример 1.41.

Доказать истинность заключения:

$$\frac{((A \vee B) \rightarrow C), (C \rightarrow (D \vee M)), (M \rightarrow N), ((\neg D) \& (\neg N))}{\neg A}$$

- посылка  $((A \vee B) \rightarrow C)$  - правило,
- посылку  $(C \rightarrow (D \vee M))$  необходимо преобразовать:  
 $(C \rightarrow (D \vee M)) \equiv (\neg C \vee D \vee M) \equiv (C \& \neg D \rightarrow M)$  - правило,
- посылка  $(M \rightarrow N)$  - правило,
- посылка  $\neg D$  - факт,
- отрицание заключения  $\neg(\neg A) \equiv A$  - факт,
- целью поставим отрицание посылки  $\neg(\neg N) \equiv N$ .

Программа:

predicates

a()

failure b()

c()

neD()

m()

n()

clauses

b.

$c:-a;b.$   
 $m:-c,neD.$   
 $n:-m.$   
 $neD.$   
 $a.$

goal  
n.

Пояснения к программе: так как в правиле  $c:-a;b.$  используется обращение к  $b$ , то необходимо, чтобы этот предикат был описан в разделе `predicates` и для него было, по крайней мере, одно предложение в разделе `clauses`. Мы не располагаем информацией об истинности  $B$  и, поэтому, можем считать, что высказывание  $B$  не порождает решения. В программу в связи с этим введен предикат  $b$ , заданный с уровнем детерминизма `failure` (такие предикаты не порождают решения при вызове их в теле или заголовке правила).

Запуск программы с помощью `Test Goal` сопровождается результатом `yes`, что говорит о том, что  $N$  логически следует из множества фактов и правил программы, следовательно, множество посылок и отрицание заключения несовместимы. Это эквивалентно получению пустой резольвенты по классическому методу резолюции.

#### Пример 1.43.

Доказать истинность заключения:

$$\frac{(A \& B \rightarrow C), (C \& D \rightarrow \neg M), (\neg N \rightarrow D \& M)}{(A \& B \rightarrow N)}$$

- посылка  $(A \& B \rightarrow C)$  - правило,

- посылка  $(C \& D \rightarrow \neg M) \equiv (\neg C \vee \neg D \vee \neg M)$  - все литеры в дизъюнкте отрицательные. Отрицание данной посылки удобно поставить в раздел цели:

$$\neg(\neg C \vee \neg D \vee \neg M) \equiv (C \& D \& M) - \text{цель,}$$

- посылка  $(\neg N \rightarrow D \& M) \equiv ((\neg N \rightarrow D) \& (\neg N \rightarrow M))$  - два правила,
- отрицание заключения  $\neg(A \& B \rightarrow N) \equiv (A \& B \& \neg N)$  - три факта.

Программа:

predicates

a()

b()

c()

d()

m()

neN()

clauses

c:-a,b.

d:-neN.

m:-neN.

a.

b.

neN.

goal

c,d,m.

Запуск программы с помощью Test Goal дает результатом yes, что говорит о том, что  $(C \& D \& M)$  логически следует из множества фактов и правил программы, следовательно, множество посылок и отрицание заключения несо-

вместимы. Это эквивалентно получению пустой резольвенты по классическому методу резолюции.

Пример 1.44.

Доказать истинность заключения:

$$\frac{(A \rightarrow B) \& (C \rightarrow D), (D \& B \rightarrow M), \neg M}{(\neg A \vee \neg C)}$$

- посылка  $(A \rightarrow B)$  - правило,
- посылка  $(C \rightarrow D)$  - правило,
- посылка  $(D \& B \rightarrow M)$  - правило,
- целью поставим отрицание посылки  $\neg(\neg M) \equiv M$ ,
- отрицание заключения  $\neg(\neg A \vee \neg C) \equiv (A \& C)$  - два факта.

Программа:

predicates

a()

b()

c()

d()

m()

clauses

b:-a.

d:-c.

m:-d,b.

a.

c.

goal

m.

Запуск программы с помощью Test Goal сопровождается результатом yes, что говорит о том, что  $M$  логически

следует из множества фактов и правил программы, следовательно, множество посылок и отрицание заключения несовместимы. Это эквивалентно получению пустой резольвенты по классическому методу резолюции.

Пример 1.45. Доказать истинность заключения:

$$\frac{((\neg A \vee \neg B \vee \neg A \& \neg B) \rightarrow C), ((A \vee B \vee A \& B) \rightarrow \neg C)}{(C \rightarrow \neg A)}$$

Упростим посылки, используя закон поглощения:

$$\frac{((\neg A \vee \neg B) \rightarrow C), ((A \vee B) \rightarrow \neg C)}{(C \rightarrow \neg A)}$$

- посылка  $((\neg A \vee \neg B) \rightarrow C)$  - правило,
- посылка  $((A \vee B) \rightarrow \neg C) \equiv ((\neg A \vee \neg C) \& (\neg B \vee \neg C))$ , отрицание этой формулы удобно поставить целью:  
 $\neg((\neg A \vee \neg C) \& (\neg B \vee \neg C)) \equiv ((A \& C) \vee (B \& C))$ ,
- отрицание заключения  $\neg(C \rightarrow \neg A) \equiv (C \& A)$  - два факта.

Программа:

predicates

a()

failure b()

c()

clauses

a.

b.

c.

c:-not(a);not(b).

goal

a,c;

b,c.

Запуск программы с помощью Test Goal дает результат yes, что говорит о том, что формула  $((A \& C) \vee (B \& C))$  логически следует из множества фактов и правил программы, следовательно, множество посылок и отрицание заключения несовместимы. Это эквивалентно получению пустой резольвенты по классическому методу резолюции.

Поскольку все последующие задачи – примеры на исчисление предикатов, все посылки и отрицание заключения будут сначала приведены к сколемовской стандартной форме.

Пример 1.59. Доказать истинность заключения:

$$\frac{\exists_x (P_1(x) \& \forall_y (P_2(y) \rightarrow P_4(x, y))), \forall_x (P_1(x) \rightarrow \forall_y (P_3(y) \rightarrow \neg P_4(x, y)))}{\forall_x (P_2(x) \rightarrow \neg P_3(x))}$$

- посылка  $\exists_x (P_1(x) \& \forall_y (P_2(y) \rightarrow P_4(x, y))) \equiv$   
 $\equiv \exists_x \forall_y (P_1(x) \& (\neg P_2(y) \vee P_4(x, y))) \equiv \forall_y (P_1(a) \& (\neg P_2(y) \vee P_4(a, y)))$  - дает факт  $P_1(a)$  и правило  $(P_2(y) \rightarrow P_4(a, y))$ ,
- посылка  $\forall_x (P_1(x) \rightarrow \forall_y (P_3(y) \rightarrow \neg P_4(x, y))) \equiv$   
 $\Leftrightarrow \forall_x \forall_y (P_1(x) \rightarrow (\neg P_3(y) \vee \neg P_4(x, y))) \equiv \forall_x \forall_y (\neg P_1(x) \vee \neg P_3(y) \vee \neg P_4(x, y))$ ,  
 так как в дизъюнкте все литералы отрицательные, то его отрицание следует поставить как цель в программе; при этом не будем заменять предметные переменные предметными постоянными, чтобы предоставить Prolog самостоятельно найти подходящие значения переменных. Получим цель  $(P_1(x) \& P_3(y) \& P_4(x, y))$ .
- отрицание заключения  $\neg \forall_x (P_2(x) \rightarrow \neg P_3(x)) \equiv$   
 $\equiv \exists_x (\neg(\neg P_2(x) \vee \neg P_3(x))) \equiv \exists_x (P_2(x) \& P_3(x)) \equiv (P_2(b) \& P_3(b))$  - два факта.

Программа:  
 predicates  
   p1(symbol)  
   p2(symbol)  
   p3(symbol)  
   p4(symbol,symbol)  
 clauses  
   p1(a).  
   p4(a,Y):-p2(Y).  
  
   p2(b).  
   p3(b).  
 goal  
   p1(X),  
   p3(Y),  
   p4(X,Y).

С помощью Test Goal получен результат:

X=a, Y=b

1 Solution,

что говорит о том, что формула  $(P_1(a) \& P_3(b) \& P_4(a,b))$  логически следует из множества фактов и правил программы, следовательно, множество посылок и отрицание заключения несовместимы. Это эквивалентно получению пустой резольвенты по классическому методу резолюции.

Пример 1.60. Доказать истинность заключения:

$$\frac{\neg \exists_x (P_1(x) \& P_2(x)), \exists_x (P_2(x) \& P_3(x))}{\exists_x (P_3(x) \& \neg P_1(x))}$$

- посылка  $\neg \exists_x (P_1(x) \& P_2(x)) \equiv \forall_x (\neg P_1(x) \vee \neg P_2(x))$ , так как в дизъюнкте все литералы отрицательные, то его отрицание сле-

дует поставить как цель в программе; при этом не будем заменять предметную переменную предметной постоянной, чтобы предоставить Prolog самостоятельно найти подходящее значение переменной. Получим цель  $(P_1(x) \& P_2(x))$ ,

- посылка  $\exists_x (P_2(x) \& P_3(x)) \equiv (P_2(a) \& P_3(a))$  - дает два факта,
- отрицание заключе-  
ния  $\neg \exists_x (P_3(x) \& \neg P_1(x)) \equiv \forall_x (\neg P_3(x) \vee P_1(x)) \equiv (\neg P_3(x) \vee P_1(x)) \equiv P_3(x) \rightarrow P_1(x)$ . -  
правило.

Программа:

```

predicates
    p1(symbol)
    p2(symbol)
    p3(symbol)
clauses
    p2(a).
    p3(a).
    p1(X):-p3(X).
goal
    p1(X),
    p2(X).

```

С помощью Test Goal получим результат:

X=a

1 Solution,

что говорит о том, что формула  $(P_1(a) \& P_2(a))$  логически следует из множества фактов и правил программы, следовательно, множество посылок и отрицание заключения несовместимы. Это эквивалентно получению пустой резольвенты по классическому методу резолюции.

Пример 1.77. Доказать истинность заключения:

$$\frac{\forall_y (P_3(y) \& \neg P_4(y) \rightarrow \forall_x (P_1(x) \& P_2(x, y))), \quad \exists_y (P_3(y) \& P_5(y)), \forall_y (P_3(y) \& P_4(y) \rightarrow \neg P_5(y))}{\exists_x (P_1(x) \& P_5(x))}$$

- посылка  $\forall_y (P_3(y) \& \neg P_4(y) \rightarrow \forall_x (P_1(x) \& P_2(x, y))) \equiv$   
 $\equiv \forall_y \forall_x (P_3(y) \& \neg P_4(y) \rightarrow (P_1(x) \& P_2(x, y))) \equiv$   
 $\equiv \forall_y \forall_x (\neg P_3(y) \vee P_4(y) \vee (P_1(x) \& P_2(x, y))) \equiv$   
 $\equiv \forall_y \forall_x ((\neg P_3(y) \vee P_4(y) \vee P_1(x)) \& (\neg P_3(y) \vee P_4(y) \vee P_2(x, y)))$

Получаем два правила:

$(P_3(y) \& \neg P_4(y) \rightarrow P_1(x))$  и  $(P_3(y) \& \neg P_4(y) \rightarrow P_2(x, y))$ ,

- посылка  $\exists_y (P_3(y) \& P_5(y)) \equiv (P_3(a) \& P_5(a))$  - два факта,
- посылка  $\forall_y (P_3(y) \& P_4(y) \rightarrow \neg P_5(y)) \equiv$   
 $\equiv \forall_y (\neg P_3(y) \vee \neg P_4(y) \vee \neg P_5(y)) \equiv \forall_y (P_3(y) \& P_5(y) \rightarrow \neg P_4(y))$  - правило,
- целью сделаем заключение  $\exists_x (P_1(x) \& P_5(x))$  (предметную переменную не будем заменять предметной постоянной, чтобы механизм поиска сам нашел подходящее значение  $x$ ).

Программа:

predicates

p1(symbol)

p2(symbol,symbol)

p3(symbol)

neP4(symbol)

p5(symbol)

clauses

p1(\_):-p3(Y),neP4(Y).

p2(\_,Y):-p3(Y),neP4(Y).

$p3(a).$   
 $p5(a).$   
 $neP4(Y):-p3(Y),p5(Y).$   
 goal  
 $p5(X),$   
 $p1(X).$

Пояснения к программе:

Знаки подчеркивания на месте аргумента в заголовках правил означают использование анонимной переменной (нельзя поставить конкретную переменную, так как она окажется несвязанной).

С помощью Test Goal получим следующий результат:

$X=a$

1 Solution,

что говорит о том, что формула  $(P_5(a) \& P_1(a))$  логически следует из множества фактов и правил программы, следовательно, множество посылок и отрицание заключения несоместимы. Это эквивалентно получению пустой резольвенты по классическому методу резолюции.

Пример 1.75

Доказать истинность заключения:

$$\frac{\forall_x (\exists_y (P_1(x,y) \& P_2(y)) \rightarrow \exists_y (P_3(y) \& P_4(x,y))), \neg \exists_x (P_3(x))}{\neg \exists_x (P_3(x)) \rightarrow \forall_x \forall_y (P_1(x,y) \rightarrow (\neg P_2(y)))}$$

- ПОСЫЛКА  $\forall_x (\exists_y (P_1(x,y) \& P_2(y)) \rightarrow \exists_y (P_3(y) \& P_4(x,y))) \equiv$   
 $\equiv \forall_x (\forall_y (\neg P_1(x,y) \vee \neg P_2(y)) \vee \exists_y (P_3(y) \& P_4(x,y))) \equiv$   
 $\equiv \forall_x \forall_v (\neg P_1(x,v) \vee \neg P_2(v) \vee \exists_y (P_3(y) \& P_4(x,y))) \equiv$   
 $\equiv \forall_x \forall_v \exists_y (\neg P_1(x,v) \vee \neg P_2(v) \vee (P_3(y) \& P_4(x,y))) \equiv$   
 $\equiv \forall_x \forall_v ((\neg P_1(x,v) \vee \neg P_2(v) \vee P_3(f(x,v))) \&$

$\&(\neg P_1(x,v) \vee \neg P_2(v) \vee P_4(x,f(x,v)))$  - два правила:

$(P_1(x,v) \& P_2(v) \rightarrow P_3(f(x,v)))$  и  $(P_1(x,v) \& P_2(v) \rightarrow P_4(x,f(x,v)))$ ,

- целью поставим отрицание посылки  $\neg(\neg\exists_x(P_3(x))) \equiv \exists_x(P_3(x))$

(не будем вводить предметную постоянную и предоставим найти значение переменной  $x$  Prolog),

- подготовим заключение к тому, чтобы взять его отрицание:

$$\begin{aligned} &\neg\exists_x(P_3(x) \rightarrow \forall_x\forall_y(P_1(x,y) \rightarrow (\neg P_2(y)))) \equiv \\ &\equiv \exists_x(P_3(x) \vee \forall_x\forall_y(\neg P_1(x,y) \vee \neg P_2(y))) \equiv \\ &\equiv \exists_v(P_3(v) \vee \forall_x\forall_y(\neg P_1(x,y) \vee \neg P_2(y))) \equiv \\ &\equiv \exists_v\forall_x\forall_y(P_3(v) \vee \neg P_1(x,y) \vee \neg P_2(y)) \end{aligned}$$

Тогда отрицание заключения:  $\neg(\exists_v\forall_x\forall_y(P_3(v) \vee \neg P_1(x,y) \vee \neg P_2(y))) \equiv$

$$\equiv \forall_v\exists_x\exists_y(\neg P_3(v) \& P_1(x,y) \& P_2(y)) \equiv$$

$\equiv \forall_v(\neg P_3(v) \& P_1(g(v),h(v)) \& P_2(h(v)))$  - три факта, но первый дизъюнкт уже есть среди посылок, поэтому его учитывать не нужно.

Программа:

domains

arg=f(arg,arg);g(arg);h(arg)

predicates

p1(arg,arg)

p2(arg)

p3(arg)

p4(arg,arg)

clauses

p3(f(X,V)):-p1(X,V),p2(V).

p4(X,f(X,V)):-p1(X,V),p2(V).

p1(g(V),h(V)).

p2(h(\_)).

goal

$p_3(X)$ .

С помощью Test Goal получим следующий результат:

$X=f(g(\_),h(\_))$

1 Solution,

что означает, что формула  $P_3(f(g(t),h(t)))$ , где  $t$  – произвольный терм, логически следует из множества фактов и правил программы, следовательно, множество посылок и отрицание заключения несовместимы. Это эквивалентно получению пустой резольвенты по классическому методу резолюции.

Пример 1.78. Доказать истинность суждения:

$\exists_x(P_1(x) \& \forall_y(P_2(y) \rightarrow P_3(x, y))),$

$\forall_x(P_1(x) \rightarrow \forall_y(P_4(y) \rightarrow \neg P_3(x, y))),$

$\forall_y(P_2(y) \rightarrow \neg P_4(y)).$

Для этого суждения уже найдено множество дизъюнктов:

$K = \{P_1(a), (\neg P_2(y) \vee P_3(a, y)), (\neg P_1(x) \vee \neg P_4(y) \vee \neg P_3(x, y)), P_2(b), P_4(b)\}$

Первый и два последних дизъюнкта – факты, второй дизъюнкт – правило, отрицание третьего – цель.

Программа:

predicates

$p_1(\text{symbol})$

$p_2(\text{symbol})$

$p_3(\text{symbol}, \text{symbol})$

$p_4(\text{symbol})$

clauses

$p_1(a)$ .

$p_3(a, Y) : \neg p_2(Y)$ .

p2(b).  
 p4(b).  
 goal  
 p1(X),  
 p4(Y),  
 p3(X,Y).

С помощью Test Goal получим следующий результат:  
 X=a, Y=b

1 Solution,

что означает, что формула  $(P_1(a) \& P_4(b) \& P_3(a,b))$  логически следует из множества фактов и правил программы, следовательно, множество посылок и отрицание заключения несовместимы. Это эквивалентно получению пустой резольвенты по классическому методу резолюции.

Пример 1.79. Доказать истинность суждения:

$$\begin{aligned}
 & \forall_x (P_3(x) \& \neg P_1(x) \rightarrow \exists_y (P_5(y) \& P_4(x, y))), \\
 & \exists_x (P_2(x) \& P_3(x) \& \forall_y (P_4(x, y) \rightarrow P_2(y))), \\
 & \forall_x (P_2(x) \rightarrow \neg P_1(x)) . \\
 & \exists_x (P_5(x) \& P_2(x)).
 \end{aligned}$$

Для этого суждения уже найдено множество дизъюнктов:

$$\begin{aligned}
 K = \{ & (\neg P_3(x) \vee P_1(x) \vee P_5(f(x))), (\neg P_3(x) \vee P_1(x) \vee P_4(x, f(x))), P_2(a), P_3(a), \\
 & (\neg P_4(a, y) \vee P_2(y)), (\neg P_1(x) \vee \neg P_2(x)), (\neg P_5(x) \vee \neg P_2(x)) \}
 \end{aligned}$$

- два первых дизъюнкта – правила:  
 $(P_3(x) \& \neg P_1(x) \rightarrow P_5(f(x)))$  и  $(P_3(x) \& \neg P_1(x) \rightarrow P_4(x, f(x)))$ .
- третий и четвертый дизъюнкты – факты.
- пятый дизъюнкт  $(\neg P_4(a, y) \vee P_2(y)) \equiv (P_4(a, y) \rightarrow P_2(y))$  - правило.
- шестой дизъюнкт  $(\neg P_1(x) \vee \neg P_2(x)) \equiv (\neg P_1(x) \leftarrow P_2(x))$  - правило

- в качестве цели возьмем отрицание седьмого дизъюнк-  
та  $\neg(\neg P_5(x) \vee \neg P_2(x)) \equiv (P_5(x) \& P_2(x))$ .

Программа:

domains

arg=a;f(arg)

predicates

neP1(arg)

p2(arg)

p3(arg)

p4(arg,arg)

p5(arg)

clauses

p5(f(X)):-p3(X),neP1(X).

p4(X,f(X)):-p3(X),neP1(X).

p2(a).

p2(Y):-p4(a,Y).

p3(a).

neP1(X):-p2(X).

goal

p5(X),

p2(X).

С помощью Test Goal получим следующий результат:

X=f(a)

1 Solution,

что означает, что формула  $(P_5(f(a)) \& P_2(f(a)))$  логически сле-  
дует из множества фактов и правил программы, следова-  
тельно, множество посылок и отрицание заключения несо-  
вместимы. Это эквивалентно получению пустой резоль-  
венты по классическому методу резолюции.

В заключении можно отметить, что в стандарте SQL/92 реляционную СУБД расширили до дедуктивной системы. При этом реляционная часть – база данных - есть экстенциональная часть дедуктивной системы, а часть, содержащая описание правил – база знаний - есть интенциональная часть дедуктивной системы\* .

---

\* см. <http://belani.narod.ru/1/postrel.htm>

## Вопросы и задачи

1.2.1. а) «все судьи - юристы, но не все юристы – судьи»,

б) «Судья, являющийся родственником потерпевшего, не может участвовать в рассмотрении дела. Судья X - родственник потерпевшего. Следовательно, судья X не может участвовать в рассмотрении дела»,

в) «К уголовной ответственности привлекаются лица, совершившие тайное похищение личного имущества граждан. Обвиняемый X не совершал тайного похищения личного имущества граждан. Следовательно, обвиняемый X не может быть привлечен к уголовной ответственности»,

г) «Если иск предъявлен недееспособным лицом, то суд оставляет иск без рассмотрения. Иск предъявлен недееспособным лицом. Следовательно, суд оставляет иск без рассмотрения»,

д) «Ни один человек не может быть вполне беспристрастным. Каждый юрист – человек. Следовательно, ни один юрист не может быть не может быть вполне беспристрастным».

1.2.2. Привести формулу к виду ПНФ и ССФ:

а)  $\forall x(\neg A(x) \rightarrow (A(x) \rightarrow \exists y(B(y))))$ ,

б)  $\forall x(A(x) \rightarrow B(x)) \rightarrow \forall y((C(y) \rightarrow A(x)) \rightarrow (C(y) \rightarrow B(x)))$ ,

в)  $\forall x(A(x) \rightarrow B(x)) \& \exists y(B(x) \rightarrow C(y)) \& \exists z(C(y) \rightarrow D(z))$ ,

г)  $\forall x(A(x) \rightarrow \exists y(B(y)) \& \forall y(C(y) \rightarrow \exists x D(x)) \rightarrow (A(x) \& C(y)) \& D(y))$

д)  $(\forall x(A(x) \rightarrow B(x)) \& \forall z(C(z) \rightarrow A(x))) \rightarrow \exists y(C(z) \rightarrow B(y))$

f)

$$\forall_x(A(x) \rightarrow B(y) \& A(x) \rightarrow \forall_y(B(y) \rightarrow C(z))) \rightarrow (A(x) \rightarrow \exists_z(C(z))) /$$

1.2.4. Какие из нижеприведенных формул являются тождественно истинными:

a)  $\exists_x(P_1(x)) \& \exists_x(P_2(x)) \rightarrow \exists_x(P_1(x) \& P_2(x)),$

b)  $\forall_y(P_1(y)) \& \forall_y(P_2(y)) \rightarrow \forall_y(P_1(y) \& P_2(y)),$

c)  $\exists_x(P_1(x) \vee P_2(x)) \rightarrow (\exists_x(P_1(x)) \vee \exists_x(P_2(x))),$

d)  $\forall_y(P_1(y) \vee P_2(y)) \rightarrow (\forall_y(P_1(y)) \vee \forall_y(P_2(y))).$

1.2.5. Доказать выводимость заключения методом дедукции и по принципу резолюции:

a) 
$$\frac{\forall_x(P_1(x) \rightarrow \neg P_2(x)), \forall_x(P_3(x) \rightarrow P_1(x))}{\forall_x(P_3(x) \rightarrow \neg P_2(x))}.$$

b) 
$$\frac{\forall_x(P_1(x, y) \rightarrow P_2(x) \& P_3(x)), \exists_x(P_1(x, y) \& P_4(x))}{\exists_x(P_4(x) \& P_3(x))}.$$

c) 
$$\frac{\exists_x(P_1(x) \& \forall_y(P_2(x) \rightarrow P_3(x, y))), \forall_x(P_1(x) \rightarrow \forall_y(P_4(y) \rightarrow \neg P_3(x, y)))}{\forall_y(P_2(y) \rightarrow \neg P_4(y))}.$$

## Расчетно-графическая работа

- 1) Преобразовать формулу к виду ПНФ и ССФ,
- 2) сформировать множество дизъюнктов К,
- 3) выполнить унификацию дизъюнктов.

вари-	Формула
1	$\forall x.(A(x) \rightarrow B(v)) \& \forall x.(A(x) \rightarrow (B(v) \rightarrow C(z))) \rightarrow (A(x)$
2	$\forall x.(B(x) \rightarrow \exists z.(A(z))) \& \exists z.(A(z) \rightarrow C(v)) \rightarrow (\neg C(v) \&$
3	$\forall x.(A(x) \rightarrow \exists v.(B(v))) \rightarrow \exists x.(\neg A(x) \vee \neg C(z) \vee B(v))$
4	$\forall x.(A(x) \rightarrow \exists z.(C(z))) \& \forall x.(C(z) \rightarrow B(v)) \rightarrow (A(x) \rightarrow B$
5	$\forall x.(A(x) \rightarrow B(v)) \& \forall x.(A(x) \rightarrow (B(v) \rightarrow C(z))) \rightarrow \exists x.(A$
6	$\forall x.(A(x) \rightarrow \exists v.(B(v) \rightarrow C(z))) \rightarrow \forall x.(A(x) \& B(v) \rightarrow C(z)$
7	$\forall x.(A(x) \rightarrow B(z)) \& \forall x.(C(v) \rightarrow A(x)) \rightarrow \exists x.(C(v) \rightarrow B(z)$
8	$\forall x.((A(x) \rightarrow B(v)) \rightarrow \forall x.((C(v) \vee A(x))) \rightarrow (C(v) \vee \exists x.($
9	$\forall x.(A(x) \rightarrow B(v)) \& \forall x.(A(x) \rightarrow (B(v) \rightarrow C(z))) \rightarrow (A(x)$
10	$\forall x.(A(x) \rightarrow B(v) \& A(x) \rightarrow \forall x.(B(v) \rightarrow C(z))) \rightarrow (A(x)$
11	$\forall x.(A(x) \rightarrow \exists v.(B(v) \rightarrow C(z))) \rightarrow \forall x.(B(v) \rightarrow (A(x) \rightarrow C$
12	$(\forall x.(A(x)) \rightarrow \exists z.(B(z))) \rightarrow \forall x.((B(x) \rightarrow C(z)) \rightarrow (A(x)$
13	$(\exists x.(\neg A(x)) \rightarrow \forall x.(\neg B(z))) \rightarrow (\neg B(x) \vee A(x))$
14	$(\forall x.(A(x)) \rightarrow \forall x.(B(v))) \rightarrow \exists x.(C(v) \& A(x) \rightarrow C(v) \& B$
15	$\forall x.(\neg A(x) \rightarrow \exists v.(B(v))) \rightarrow (\neg B(v) \rightarrow A(x))$
16	$\forall x.(B(x) \rightarrow \exists z.(A(z))) \& \exists z.(A(z) \rightarrow C(v)) \rightarrow (\neg C(v) \&$
17	$\forall x.(A(x) \rightarrow \exists z.(C(z))) \& \forall x.(C(z) \rightarrow B(v)) \rightarrow (A(x) \rightarrow B$
18	$\forall x.(\neg A(x) \rightarrow \exists v.(\neg B(v))) \rightarrow (B(v) \rightarrow A(x))$
19	$\forall x.(A(x) \rightarrow B(x)) \& \exists x.(B(x) \rightarrow C(v)) \rightarrow \exists x.(C(v) \rightarrow D(z)$
20	$\forall x.(A(x) \rightarrow B(x)) \& \forall x.(C(z) \rightarrow A(x)) \rightarrow \exists x.(C(z) \rightarrow B(v)$
21	$\forall x.(B(x) \rightarrow \forall v.(A(v))) \& \forall x.(B(v) \rightarrow (A(x) \rightarrow C(z))) \rightarrow$
22	$\forall x.(A(x) \rightarrow B(v) \& A(x) \rightarrow \forall x.(B(v) \rightarrow C(z))) \rightarrow (A(x)$
23	$\forall x.(A(x) \rightarrow B(x)) \rightarrow (\forall x.(C(v) \rightarrow A(x)) \rightarrow \exists x.(C(z) \rightarrow B$

24	$\forall x.(B(x) \rightarrow A(x)) \& (B(x) \rightarrow \forall y.(A(y) \rightarrow C(z))) \rightarrow \exists z.(C$
25	$(\exists x.(A(x) \rightarrow B(z)) \rightarrow \exists y.(C(y) \vee A(x))) \rightarrow \forall z.(C(y) \vee B(z)$
2	$(\forall x.(B(x) \rightarrow \exists y.(A(y))) \& (A(y) \rightarrow \exists z.C(z))) \rightarrow (\neg A(x)$
2	$((\forall x.(A(x) \rightarrow \exists y.(B(y))) \rightarrow \exists z.(A(x) \vee C(z))) \rightarrow (B(x) \vee$
2	$\exists x.(A(x) \rightarrow \forall y.(B(y))) \& (\neg A(x) \rightarrow \forall y.(B(y))) \rightarrow B(y)$
2	$\forall x.(A(x) \rightarrow \exists y.(B(y))) \& (\neg A(x) \rightarrow B(x)) \rightarrow B(x)$
3	$\forall x.(A(x) \rightarrow B(x)) \& \forall z.(C(z) \rightarrow A(x)) \rightarrow \exists z.(C(z) \rightarrow B($
3	$(\exists x.(B(x) \rightarrow \forall y.(A(y))) \& (\neg B(x) \rightarrow A(y)) \rightarrow A(z)$
3	$(\forall x.(B(x) \rightarrow \exists y.(C(z))) \rightarrow (A(y) \& B(x) \rightarrow A(y) \& C(z)$
3	$\exists x.(A(x) \rightarrow B(y)) \rightarrow \forall z.\forall x.((C(z) \rightarrow A(x)) \rightarrow (C(z) \rightarrow B$
3	$(\forall x.(A(x) \rightarrow \exists y.(C(z))) \& \forall z.(C(z) \rightarrow B(y)) \rightarrow (A(x) \rightarrow$
3	$\forall x.(A(x) \rightarrow \exists y.(B(y))) \& \forall z.(C(z) \rightarrow \exists y.D(x)) \rightarrow (A(x)$
3	$\forall x.(A(x) \rightarrow B(y)) \& \forall z.(C(z) \rightarrow A(x)) \rightarrow \exists z.(C(z) \rightarrow B($
3	$(\forall x.(A(x) \rightarrow \exists y.(B(z))) \rightarrow \exists z.(A(x) \vee C(y))) \rightarrow (B(z) \vee C$
3	$\forall x.(B(x) \rightarrow \forall y.(A(y))) \& \forall z.(B(x) \rightarrow (A(y) \rightarrow C(z))) \rightarrow$
3	$\forall x.(B(x) \rightarrow A(y)) \& (B(x) \rightarrow \forall y.(A(y) \rightarrow C(z))) \rightarrow \exists z.(B$
4	$\forall x.(A(x) \rightarrow B(x)) \rightarrow \forall y.((C(y) \rightarrow A(x)) \rightarrow (C(y) \rightarrow B(x$
4	$\forall x.(A(x) \rightarrow B(y)) \& (A(x) \rightarrow \forall z.(B(y) \rightarrow C(z))) \rightarrow (A(x$
4	$(\forall x.(B(x) \rightarrow \exists y.(C(z))) \rightarrow (A(y) \& B(x) \rightarrow A(y) \& C(z)$
4	$\exists x.(A(x) \rightarrow B(z)) \rightarrow \exists y.((C(y) \vee A(x)) \rightarrow \forall z.(C(y) \vee B(z)$
4	$\forall x.A(x) \rightarrow B(y) \& \forall z.(C(z) \rightarrow A(x)) \rightarrow \exists z.(C(z) \rightarrow B(y$
4	$\forall x.(A(x) \rightarrow B(x)) \& \exists y.(B(x) \rightarrow C(y)) \& \exists z.(C(y) \rightarrow D(z$
4	$(\forall x.(A(x) \rightarrow \exists y.(C(z))) \& \forall z.(C(z) \rightarrow B(y)) \rightarrow (A(x) \rightarrow$
4	$\forall x.(A(x) \rightarrow B(y)) \& A(x) \rightarrow \forall z.(B(y) \rightarrow C(z)) \rightarrow (A(x)$
4	$(\forall x.(B(x) \rightarrow \exists y.(A(y))) \& \exists z.(A(x) \rightarrow C(y)) \rightarrow \neg C(y)$
4	$(\forall x.(A(x) \rightarrow \exists y.(C(z))) \& \forall z.(C(z) \rightarrow B(y)) \rightarrow (A(x) \rightarrow$
5	$\forall x.(A(x) \rightarrow B(y)) \& \forall z.(A(x) \rightarrow (B(y) \rightarrow C(z))) \rightarrow \exists z.(A$

## 1.3. Логика реляционная

Двухмерная таблица - удобная форма представления информации в различных сферах человеческой деятельности. Именами столбцов такой таблицы являются имена свойств объектов, чаще всего называемых *атрибутами*, а строками, называемыми *кортежами* - цепочки значений атрибутов. Область определения атрибута называют *доменом*. Число строк таблицы называют её *мощностью*.

Таблица 1.12

Дисциплина	Лекции (ч)	Лаб_занятия (ч)	Практ_занятия (ч)	Отчетность (зач., экз.)
физика	34	34	17	экз
информатика	51	34	0	зач.
...	...	...	...	...

Таблица 1.12 отражает связи между наименованиями учебных дисциплин, видом и числом часов аудиторных занятий и формой отчетности. Верхнюю строку таблицы называют «*шапкой таблицы*», а остальные строки «*телом таблицы*». Именами атрибутов являются «Дисциплина», «Лекции», «Лаб\_занятия», «Практ\_занятия», «Отчетность». Доменом атрибута «Дисциплина» является <слово>= {'физика', 'информатика',...}, а его тип - CHAR (строковое). Доменом атрибутов «Лекции», «Лаб\_занятия», «Практ\_занятия» являются <целое число>= <часы>{'', ''<часы>}, а тип домена – INTEGER. Доменом атрибута «Отчетность» – <слово>= {'зач.', 'экз.'}, а его тип - CHAR (строковое).

Множество совместимых кортежей называют *отношением*, а множество отношений для определенной области деятельности называют *базой данных*. Кортежи называют *совместимыми*, если они имеют одинаковую длину (или ранг), одинаковые имена компонент и одинаковый их порядок в кортеже.

Отношение можно рассматривать как *файл* определенного типа. Такой файл состоит из последовательности *записей*, по одной на каждый кортеж, причем в файле не должно быть одинаковых записей. Все записи файла должны иметь одинаковое число *полей*, так как кортежи должны быть совместимыми.

Между таблицей, отношением и файлом есть соответствие:

<u>ТАБЛИЦА</u>	$\longleftrightarrow$	<u>ОТНОШЕНИЕ</u>	$\longleftrightarrow$	<u>ФАЙЛ</u>
строка		$\longleftrightarrow$		кортеж
$\longleftrightarrow$				запись
имя столбца	$\longleftrightarrow$	имя атрибута	$\longleftrightarrow$	имя поля
тип атрибута	$\longleftrightarrow$	тип домена	$\longleftrightarrow$	тип поля

В одноименных полях различных записей должна храниться информация одного и того же типа, соответствующего заданному типу атрибута. Это является одним из основных требований реляционной модели: нормализация отношения

Таким образом, если дано множество атрибутов  $A = \{A_1, A_2, \dots, A_n\}$  и множество доменов  $D = \{D_1, D_2, \dots, D_m\}$ , то кортеж отношения есть  $t = (d_1, d_2, \dots, d_n)$  где  $d_i \in D_j$ .

Так как отношение есть множество совместимых кортежей, то  $r = \{t \mid t = (d_1, d_2, \dots, d_n), d_i \in D_j\}$ .

Отношение, заданное на множестве упорядоченных кортежей, есть подмножество  $n$ -арного прямого произведения доменов, т.е.

$$r = \{t \mid t = (d_1, d_2, \dots, d_n), d_i \in D\} \subseteq \otimes^n D.$$

Отношение на множестве упорядоченных кортежей задают *схемой отношения* с указанием имени отношения, числа и порядка следования атрибутов в кортеже и обозначают  $\mathbf{rel}(r) = (A_1, A_2, \dots, A_n)$ .

Например,  $\mathbf{rel}(\text{уч\_план\_1}) = (\text{дисциплина}, \text{лекции\_}(ч), \text{лаб\_занятия\_}(ч), \text{практ\_занятия\_}(ч), \text{отчетность\_}(зач., \text{экз.}))$ .

Множество схем отношений, используемых в реляционной базе данных, называют *схемой реляционной базы данных* и обозначают  $REL(R) = \{\mathbf{rel}(r)\}$ .

Схемы отношений базы данных имеют различную длину и различные наборы атрибутов. Поэтому чаще всего в каждом поле записи указывают <имя атрибута> = <значение>. Наличие в поле записи имени атрибута снимает ограничение на упорядоченность полей записи. При этом оказываются совместимыми записи, заданные одинаковым набором одинаковых имен полей, но имеющих различный порядок следования атрибутов.

Например, запись файла «уч\_план\_1» может быть (дисциплина='физика', лекции='34', лаб.занятия='34', практ.занятия='17', отчетность='экз.') или такой (практ.занятия='17', дисциплина='физика', отчетность='экз', лаб.занятия='34', лекции='34').

Для каждого отношения должен быть задан ключ. **Ключ** – это один или несколько атрибутов, выделяющих единственный кортеж отношения или единственную за-

пись файла. Так, например, ключом учебного плана может быть атрибут «дисциплина».

Над отношениями оказалось возможным осуществлять алгебраические и логические операции. Так математическая логика нашла применение в управлении реляционными базами данных.

При изложении основ реляционной алгебры и реляционного исчисления будем придерживаться следующих ограничений:

- все атрибуты кортежа должны быть элементарными (или каждое поле записи должно иметь один тип: INTEGER (целые), REAL (вещественные), CHAR (символьные) или BOOLEAN (логические) и не должны быть функционально связаны между собой,
- все кортежи должны быть упорядоченными и иметь одинаковое число компонент в одном отношении (или все записи в файле должны быть упорядочены и иметь одинаковое число полей),
- каждое отношение должно иметь ключ, в роли которого выступают один или несколько атрибутов и каждое отношение не должно содержать по одному ключу двух или более одинаковых кортежей (или файл не должен содержать двух или более одинаковых записей),
- никакие два столбца таблицы не должны иметь совпадающие имена атрибутов, но их значения могут принадлежать одному домену.

## 1.3.1. Реляционная алгебра\*

Пусть дано множество отношений  $R = \{r_1, r_2, \dots\}$ ,  
 где  $r_i = \{t_1, t_2, \dots\} \mid t_i = (d_1, d_2, \dots), d_i \in D\}$  - отношение,  $t = (d_1, d_2, \dots)$  - кортеж,

$d_i \in D_i$  - значение атрибута  $A_i$ ,  $D_i$  - домен атрибута  $A_i$ ,

множество доменов  $D = \{D_1, D_2, \dots\}$ ,

множество атрибутов  $A = \{A_1, A_2, \dots\}$ ,

множество схем отношений  $Rel(R) = \{rel(r_1), rel(r_2), \dots\}$ ?

где  $rel(r_i) = (A_1, A_2, \dots, A_n)$  - схема отношения  $r_i$

и множество операторов  $\Sigma = \{\cup, \cap, \setminus, \otimes, \neg, \delta_B(r), \pi_{rel(r)}, \succ, \succ^{\theta}\}$ ,

где

$\cup$  - оператор объединения отношений,

$\cap$  - оператор пересечения отношений,

$\setminus$  - оператор разности отношений,

$\otimes$  - оператор прямого произведения отношений,

$\neg$  - оператор дополнения отношения,

$\delta_B(r)$  - оператор выбора кортежа по условию  $B$ ,

$\pi_{rel(r)}$  - оператор проекции отношения на новую схему  $rel$ ,

$\succ$  - оператор естественного соединения отношений,

$\succ^{\theta}$  - оператор  $\theta$  - соединения отношений,

$:$  - оператор деления отношений.

Совокупность  $R$  и  $\Sigma$  представляют реляционную алгебру

$$A_{\text{рел}} = \langle R, \Sigma \rangle,$$

где  $R$  - носитель алгебры,  $\Sigma$  - алгебраические операторы.

Задание логических операторов  $\Psi = \{\&, \vee, \neg\}$  и опера-

---

\* по материалам [14]



шения  $r$  кортежи по условию  $B$ , и создавать новое отношение. Условия выбора  $B$  описывают с помощью арифметических операторов сравнения  $\theta = \{=, \neq, >, \geq, <, \leq\}$  и/или логических операторов  $\Psi = \{\&, \vee, \neg\}$ .

Например,  $B=(A_i=k_i)$  означает «выбрать в заданном отношении кортежи, для которых атрибут  $A_i$  имеет значение  $d_i=k_i$ ». Также можно описать  $B=(A_i \neq k_i)$ ,  $B=(A_i > k_i)$ ,  $B=(A_i \geq k_i)$ , ... Условие  $B=((A_i, \dots, A_j, \dots, A_k)=(k_i, \dots, k_j, \dots, k_k))$  означает «выбрать в заданном отношении кортежи, для которых атрибут  $A_i$  имеет значение  $k_i$ , атрибут  $A_j$  имеет значение  $k_j$ , атрибут  $A_k$  имеет значение  $k_k$ ». Если условия заданы логическими операторами, то  $B=(B_i \& B_j)$  означает «выполнить условие  $B_i$  и  $B_j$ », а  $B=(B_i \vee B_j)$  – «выполнить условие  $B_i$  или  $B_j$ » и т. п.

Результатом исполнения операции выбора будет новое отношение  $r' = \delta_B(r) = \{t' \mid t' \subseteq r, B, \text{rel}(r') = \text{rel}(r)\} \subseteq r$ .

**Пример 1.80.** Выбрать кортежи отношения  $r_1$  по значению ключа  $A_1=a_2$ , т.е.  $r'_1 = \{t' \mid B := (A_1=a_2)\}$ ,  $r_2$  – по значению атрибута  $A_3=1$ , т.е.  $r'_2 = \{t' \mid B := (A_3=1)\}$ ,  $r_5$  – по значениям атрибутов  $\{A_1=a_1, A_2=b_1, A_3=1\}$ , т.е.  $r'_5 = \{t' \mid B := (A_1=a_1) \& (A_2=b_1) \& (A_3=1)\}$ .

В табл. 1.12а. - 1.12с. соответственно показаны результаты исполнения этих операций.

Таблица 1.12а	Таблица 1.12б	Таблица 1.12с
$r'_1$	$r'_2$	$r'_5$
$A_1$ $A_2$ $A_3$	$A_1$ $A_2$ $A_3$	$A_1$ $A_2$ $A_3$ $A_4$ $A_5$ $A_6$
$a_2$ $b_2$ 3	$a_2$ $b_3$ 1	$a_1$ $b_1$ 1 $c_2$ $d_3$ 1
	$a_1$ $b_1$ 1	$a_1$ $b_1$ 1 $c_3$ $d_3$ 3
		$a_1$ $b_1$ 1 $c_1$ $d_1$ 2

Оператор выбора как бы разрезает таблицу на отдельные строки, удаляет строки, не удовлетворяющие заданным условиям? и склеивает новую таблицу.

В программировании это будет записано так:

$r' = \text{SELECT}$  (отношение, УСЛОВИЯ).

Для данного примера это будет:

$r'_1 = \text{SELECT}(r_1, A_1 = 'a_2')$ ,

$r'_2 = \text{SELECT}(r_2, A_3 = 1)$ ,

$r'_5 = \text{SELECT}(r_5, A_1 = 'a_1' \text{ AND } A_2 = 'b_1' \text{ AND } A_3 = 1)$ .

Значения атрибутов заключены в апострофы. Это означает, что строка символов, заключенная в апострофы является значением поля записи. Данное правило относится к любым одиночным символам и строкам, представляющим значение атрибута. Это делается для того, чтобы отличить значение от имени атрибута, переменных параметров и числовых констант.

При программировании логические связки  $\{\&, \vee, \neg\}$  замещаются операторами AND, OR и NOT соответственно.

Пример 1.81. Выбрать кортежи отношения  $r_1$  по значению атрибута  $A_3 = \neg 3$ , т.е.  $r'_1 = \{t' \mid V := (A_3 = \neg 3)\}$ ,  $r_2$  - по значению атрибутов  $A_3 = 1$  или  $A_2 = b_1$ , т.е.  $r'_2 = \{t' \mid V := (A_3 = 1 \vee A_2 = b_1)\}$ ,  $r_5$  - по значению атрибутов ( $A_1 = a_1$ ,  $A_2 = b_1$ ,  $A_3 = 1$  и  $A_4 \neq c_2$ ), т.е.  $r'_5 = \{t' \mid V := ((A_1 = a_1) \& (A_2 = b_1) \& (A_3 = 1) \& (A_4 \neq c_2))\}$ .

В программировании исполнение этих операций будет описано так:

$r'_1 = \text{SELECT}(r_1, A_3 = \text{NOT } 3)$  (см. табл. 1.13а),

$r'_2 = \text{SELECT}(r_2, A_3 = 1 \text{ OR } A_2 = 'b_1')$  (см. табл. 1.13б),

$r'_5 = \text{SELECT}(r_5, A_1='a_1' \text{ AND } A_2='b_1' \text{ AND } A_3=1 \text{ AND } A_4=\text{NOT}'c_2')$  (см. табл. 1.13с).

Таблица 1.13а	Таблица 1.13b	Таблица 1.13с
$r'_1$   <u>A<sub>1</sub></u> A <sub>2</sub> A <sub>3</sub>   a <sub>1</sub> b <sub>1</sub> 1   a <sub>3</sub> b <sub>3</sub> 2	$r'_2$   <u>A<sub>1</sub></u> <u>A<sub>2</sub></u> A <sub>3</sub>   a <sub>2</sub> b <sub>3</sub> 1   a <sub>1</sub> b <sub>1</sub> 1	$r_5$   <u>A<sub>1</sub></u> A <sub>2</sub> A <sub>3</sub> A <sub>4</sub> A <sub>5</sub> A <sub>6</sub>   a <sub>1</sub> b <sub>1</sub> 1 c <sub>3</sub> d <sub>3</sub> 3   a <sub>1</sub> b <sub>1</sub> 1 c <sub>1</sub> d <sub>1</sub> 2

**Оператор проекции  $\pi_{rel}(r)$**  позволяет формировать из данного отношения  $r$  со схемой  $rel(r)=(A_1, A_2, \dots, A_n)$  новое отношение  $r'$  со схемой  $rel(r')=(A_i, A_j, \dots, A_k)$ , где  $1 \leq i, j, k \leq n$ , путем удаления и/или переупорядочения атрибутов.

Результатом выполнения этой операции будет новое отношение, удовлетворяющее новой схеме  $r' = \pi_{rel}(r) = \{ t' \mid |rel(r')| \leq |rel(r)| \}$ .

Число атрибутов схемы формируемого отношения меньше или равно числу атрибутов схемы заданного отношения, т.е.  $|rel(r')| \leq |rel(r)|$ , а число кортежей формируемого отношения равно числу кортежей заданного отношения, т.е.  $|r'| = |r|$ .

**Пример 1.82.** Выбрать только ключи отношений  $r_1$ ,  $r_3$  и  $r_5$ , т.е.  $rel(r'_1) = (A_1)$ ,  $rel(r'_3) = (A_1, A_5)$ ,  $rel(r'_5) = (A_1, A_2, A_3)$ .

В программировании операцию проекции записывают так  $r' = \text{ПРОЕКТ}(\text{отношение}, \text{СПИСОК АТТРИБУТОВ})$ . Для данного примера это -  $r'_1 = \text{ПРОЕКТ}(r_1, A_1)$ ,  $r'_3 = \text{ПРОЕКТ}(r_3, A_1, A_5)$ ,  $r'_5 = \text{ПРОЕКТ}(r_5, A_1, A_2, A_3)$ .

В таблицах 1.14а.-1.14с. соответственно показаны результаты исполнения этих операций.

Таблица 1.14а      Таблица 1.14b      Таблица 1.14с

$r'_1$	$A_1$
	$a_1$
	$a_2$
	$a_3$
	$a_4$

$r'_3$	$A_1$	$A_5$
	$a_1$	$d_3$
	$a_2$	$d_1$
	$a_3$	$d_2$
	$a_1$	$d_1$

$r'_5$	$A_1$	$A_2$	$A_3$
	$a_1$	$b_1$	1
	$a_2$	$b_2$	3
	$a_1$	$b_1$	1
	$a_2$	$b_2$	3
	$a_1$	$b_1$	1
	$a_3$	$b_3$	2
	$a_3$	$b_3$	2
	$a_4$	$b_1$	3

**Оператор дополнения  $\neg r$ .** Для этого необходимо найти множество всех кортежей со схемой заданного отношения  $rel(r)$  на области определения  $D$ , т. е. найти число размещений всех возможных значений домена отношения  $|D|=n$  на число компонент схемы отношения  $rel(r)$ , т.е. найти  $dom(r) = (n)_k = n \cdot (n-1) \cdot (n-2) \cdot \dots \cdot (n-k+1)$  и удалить из этого множества кортежи заданного отношения  $r$ . Множество таких кортежей очень большое. Если хотя бы один атрибут отношения счетный, но не конечный, то  $|dom(r)| = \infty$ .

**Пример 1.83.** Если заданно отношение  $r_3$ , то  $D_1 = \{a_1, a_2, a_3\}$ ,  $D_4 = \{c_1, c_2\}$ ,  $D_5 = \{d_1, d_2, d_3\}$ , т. е. имеем  $n = |D| = |D_1 \cup D_4 \cup D_5| = 8$  и  $k = 3$ .

Следовательно, число кортежей дополнения равно  $|dom(\neg r_3)| = (8)_3 = 8 \cdot 7 \cdot 6 \cdot 5 \cdot 4 = 6720$ . Однако, если значения атрибутов ограничить пределами своих доменов, то  $|dom(r)| = |D_1| \cdot |D_2| \cdot \dots \cdot |D_m|$ . Например, для  $r_3$  имеем  $|dom(r)| = 3 \cdot 2 \cdot 3 = 18$ , а  $|\neg r| = 14$ . Для того чтобы получить кортежи отношения  $\neg r_3$ , следует из множества кортежей  $dom(r_3)$  удалить кор-

тежи  $r_3$ , т. е.  $\neg r_3 = \text{dom}(r_3) \setminus r_3$ . Таблицами 1.15a. и 1.15b. изображены  $\text{dom}(r_3)$  и  $\neg r_3$ .

Таблица 1.15a

$\text{dom}(r_3)$	$A_1$	$A_4$	$A_5$
	$a_1$	$c_1$	$d_1$
	$a_1$	$c_1$	$d_2$
	$a_1$	$c_1$	$d_3$
	$a_1$	$c_2$	$d_1$
	$a_1$	$c_2$	$d_2$
	$a_1$	$c_2$	$d_3$
	$a_2$	$c_1$	$d_1$
	$a_2$	$c_1$	$d_2$
	$a_2$	$c_1$	$d_3$
	$a_2$	$c_2$	$d_1$
	$a_2$	$c_2$	$d_2$
	$a_2$	$c_2$	$d_3$
	$a_3$	$c_1$	$d_1$
	$a_3$	$c_1$	$d_2$
	$a_3$	$c_1$	$d_3$
	$a_3$	$c_2$	$d_1$
	$a_3$	$c_2$	$d_2$
	$a_3$	$c_2$	$d_3$

Таблица 1.15b

$\neg r_3$	$A_1$	$A_4$	$A_5$
	$a_1$	$c_1$	$d_1$
	$a_1$	$c_1$	$d_2$
	$a_1$	$c_1$	$d_3$
	$a_1$	$c_2$	$d_2$
	$a_2$	$c_1$	$d_2$
	$a_2$	$c_1$	$d_3$
	$a_2$	$c_2$	$d_1$
	$a_2$	$c_2$	$d_2$
	$a_2$	$c_2$	$d_3$
	$a_3$	$c_1$	$d_1$
	$a_3$	$c_1$	$d_3$
	$a_3$	$c_2$	$d_1$
	$a_3$	$c_2$	$d_2$
	$a_3$	$c_2$	$d_3$

### 1.3.1.2. Бинарные операции

**Оператор объединения  $\cup(r_1, r_2)$**  двух отношений, имеющих

одинаковые схемы, формирует новое отношение  $r'$ , объединяя все кортежи первого и второго отношений. При этом одинаковые кортежи двух отношений замещаются

одним кортежем.

**Таблица 1.16**      **Пример 1.84.**  $r' = \{t' \mid t' = t_1 \in r_1$   
 $r_1 \cup r_2 \mid \underline{A_1} \underline{A_2} A_3 \text{ или } t' = t_2 \in r_2, \text{rel}(r') = \text{rel}(r_1) \cup \text{rel}(r_2)\}$ .

В нотации языков программирования эту операцию описывают так:  $r' = \text{UNION}(r_1, r_2)$ .

В таблице 1.16 приведены результаты исполнения операции.

$r_1 \cup r_2$	$\underline{A_1}$	$\underline{A_2}$	$A_3$
2			
	$a_1$	$b_1$	1
	$a_2$	$b_2$	3
	$a_3$	$b_3$	2
	$a_4$	$b_1$	3
	$a_2$	$b_3$	1
	$a_2$	$b_4$	2
	$a_1$	$b_2$	3

**Оператор прямого произведения  $\otimes(r_1, r_2)$**  формирует из двух отношений арности  $n_1$  и  $n_2$  новое отношение  $r'$  арности  $(n_1+n_2)$  и числом строк, равным  $(m_1 \cdot m_2)$ , где  $n_i$  – число атрибутов схемы  $i$ -го отношения,  $m_i$  – число кортежей  $i$ -го отношения. При этом первые  $n_1$  компонент кортежа отношения  $r'$  образованы отношением  $r_1$ , а последние  $n_2$  – отношением  $r_2$ . В результате формируется множество кортежей  $r' = \{t' = (t_1, t_2) \mid t_1 \in r_1, t_2 \in r_2, \text{rel}(r') = (\text{rel}(r_1), \text{rel}(r_2))\}$ . В программировании это записывают так:  $r' = \text{PRODUCT}(r_1, r_2)$ . Схема формируемого отношения есть  $\text{rel}(r') = (\text{rel}(r_1), \text{rel}(r_2))$ .

**Таблица 1.17**

**Пример 1.85.** Найти  $r' = (r_1 \otimes r_4)$ .

В таблице 1.17 приведены результаты исполнения этой операции.

$r_1 \otimes r_4$	A	A	A	A	A	A
$r_4$	1	2	3	4	5	6
	$a_1$	$b_1$	1	$c_2$	$d_3$	1
	$a_1$	$b_1$	1	$c_1$	$d_1$	2
	$a_1$	$b_1$	1	$c_2$	$d_2$	3
	$a_1$	$b_1$	1	$c_3$	$d_3$	2

a <sub>2</sub>	b <sub>2</sub>	3	c <sub>2</sub>	d <sub>3</sub>	1
a <sub>2</sub>	b <sub>2</sub>	3	c <sub>1</sub>	d <sub>1</sub>	2
a <sub>2</sub>	b <sub>2</sub>	3	c <sub>2</sub>	d <sub>2</sub>	3
a <sub>2</sub>	b <sub>2</sub>	3	c <sub>3</sub>	d <sub>3</sub>	2
a <sub>3</sub>	b <sub>3</sub>	2	c <sub>2</sub>	d <sub>3</sub>	1
a <sub>3</sub>	b <sub>3</sub>	2	c <sub>1</sub>	d <sub>1</sub>	2
a <sub>3</sub>	b <sub>3</sub>	2	c <sub>2</sub>	d <sub>2</sub>	3
a <sub>3</sub>	b <sub>3</sub>	2	c <sub>3</sub>	d <sub>3</sub>	2
a <sub>4</sub>	b <sub>1</sub>	3	c <sub>2</sub>	d <sub>3</sub>	1
a <sub>4</sub>	b <sub>1</sub>	3	c <sub>1</sub>	d <sub>1</sub>	2
a <sub>4</sub>	b <sub>1</sub>	3	c <sub>2</sub>	d <sub>2</sub>	3
a <sub>4</sub>	b <sub>1</sub>	3	c <sub>3</sub>	d <sub>3</sub>	2

**Оператор разности  $\setminus(r_1, r_2)$**  двух отношений, имеющих одинаковые схемы, формирует новое отношение  $r'$ , выбирая из первого отношения только те кортежи, которых нет во втором отношении. В результате выполнения этой операции формируется множество кортежей по правилу:  $r' = \{t' \mid t = t_1 \in r_1 \text{ и } t_1 \notin r_2, \text{ rel}(r_1) = \text{rel}(r_2)\}$ .

В программировании эту операцию записывают так:  $r' = \text{MINUS}(r_1, r_2)$  или  $r' = \text{DIFFERENCE}(r_1, r_2)$ .

**Пример 1.86.** Найти разность отношений  $r_1$  и  $r_2$ . **Таблица 1.18**

$r_1 \setminus r_2$	<u>A<sub>1</sub></u>	<u>A<sub>2</sub></u>	<u>A<sub>3</sub></u>
В таблице 1.18 приведены результаты исполнения этой операции.	a <sub>2</sub>	b <sub>2</sub>	3
	a <sub>3</sub>	b <sub>3</sub>	2
	a <sub>4</sub>	b <sub>1</sub>	3

**Оператор пересечения  $\cap(r_1, r_2)$**  двух отношений, имеющих одинаковые схемы, формирует новое отношение из кортежей первого и второго отношений, имеющих оди-

наковые значения всех одноименных атрибутов. В результате выполнения этой операции имеем

$r' = \{t' \mid t' = t_1 \in r_1 \text{ и } t' = t_2 \in r_2, \text{ rel}(r_1) = \text{rel}(r_2)\}$ . Операция пересечения может быть выполнена с помощью операторов разности  $r_1 \cap r_2 = r_1 \setminus (r_1 \setminus r_2)$ .

В программировании эту операцию записывают так:  
 $r' = \text{INTERSECT}(r_1, r_2)$ .

Пример 1.87. Найти  $r' = r_1 \cap r_2$ . **Таблица 1.19**

В таблице 1.19 приведены ре-

$r_1 \cap r_2$	$A_1$	$A_2$	$A_3$
	a	b	c

**Оператор естественного соединения**  $\bowtie(r_1, r_2)$  позволяет из отношений  $r_1$  и  $r_2$  арности  $n_1$  и  $n_2$ , имеющих одинаковые имена атрибутов, создавать новое отношение  $r'$ , схема которого есть объединение схем отношений, а кортежи нового отношения есть результат соединения кортежей первого и второго отношений при одинаковых значениях одноименных атрибутов. В результате выполнения этой операции в новом отношении остаются только те кортежи, одноименные атрибуты которых имеют одинаковое значение. Одноименные атрибуты кортежа замещаются одним атрибутом:

$r' = \{t' = (t_1, t_2) \mid t_1 \in r_1, t_2 \in r_2, \text{rel}(r') \subseteq (\text{rel}(r_1) \cup \text{rel}(r_2)) \& (r_1.A_i = r_2.A_i)\}$ .

В программировании эту операцию записывают так:  
 $r' = \text{JOIN}(r_1, r_2)$ .

Пример 1.88. Найти  $r' = r_1 \bowtie r_2$ .

В таблицах 1.20а. и 1.20б. приведены результаты  $r' = r_1 \bowtie r_2$  и  $r' = r_3 \bowtie r_4$ .

**Таблица 1.20а**

$r_1 ><$	$A_1$	$A_2$	$A_3$	$A_4$	$A_5$
	$a_1$	$b_1$	1	$c_2$	$d_2$
	$a_1$	$b_1$	1	$c_2$	$d_1$
	$a_2$	$b_2$	3	$c_1$	$d_1$
	$a_2$	$b_2$	2	$c_1$	$d_2$

**Таблица 1.20б**

$r_2 ><$	$A_1$	$A_4$	$A_5$	$A_6$
	$a_1$	$c_2$	$d_2$	1
	$a_2$	$c_1$	$d_1$	2

**Оператор  $\theta$ -соединения  $>^\theta<(r_1, r_2)$**  позволяет из отношений  $r_1$  и  $r_2$  арности  $n_1$  и  $n_2$  создавать новое отношение  $r'$  арности  $(n_1+n_2)$  при выполнении некоторого условия  $V=(A_{1i}\theta A_{2j})$ , где  $\theta=\{=, \neq, >, \geq, <, \leq\}$ . Первые  $n_1$  компонентов кортежа образованы кортежами, принадлежащими отношению  $r_1$ , а последние - кортежами, принадлежащими отношению  $r_2$ .

В результате исполнения этой операции формируется отношение:

$r'=\{t'=(t_1, t_2) \mid (t_1, t_2) \in (r_1 \otimes r_2), V=(A_{1i}\theta A_{2j}), \text{rel}(r')=(\text{rel}(r_1), \text{rel}(r_2))\}$ . В программировании эту операцию описывают так : $r'=\text{JOIN}(r_1, r_2, \text{УСЛОВИЕ})$ .

**Пример 1.89.** Найти  $>^\theta<(r_1, r_2)$  при условии  $V=(r_1.A_3 > r_2.A_6)$ .

**Таблица 1.21** Результаты этой операции

$r'$	$A_1$	$A_2$	$A_3$	$A_4$	$A_5$	$A_6$
	$a_1$	$b_1$	1	$c_1$	$d_1$	2
	$a_1$	$b_1$	1	$c_2$	$d_1$	3
	$a_1$	$b_1$	1	$c_1$	$d_2$	2
	$a_2$	$b_2$	2	$c_2$	$d_1$	3

представлены в таблице 1.21.

Частным случаем является *эквисоединение*, когда различные атрибуты различных отношений имеют одинаковое значение. В результате исполнения этой операции:  $r'=\{t'=(t_1, t_2) \mid (t_1, t_2) \in (r_1 \otimes r_2), \text{rel}(r')=(\text{rel}(r_1), \text{rel}(r_2)), (r_1.A_i=r_2.A_j)\}$ . Например,  $\langle \text{имя\_порт\_приписки} \rangle = \langle \text{имя\_порт\_назначения} \rangle$ .



### 1.3.1.3. Правила реляционной алгебры

Так как при выполнении алгебраических операций всегда формируется одно отношение, т. е.  $(\Sigma, \Psi, \theta): R \rightarrow r'$ , то последовательность их исполнения влияет на скорость создания  $r'$ . Поэтому при выполнении алгебраических операций следует соблюдать некоторые ограничения и правила:

- $r' = \delta_{B_1}(\delta_{B_2}(r)) = \delta_{B_2}(\delta_{B_1}(r))$  – операция выбора коммутативна,
- $r' = \delta_B(r_1 \cap r_2) = \delta_B(r_1) \cap \delta_B(r_2)$  – операция выбора над пересечением отношений равносильна пересечению операций выбора над каждым отношением, но  $\delta_B(r_1) \cap \delta_B(r_2)$  более рациональна по времени,
- $r' = \delta_B(r_1 \cup r_2) = \delta_B(r_1) \cup \delta_B(r_2)$  – операция выбора над объединением отношений равносильна объединению операций выбора над каждым отношением, но  $\delta_B(r_1) \cup \delta_B(r_2)$  более рациональна по времени,
- $r' = \delta_B(r_1 \setminus r_2) = \delta_B(r_1) \setminus \delta_B(r_2)$  – операция выбора над разностью отношений равносильна разности операций выбора над каждым отношением, но  $\delta_B(r_1) \setminus \delta_B(r_2)$  более рациональна по времени,
- $r' = \delta_B(r_1 \bowtie r_2) = \delta_B(r_1) \bowtie r_2$  – операция выбора над соединением отношений равносильна соединению одного отношения с операцией выбора над другим отношением, но  $\delta_B(r_1) \bowtie r_2$  более рациональна по времени, так как  $\delta_B(r_1 \bowtie r_2)$  обрабатывает все кортежи прямого произведения  $r_1$  и  $r_2$ ,
- $r' = \delta_B(\pi_{rel}(r_1)) = \pi_{rel}(\delta_B(r_1))$  – операция проекции коммутативна с операцией выбора,

- $r'=(r_1 \times r_2) = (r_2 \times r_1)$  - операция соединения коммутативна,
- $r'=(r_1 \times r_2) \times r_3 = r_1 \times (r_2 \times r_3)$  – операция соединения ассоциативна.

### 1.3.2. Реляционное исчисление\*

Если результатом алгебраической операции над отношениями является отношение, которому может быть присвоено имя, то результатом реляционного исчисления является множество кортежей  $t$ , каждый из которых определяется истинным значением некоторой логической функции  $F(t)$ .

На множестве кортежей  $t$  определим *переменные-кортежи* ( $x, y, z, \dots$ ) и *постоянные-кортежи* ( $a, b, c, \dots$ ).

Тогда *элементарная формула* отношения есть:

- если  $r$  - отношение, а  $t$  - кортеж, то  $r(t)$  –элементарная формула,
- если  $x$  и  $y$  – переменные-кортежи и дан оператор  $\theta$ , то  $(x(A_i)\theta y(A_j))$  - элементарная формула, где  $A_i$  и  $A_j$  – атрибуты кортежа.
- никаких других элементарных формул нет.

Введем также понятия *свободных переменных-кортежей* и *связанных переменных-кортежей*. Переменная-кортеж является связанной, если ей предшествует квантор по этой же переменной и свободной в противном случае. Тогда *формулы* реляционного исчисления с переменными-кортежами могут быть определены из элементарных формул так:

- любая элементарная формула есть формула, т.е.  $F= r(t)$

---

\* по материалам [14]

и

$$F = x(A_i) \theta y(A_j),$$

- если  $F_1$  и  $F_2$  - формулы, то  $(\neg F)$ ,  $(F_1 \vee F_2)$ ,  $(F_1 \& F_2)$  - также формулы,
- если  $x$  - переменная-кортеж и  $F(x)$  - формула, то  $\exists_x(F(x))$  и  $\forall_x(F(x))$  - также формулы.
- никаких иных формул нет.

Так  $r' = \{t' \mid t' \in t, F(t)\}$  есть множество свободных переменных-кортежей  $t'$  при значении формулы  $F(t)$  и для связанных переменных-кортежей.

### Операция выборки

$$r' = \{t' \mid \exists_x(r(x) \& (\text{<условия на атрибуты переменного-кортежа>}))\}.$$

Например,  $(x(A_i) \theta k_i)$ ,  $(x(A_i) \theta k_i) \text{ and } (x(A_s) \theta k_s)$ ,  $(x(A_i) \theta k_i) \text{ or } (x(A_s) \theta k_s)$ ,  $(x(A_i) \theta x(A_s))$  или  $A_j := f(A_s)$ .

Квантор существования выбирает такие переменные-кортежи, для которых формула (<условия на атрибуты переменного-кортежа>) имеет значение истины.

### Операция проекции

$$r' = \{t' \mid \forall_x(r(x) \& (\text{<порядок следования атрибутов>}))\}.$$

Квантор всеобщности использует в отношении-результате все кортежи исходного отношения, но не все его компоненты-атрибуты. Эта операция позволяет также переупорядочить расположение атрибутов в кортеже.

### Операция объединения

$$r' = \{t' \mid \forall_x \forall_y(r_1(x) \& r_2(y) \& ((t' = x) \vee (t' = y)))\}.$$

Квантор всеобщности объединяет в отношении  $r'$  все кортежи отношений  $r_1(x)$  и  $r_2(y)$ . Если атрибуты кортежей

неупорядочены, то атрибуты кортежа второго отношения должны быть упорядочены также, как у кортежей первого отношения (см. операцию проекции).

#### **Операция разности**

$$r' = \{t' \mid \exists_x \exists_y (r_1(x) \& r_2(y) \& ((t'=x) \& \neg(t'=y)))\}.$$

Кванторы существования накладывают условия для извлечения только таких кортежей первого отношения, которых нет во втором.

#### **Операция пересечения**

$$r' = \{t' \mid \exists_x \exists_y (r_1(x) \& r_2(y) \& (t'=x) \& (t'=y))\}.$$

Кванторы существования позволяют выбрать одинаковые кортежи в первом и втором отношениях.

#### **Операция прямого произведения**

$$r' = \{t' \mid \forall_x \forall_y (r_1(x) \& r_2(y) \& (t'[1]=x[1]) \& (t'[2]=x[2]) \& \dots \& (t'[n_1]=x[n_1]) \& (t'[n_1+1]=y[1]) \& (t'[n_1+2]=y[2]) \& \dots \& (t'[n_1+n_2]=y[n_2]))\}.$$

Кванторы всеобщности позволяют приписать каждый кортеж второго отношения к каждому кортежу первого отношения (операция конкатенации). В каждом поле формируемого отношения должно быть указано (<имя\_отношения>”.”<имя\_атрибута>).

#### **Операция естественного соединения**

$$r' = \{t' \mid \exists_x \exists_y (r_1(x) \& r_2(y) \& (<отношение_1>”.”<атрибут_переменного_кортежа><равенство><отношение_2>”.”<атрибут_переменного_кортежа>) \& (t'[1]=x[1]) \& \dots \& (t'[n_1]=x[n_1]) \& \dots \& (t'[n_1+1]=y[$$

$1]) \& (t'[n_1+2] = y[2]) \& \dots \& (t'[n_1+n_2-1] = y[n_2]) \}}.$

Кванторы существования обусловлены выбором переменных-кортежей, имеющих одинаковые значения одноименных атрибутов двух отношений.

### Операция $\theta$ -соединения

$r' = \{t' | \exists_x \exists_y (r_1(x) \& r_2(y) \& \langle \text{отношение\_1} \rangle \text{''} \langle \text{атрибут\_переменного-кортежа} \rangle \langle \text{оператор\_сравнения} \rangle \langle \text{отношение\_2} \rangle \text{''} \text{''} \langle \text{атрибут\_переменного-кортежа} \rangle) \& (t'[1] = x[1]) \& \dots \& (t'[i] = x[i]) \& \dots \& (t'[n_1] = x[n_1]) \& (t'[n_1+1] = y[1]) \& \dots \& (t'[n_1+j] = y[j]) \& \dots \& (t'[n_1+n_2] = y[n_2]) \}}.$

Условиями могут быть, например,  $(r_1.x(A_i) \theta r_2.y(f(A_i)))$ , или  $(r_1.x(f(A)) \theta r_2.y(A_i))$ , или  $((r_1.x(A_i) \theta r_2.y(A_i)) \text{and} (r_1.x(A_u) \theta r_2.y(A_v)))$ , или  $((r_1.x(A_i) \theta r_2.y(A_i)) \text{or} (r_1.x(A_u) \theta r_2.y(A_v)))$  и т. п.

### 1.3.3. Языки реляционной логики

Одним из таких языков является язык SQL (Structured Query Language – структурный язык запросов), используемый во всех инструментальных средствах разработки INFORMIX.

Формат записи операторов SQL свободный. Можно писать прописными или строчными буквами все подряд на одной строке, один оператор на нескольких строках, слова операторов можно разделять произвольным количеством пробелов и т.п.

Несмотря на свободный формат записи операторов

SQL, синтаксическую структуру любого запроса удобнее представить так:

```

SELECT <список атрибутов>
FROM <список отношений>
WHERE <предикат >, где
<список атрибутов>:=<атрибут > {«,»<атрибут>},
<список отношений>:=<отношение> {«,»<отношение>},
<предикат >:=< выражение>.
<выраже-
ние>:=<отношение>”.”<атрибут><оператор_сравнения>
<отношение>”.”<атрибут>| <выраже-
ние>«AND»<выражение>| <выра- жение
ние>«OR»<выражение >|«NOT»<выражение>.
<оператор_сравнения>:= «=»| «≠»| «<»| «>»| «≤»| «≥».
```

Для объяснения элементов реляционной логики достаточно ознакомиться с одним оператором группы манипулирования данными. Это - SELECT – выбор кортежа из отношения.

Первая строка инструкции - оператор SELECT – определяет схему формируемого отношения  $rel(r')$ , что в реляционной алгебре исполняет оператор  $\pi_{rel}(r)$ . Порядок и число атрибутов в формируемом отношении задан <списком атрибутов>. Если в <списке атрибутов> должны быть атрибуты из разных отношений, то в имени атрибута должны быть указаны имена отношений <атрибут>:= <имя\_отношения>«.»<имя\_атрибута>.

Например, SELECT  $r_1.A_1, r_3.A_4$ .

Вторая строка - оператор FROM – указывает список используемых отношений. Порядок кортежей в формируемом отношении управляется оператором ORDER BY,

после которого стоит имя атрибута и ключевое слово ASC (сортировка по возрастанию) или DESC (сортировка по убыванию): SELECT <список атрибутов>

FROM <список отношений>

ORDER BY <ИМЯ\_АТТРИБУТА> ASC.

или ORDER BY <ИМЯ\_АТТРИБУТА> DESC.

Пример 1.91.

SELECT	$A_1, r_1$	A A A	SELECT	$A_1, r_1$	A A A
$A_2, A_3$		1 2 3	$A_2, A_3$		1 1 2 3
FROM $r_1$		a <sub>1</sub> b <sub>1</sub> 1	FROM $r_1$		a <sub>2</sub> b 3
ORDER BY			ORDER BY		2
$A_3$ ASC.		a <sub>3</sub> b <sub>3</sub> 2	$A_3$ DESC.		a <sub>4</sub> b 3
		a <sub>2</sub> b <sub>2</sub> 3			1
		a <sub>4</sub> b <sub>1</sub> 3			a <sub>3</sub> b 2
					3
					a <sub>1</sub> b 1
					1

Чтобы в результирующем отношении не было дубликатов кортежей нужно после SELECT писать ключевое слово UNIQUE. Этот оператор представляет закон идемпотентности.

Пример 1.92.

SELECT	$A_1, r_1$	A A A	SELECT	$r_1$	A A A
$A_2, A_3$		1 1 2 3	UNIQUE $A_1, A_2, A_3$		1 1 2 3
FROM $r_1$		a b <sub>1</sub> 3	FROM $r_1$		a b 3
ORDER BY		4			4 1

$A_3$		ORDER BY	
DESC.	a b <sub>2</sub> 3	$A_3$ DESC.	a b 3
	2		2 2
	a b <sub>1</sub> 3		a b 2
	4		3 3
	a b <sub>3</sub> 2		a b 1
	3		1 1
	a b <sub>1</sub> 2		
	1		
	a b <sub>3</sub> 2		
	3		

Третья строка - оператор WHERE –формирует заданием предиката условия для извлечения из анализируемых отношений свободных переменных-кортежей, что в реляционной алгебре исполняет оператор  $\delta_B(r_1)$ .

В предложении WHERE при формировании условий могут быть использованы логические связки AND, OR, NOT. Синтаксическая структура предложения имеет вид: <выражение>:=<выражение>AND<выражение>|<выражение>OR<выражение>|”NOT”<выражение>.

Например, можно создать такие запросы и получить соответствующие ответы в виде таблиц.

Пример 1.93.

SELECT $A_1, A_2, A_3$	$r'_1$	$A_1$	$A_2$	$A_3$
FROM $r_1$		$a_2$	$b_2$	3
WHERE $A_1=a_3$ OR		$a_3$	$b_3$	2
$A_2=b_2$ .				
или				
SELECT $A_1, A_2, A_3$	$r'_1$	$A_1$	$A_2$	$A_3$

FROM r <sub>1</sub>	a <sub>4</sub>	b <sub>1</sub>	3
WHERE A <sub>3</sub> >2 AND			
A <sub>1</sub> =a <sub>4</sub> .			

SELECT A <sub>1</sub> , A <sub>2</sub> , A <sub>3</sub>	r' <sub>1</sub>	A <sub>1</sub>	A <sub>2</sub>	A <sub>3</sub>
FROM r <sub>1</sub>		a <sub>3</sub>	b <sub>3</sub>	2
WHERE NOT A <sub>3</sub> >2				
AND				
A <sub>1</sub> =a <sub>3</sub> OR a <sub>2</sub> =b <sub>1</sub> .				

Синтаксическая структура бинарных операций UNION, MINUS, (DIFFERENCE), INTERSECTION на языке SQL имеет следующий вид:

SELECT <список атрибутов>	SELECT
<список атрибутов>	
FROM <отношение>	
FROM <отношение>	
[WHERE <предикат >]	
[WHERE <предикат >]	
UNION	
	MINUS
SELECT <список атрибутов>	
SELECT <список атрибутов>	
FROM <отношение>	
FROM <отношение>	
[WHERE <предикат >].	
[WHERE <предикат >].	

Синтаксическая структура операции естественного соединения (JOIN) на языке SQL имеет следующий вид:

SELECT <список атрибутов>

```
FROM <отношение_1>INNER JOIN <отношение_2>
ON<отношение_1><«.»><атрибут><оператор_сравнения>
><отношение_2><«.»><атрибут>.
```

Синтаксическая структура операции  $\theta$ -соединения ( $\theta$ -JOIN) на языке SQL имеет следующий вид:

```
SELECT <список атрибутов>
FROM <отношение_1>INNER JOIN <отношение_2>
ON<отношение_1><«.»><атрибут><оператор_сравнения>
><отношение_2>”.”<атрибут>.
WHERE <предикат>.
```

Часто используют в операторе WHERE *вложенные подзапросы*, которые генерируют промежуточные отношения. На это указывает оператор IN, используемый для выяснения принадлежности элемента множеству.

```
SELECT<список атрибутов>
FROM <список отношений>
WHERE <предикат> IN
      SELECT<список атрибутов>
      FROM<список отношений>
      WHERE<предикат>.
```

Вложенный подзапрос формирует множество кортежей непоименованного отношения, которое было динамически создано и может отличаться от любого хранимого отношения. Поскольку это отношение неименовано, оно может использоваться только в том месте, в котором оно появляется в подзапросе. К такому отношению невозможно обратиться по имени из какого-либо другого запроса.

Операции, которые можно применить к подзапросу, основаны на тех операциях, которые можно применить к

множеству:  $x \text{ IN } X$ , т.е.  $x \in X$ ,  $x \text{ NOT IN } X$ , т.е.  $x \notin X$ .

При использовании оператора IN неявно применяется квантор существования, т.е. WHERE  $x \text{ IN } P$  эквивалентно формуле  $\exists x(P(x))$ . При использовании оператора NOT IN неявно применяется квантор всеобщности, т. е. WHERE  $x \text{ NOT IN } P$  эквивалентно формуле  $\forall x(P(x))$ .

Пример 1.94.

```
SELECT      ФАМИЛИЯ,          ДИСЦИПЛИНА,
ОТЧЕТНОСТЬ
FROM преподаватель_1
WHERE ДИСЦИПЛИНА=электроника IN
      SELECT ДИСЦИПЛИНА
      FROM учебный_план_1
      WHERE учебный_план_1.ДИСЦИПЛИНА=
            преподаватель_1.ДИСЦИПЛИНА
```

В данном запросе требуется найти всех преподавателей кафедры ..., ведущих занятия по дисциплине «электроника».

Некоторые операции языка SQL основаны на арифметических действиях, они являются встроенными функциями, их результатом является число. а именно:

COUNT (X) – количество элементов множества X, т.е. |X|.

SUM(X) – сумма всех элементов множества X,

MAX(X) – максимальный элемент множества X,

MIN(X) – минимальный элемент множества X,

AVG(X)= SUM(X)/COUNT(X) – среднее значение элемента множества X.

Синтаксическая структура этих операций имеет вид:

```
SELECT
```

---

[COUNT<атрибут>|SUM<атрибут>|MAX<атрибут>|  
MIN<атрибут>|AVG<атрибут>].

## Вопросы и задачи

1.3.1. Даны отношения  $r_1, r_2, r_3, r_4$ . Написать формулы на языке реляционного исчисления с переменными-кортежами и на SQL.

r	A	A	A	A	A	r <sub>2</sub>	A	A	A	A	A <sub>5</sub>	
1	1	2	3	4	5		1	2	3	4		union(r <sub>1</sub> , r <sub>2</sub> ),
	b	c <sub>1</sub>	d	1	4		b <sub>2</sub>	c <sub>3</sub>	d	2	3	minus (r <sub>3</sub> , r <sub>4</sub> ),
	1		1						4			product (r <sub>2</sub> , r <sub>4</sub> ),
	b	c <sub>2</sub>	d	2	3		b <sub>3</sub>	c <sub>3</sub>	d	3	2	intersection(r <sub>2</sub> ,
	2		2						3			r <sub>3</sub> ),
	b	c <sub>3</sub>	d	3	2		b <sub>4</sub>	c <sub>4</sub>	d	4	1	select(r <sub>3</sub> , A <sub>5</sub> >1 ),
	3		3						4			join(r <sub>1</sub> , r <sub>4</sub> ,
	b	c <sub>4</sub>	d	4	1		b <sub>1</sub>	c <sub>2</sub>	d	4	1	r <sub>1</sub> .A <sub>4</sub> =r <sub>4</sub> .A <sub>4</sub> ),
	4		4						3			join(r <sub>2</sub> , r <sub>3</sub> ,
												r <sub>2</sub> .A <sub>4</sub> >r <sub>3</sub> .A <sub>5</sub> ).
												Составить таб-
r	A	A	A	A	A	r <sub>4</sub>	A	A	A	A	A <sub>5</sub>	лицы по резуль-
3	1	2	3	4	5		1	2	3	4		татам исполне-
	b	c <sub>3</sub>	d	3	2		b <sub>3</sub>	c <sub>2</sub>	d	3	2	ния операций.
	4		2						1			
	b	c <sub>3</sub>	d	3	2		b <sub>3</sub>	c <sub>3</sub>	d	3	2	
	3		3						3			
	b	c <sub>2</sub>	d	1	4		b <sub>1</sub>	c <sub>2</sub>	d	1	4	
	1		3						3			
	b	c <sub>2</sub>	d	4	1		b <sub>2</sub>	c <sub>2</sub>	d	2	3	
	1		3						2			

1.3.2. По «Расписанию движения самолетов из Калининграда (аэропорт Храброво)» – Расписание\_1 и «Расписанию движения самолетов из Москвы (аэропорт Шере-

метьево)» - Расписание\_2 ответить на вопросы:

Расписание\_1

Аэропорт назначения	Отправление			
	Номер рейса	Дни вылета	Время (местное) вылета	Время (местное) прилета
Москва (Внуково)	K8986	1,2,3,4,5,6,7	08.15	11.05
Москва (Внуково)		1,2,3,4,5,6,7	16.00	18.50
Москва (Шереметьево)	K8990	2,5	13.00	15.50
Новосибирск	K8351	5,6	19.00	05.30
Новосибирск	K8353	4	21.00	05.45
С-Петербург	K8485*	1,3,5	09.15	12.00
С-Петербург	ПЛ8670	4	13.40	16.25
С-ПетербургГ	ПЛ8672	6	16.00	18.45
С-Петербург	ПЛ8668	2	19.05	21.50

Расписание\_2

Аэропорт назначения	Номер рейса	Дни вылета	Время (местное) вылета	Время (местное) прилета
Киев	UN201	1,2,3,4,5	09.10	09.30
Киев	UN211	1,2,3,4,5	18.30	18.50
Красноярск	UN511 1	2,4,6	20.00	04.25
Красноярск	UN514 7	1,2,3,4,5,6,7	23.35	08.15
Новосибирск	UN107	6	21.50	05.55
Новосибирск	UN107	3	22.50	05.50
Санкт-Петербург**	UN121	1,2,3,4,5	07.50	09.00
Санкт-Петербург**	UN141	1,2,3,4,5	19.00	20.15
Тель-Авив	UN311	4,6,7	19.30	22.45

а) как организовать наиболее удобный перелет Калининград–Москва–Санкт-Петербург?

б) Как организовать перелет Калининград–Москва–Красноярск?

Как организовать перелет Калининград–Москва–Киев?

д) Как организовать перелет Калининград–Москва–Новосибирск так, чтобы в четверг принять участие в работе конференции в 10.00?

е) Как организовать перелет в среду Калининград – Москва–Красноярск?

ф) Как организовать перелет Калининград–Москва – Тель-Авив?

---

g) На каких маршрутах вылетают самолеты из Калининграда после 18.00?

Примечание: резерв времени для переезда в Москве из одного аэропорта в другой не менее 3 часов.

### Расчетно-графическая работа

1) Из таблиц  $r_1$  и  $r_2$  ( $8 \times 8$ ) удалить согласно варианту четыре пары (номер столбца, номер строки), сформировать из оставшихся строк и столбцов таблицы индивидуального задания ( $r_1$  и  $r_2$ ),

2) по п.п. 1, 2, 3 задания выполнить операции  $(r_1 \cup r_2)$ ,  $(r_1 \cap r_2)$ ,  $(r_1 \setminus r_2)$ , написать формулы реляционной алгебры, реляционного исчисления с переменными-кортежами и составить запрос на языке SQL, нарисовать результирующие таблицы  $r'$ ,

4) по п. 4 задания написать формулы реляционной алгебры, реляционного исчисления с переменными-кортежами и написать запрос на языке SQL с подзапросом, составить таблицы для  $(\otimes)$ ,  $(\succ$  или  $\succ^\theta <)$ ,  $\delta$ ,  $\pi$ .

$r_1$	$\Delta$							
a1	b	c	d	1	2	3	4	
a2	b	c	d	2	3	4	1	
a3	b	c	d	3	4	1	2	
a4	b	c	d	4	1	2	3	
a1	b	c	d	4	3	2	1	
a2	b	c	d	3	2	1	4	
a3	b	c	d	2	1	4	3	
a4	b	c	d	1	4	3	2	

$r_2$	$\Delta$							
a1	b	c	3	d	1	2	3	4
a2	b	c	4	d	2	3	4	1
a3	b	c	1	d	3	4	1	2
a4	b	c	2	d	4	1	2	3
a1	b	c	1	d	4	3	2	1
a2	b	c	2	d	3	2	1	4
a3	b	c	3	d	2	1	4	3
a4	b	c	4	d	1	4	3	2

Вари-	Удалить (столбец, строка)	Задание
1	для $r_1$ : (3, 1), (4, 2), (7, 7), (8, 8);	$(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A2, r_2.A5, r_2.A6)}(\delta((r_1 \succ^\theta r_2)$

	для $r_2$ : (3, 4), (4, 5), (7, 6), (8, 8)	$r_1.A_6 < r_2.A_6$ , $r_2.A_5 > 1$ or $r_2.A_6 > 1$ )).
2	для $r_1$ : (3, 1), (4, 2), (7, 7), (8, 8); для $r_2$ : (3, 3), (4, 5), (7, 6), (8, 8)	1) $(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A_2, r_2.A_5, r_2.A_6)}(\delta((r_1 > r_2,$ $r_1.A_2 = r_2.A_2),$ $r_1.A_5 \geq 2$ and $r_2.A_5 < 4))$ .
3	для $r_1$ : (3, 1), (4, 2), (7, 7), (8, 8); для $r_2$ : (3, 3), (4, 4), (7, 6), (8, 8)	$(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A_2, r_2.A_5, r_2.A_6)}(\delta((r_1 >^{\theta} r_2,$ $r_1.A_6 < r_2.A_5),$ $r_1.A_5 > 2$ and $r_2.A_5 > 2))$ .
4	для $r_1$ : (3, 1), (4, 2), (7, 7), (8, 8); для $r_2$ : (3, 3), (4, 5), (7, 6), (8, 8)	1) $(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A_2, r_1.A_5, r_2.A_5)}(\delta((r_1 >^{\theta} r_2,$ $r_1.A_5 < r_2.A_5),$ $r_1.A_2 = b_1$ or $r_1.A_2 = b_2))$ .

5	для $r_1$ : (3, 1), (4, 2), (7, 7), (8, 8); для $r_2$ : (3, 3), (4, 5), (7, 6), (8, 7)	1) $(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A_1, r_1.A_5, r_2.A_6)}(\delta((r_1 >^\theta < r_2,$ $r_1.A_6 \geq r_2.A_5),$ $r_1.A_1 = a_3$ or $r_1.A_1 = a_4))$ .
6	для $r_1$ : (3, 1), (4, 2), (7, 6), (8, 8); для $r_2$ : (3, 4), (4, 5), (7, 6), (8, 8)	1) $(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A_1, r_1.A_5, r_2.A_5)}(\delta((r_1 > < r_2,$ $r_1.A_2 = r_2.A_2),$ $r_2.A_6 \geq 2$ and $r_2.A_5 \geq 2))$ .
7	для $r_1$ : (3, 1), (4, 2), (7, 6), (8, 8); для $r_2$ : (3, 3), (4, 5), (7, 6), (8, 8)	1) $(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A_1, r_1.A_5, r_2.A_6)}(\delta((r_1 > < r_2,$ $r_1.A_1 = r_2.A_1),$ $r_1.A_1 = a_3$ or $r_1.A_1 = a_4))$ .
8	для $r_1$ : (3, 1), (4, 2), (7, 6), (8, 8); для $r_2$ : (3, 3), (4, 4), (7, 6), (8, 8)	1) $(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A_1, r_1.A_5, r_2.A_6)}(\delta((r_1 >^\theta < r_2,$ $r_1.A_5 < r_2.A_6),$ $r_1.A_1 = a_3$ and $r_2.A_1 = a_4))$ .
9	для $r_1$ : (3, 1), (4, 2), (7, 6), (8, 8); для $r_2$ : (3, 3), (4, 4), (7, 5), (8, 8)	1) $(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A_1, r_1.A_2, r_2.A_5)}(\delta((r_1 >^\theta < r_2,$ $r_1.A_6 < r_2.A_6),$ $r_1.A_5 \geq 1$ or $r_2.A_5 \geq 1))$ .
10	для $r_1$ : (3, 1), (4, 2), (7, 6), (8, 8); для $r_2$ : (3, 3), (4, 4),	1) $(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A_1, r_1.A_2, r_2.A_5)}(\delta((r_1 >^\theta < r_2,$ $r_1.A_5 \geq r_2.A_5),$

	(7, 5), (8, 6)	$r_1.A_1=a_3$ and $r_2.A_6 \geq 2$ )).
11	для $r_1$ : (3, 1), (4, 2), (7, 5), (8, 8); для $r_2$ : (3, 3), (4, 4), (7, 5), (8, 6)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 1) $\pi_{(r_1.A_1, r_2.A_2, r_2.A_6)}(\delta((r_1 > r_2,$ $r_1.A_6 = r_2.A_6),$ $r_1.A_5 \geq 2$ and $r_2.A_5 < 4))$ .
12	для $r_1$ : (3, 1), (4, 2), (7, 5), (8, 8); для $r_2$ : (3, 3), (4, 4), (7, 6), (8, 7)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_1, r_2.A_2, r_2.A_6)}(\delta((r_1 >^{\theta} r_2,$ $r_1.A_5 \neq r_2.A_5),$ $r_1.A_1 = a_3$ or $r_2.A_5 \geq 2))$
13	для $r_1$ : (3, 1), (4, 2), (7, 5), (8, 8); для $r_2$ : (3, 3), (4, 4), (7, 5), (8, 8)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_1, r_2.A_2, r_1.A_5)}(\delta((r_1 >^{\theta} r_2,$ $r_1.A_5 \neq r_2.A_6),$ $r_1.A_1 = a_3$ or $r_2.A_5 \geq 2))$ .
14	для $r_1$ : (3, 1), (4, 2), (7, 5), (8, 8); для $r_2$ : (3, 3), (4, 4), (7, 6), (8, 8)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_1, r_1.A_5, r_2.A_6)}(\delta((r_1 >^{\theta} r_2,$ $r_1.A_5 \geq r_2.A_5),$ $r_1.A_5 \geq 2$ and $r_2.A_6 \geq 2)$ .
15	для $r_1$ : (3, 1), (4, 2), (7, 5), (8, 8); для $r_2$ : (3, 3), (4, 4), (7, 7), (8, 8)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_1, r_2.A_2, r_2.A_6)}(\delta((r_1 >^{\theta} r_2,$ $r_1.A_5 = r_2.A_6),$ $r_1.A_1 = a_3$ and $r_2.A_1 = a_4))$ .
16	для $r_1$ : (3, 1), (4, 2), (7, 4), (8, 8);	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_1, r_2.A_2, r_2.A_6)}(\delta((r_1 >^{\theta} r_2,$

	для $r_2$ : (3, 3), (4, 4), (7, 7), (8, 8)	$r_1.A_6 \neq r_2.A_6$ , $r_2.A_1 = a_2$ and $r_1.A_5 \geq 2$ )).
17	для $r_1$ : (3, 1), (4, 2), (7, 4), (8, 8); для $r_2$ : (3, 3), (4, 4), (7, 6), (8, 8)	1) $(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A_1, r_2.A_2, r_2.A_5)}(\delta((r_1 >^\theta < r_2,$ $r_1.A_5 \leq 3$ or $r_2.A_5 \leq 3), r_1.A_5 = 1$ or $r_2.A_5 \geq 2)$ .
18	для $r_1$ : (3, 1), (4, 2), (7, 4), (8, 8); для $r_2$ : (3, 3), (4, 4), (7, 5), (8, 8)	1) $(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A_1, r_1.A_5, r_2.A_6)}(\delta((r_1 >^\theta < r_2,$ $r_1.A_2 \neq r_2.A_2),$ $r_1.A_1 \neq a_2$ and $r_2.A_5 \neq 3))$ .

1 9	для $r_1$ : (3, 1), (4, 2), (7, 4), (8, 8); для $r_2$ : (3, 3), (4, 4), (7, 5), (8, 7)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_1, r_2.A_2, r_2.A_5)}(\delta((r_1 >^{\theta} < r_2, r_1.A_6 \leq r_2.A_6), r_1.A_1 \neq a_3$ or $r_2.A_5 \geq 2))$ .
2 0	для $r_1$ : (3, 1), (4, 2), (7, 4), (8, 8); для $r_2$ : (3, 3), (4, 4), (7, 5), (8, 7)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_1, r_2.A_2, r_2.A_6)}(\delta((r_1 > < r_2,$ $r_1.A_2 = r_2.A_2),$ $r_1.A_1 \neq a_2$ and $r_2.A_5 > 1))$ .
2 1	для $r_1$ : (3, 1), (4, 2), (7, 3), (8, 8); для $r_2$ : (3, 3), (4, 4), (7, 7), (8, 8)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_1, r_2.A_5, r_1.A_6)}(\delta((r_1 > < r_2,$ $r_1.A_5 = r_2.A_5),$ $r_1.A_1 \neq a_2$ and $r_2.A_5 \neq 3))$ .
2 2	для $r_1$ : (3, 1), (4, 2), (7, 3), (8, 8); для $r_2$ : (3, 3), (4, 4), (7, 6), (8, 8)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_1, r_1.A_2, r_2.A_5)}(\delta((r_1 >^{\theta} < r_2,$ $r_1.A_5 < r_2.A_5),$ $r_1.A_1 \neq a_3$ and $r_2.A_5 \geq 2))$ .
2 3	для $r_1$ : (3, 1), (4, 2), (7, 4), (8, 8); для $r_2$ : (3, 3), (4, 4), (7, 5), (8, 8)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_1, r_1.A_5, r_2.A_5)}(\delta((r_1 >^{\theta} < r_2,$ $r_1.A_5 \geq r_2.A_6),$ $r_1.A_1 \neq a_2$ and $r_2.A_5 \neq 3))$ .
2 4	для $r_1$ : (3, 1), (4, 2), (7, 3), (8, 8); для $r_2$ : (3, 3), (4, 4), (7, 5), (8, 7)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_1, r_1.A_2, r_2.A_5)}(\delta((r_1 > < r_2,$ $r_1.A_6 = r_2.A_6),$

		$r_2.A_1 \neq a_2$ or $r_2.A_5 \geq 2$ )).
2 5	для $r_1$ : (3, 1), (4, 2), (7, 3), (8, 8); для $r_2$ : (3, 3), (4, 4), (7, 5), (8, 6)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_1, r_1.A_2, r_2.A_5)}((\delta((r_1 >^\theta < r_2,$ $r_2.A_5 < r_1.A_5),$ $r_1.A_1 \neq a_3$ or $r_2.A_6 \geq 2))$ ).
2 6	для $r_1$ : (3, 1), (4, 2), (7, 3), (8, 7); для $r_2$ : (3, 3), (4, 4), (7, 7), (8, 8)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_1, r_1.A_5, r_2.A_6)}(\delta((r_1 >^\theta < r_2,$ $r_1.A_5 = r_2.A_6),$ $r_2.A_2 \neq b_3$ or $r_2.A_6 \geq 2))$ ).
2 7	для $r_1$ : (3, 1), (4, 2), (7, 3), (8, 7); для $r_2$ : (3, 3), (4, 4), (7, 6), (8, 8)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_1, r_2.A_2, r_1.A_6)}(\delta((r_1 >^\theta < r_2,$ $r_1.A_5 \neq r_2.A_5),$ $r_1.A_6 \geq 2$ and $r_2.A_6 < 4))$ ).
2 8	для $r_1$ : (3, 1), (4, 2), (7, 3), (8, 7); для $r_2$ : (3, 3), (4, 4), (7, 5), (8, 8)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_2, r_1.A_6, r_2.A_5)}(\delta((r_1 >^\theta < r_2,$ $r_1.A_6 \neq r_2.A_6),$ $r_1.A_1 \neq a_2$ and $r_2.A_6 < 4))$ ).
2 9	для $r_1$ : (3, 1), (4, 2), (7, 3), (8, 7); для $r_2$ : (3, 3), (4, 4), (7, 5), (8, 7)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_2.A_2, r_2.A_5, r_1.A_6)}((r_1 >^\theta < r_2,$ $r_1.A_5 = r_2.A_6),$ $r_1.A_2 \neq b_3$ and $r_2.A_5 \geq 2))$ ).

3 0	для $r_1$ : (3, 1), (4, 2), (7, 3), (8, 7); для $r_2$ : (3, 3), (4, 4), (7, 5), (8, 6)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_1, r_1.A_6, r_2.A_5)}(\delta((r_1 >^\theta < r_2,$ $r_1.A_6 \geq 3 \text{ and}$ $r_2.A_6 \geq 3),$ $r_1.A_1 \neq a_4)).$
3 1	для $r_1$ : (3, 1), (4, 2), (7, 3), (8, 6); для $r_2$ : (3, 3), (4, 4), (7, 7), (8, 8)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_1, r_2.A_5, r_1.A_6)}(\delta((r_1 >^\theta < r_2,$ $r_1.A_5 < r_2.A_6),$ $r_1.A_1 \neq a_2 \text{ or}$ $r_2.A_1 \neq a_3)).$
3 2	для $r_1$ : (3, 1), (4, 2), (7, 3), (8, 6); для $r_2$ : (3, 3), (4, 4), (7, 6), (8, 8)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_1, r_2.A_5, r_1.A_6)}(\delta((r_1 > < r_2,$ $r_1.A_6 = r_2.A_6),$ $r_1.A_1 \neq a_3 \text{ and}$ $r_2.A_1 \neq a_3)).$
3 3	для $r_1$ : (3, 1), (4, 2), (7, 3), (8, 6); для $r_2$ : (3, 3), (4, 4), (7, 5), (8, 8)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_1, r_2.A_2, r_2.A_2)}(\delta((r_1 >^\theta < r_2,$ $r_1.A_6 \leq 3 \text{ and}$ $r_2.A_6 < 3),$ $r_1.A_2 \neq b_2)).$
3 4	для $r_1$ : (3, 1), (4, 2), (7, 3), (8, 6); для $r_2$ : (3, 3), (4, 4), (7, 5), (8, 7)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_1, r_2.A_5, r_1.A_6)}(\delta((r_1 > < r_2,$ $r_1.A_5 = r_2.A_5),$ $r_1.A_1 \neq a_3 \text{ and}$ $r_2.A_1 \neq a_3)).$
3 5	для $r_1$ : (3, 1), (4, 2), (7, 3), (8, 6); для $r_2$ : (3, 3), (4, 4), (7, 5), (8, 6)	1) $(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ 4) $\pi_{(r_1.A_3, r_2.A_4, r_2.A_8)}(\delta((r_1 > < r_2,$ $r_1.A_6 = r_2.A_6),$

		$r_1.A_3 \neq c_1$ and $r_2.A_3 \neq c_1$ )).
3 6	для $r_1$ : (1, 1), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 4), (2, 5), (5, 6), (6, 8)	$(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ $\pi_{(r_1.A_3, r_2.A_7, r_2.A_8)}(\delta((r_1 >^{\theta} < r_2,$ $r_1.A_8 < r_2.A_8),$ $r_1.A_3 \neq c_1$ or $r_2.$ $A_3 \neq c_1$ )).
3 7	для $r_1$ : (1, 1), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 4), (2, 5), (5, 3), (6, 8)	$1) (r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ $\pi_{(r_1.A_3, r_2.A_3, r_1.A_8)}(\delta((r_1 > < r_2,$ $r_1.A_3 = r_2.A_3),$ $r_1.A_3 \neq c_1$ or $r_2.$ $A_3 \neq c_1$ )).
3 8	для $r_1$ : (1, 1), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 4), (2, 2), (5, 3), (6, 8)	$(r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ $4) \pi_{(r_1.A_3, r_2.A_4,$ $r_2.A_8)}(\delta((r_1 >^{\theta} < r_2, r_1.A_7 < r_2.A_8),$ $r_1.A_4 \neq d_1$ and $r_2.A_4 \neq c_1$ )).
3 9	для $r_1$ : (1, 1), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 3), (2, 2), (5, 1), (6, 8)	$1) (r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ $4) \pi_{(r_1.A_3, r_2.A_3, r_1.A_8)}(\delta((r_1 >^{\theta} < r_2,$ $r_1.A_7 < r_2.A_7),$ $r_1.A_3 \neq c_2$ and $r_2.A_7 > 2$ )).
4 0	для $r_1$ : (1, 1), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 3), (2, 2), (5, 1), (6, 7)	$1) (r_1 \cup r_2), 2) (r_1 \cap r_2), 3) (r_1 \setminus r_2),$ $4) \pi_{(r_1.A_3, r_1.A_4, r_2.A_8)}(\delta((r_1 >^{\theta} < r_2,$ $r_1.A_8 \geq r_2.A_8),$ $r_1.A_3 \neq c_1$ or $r_2.A_8 \neq 2$ )).

4 1	для $r_1$ : (1, 1), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 1), (2, 2), (5, 3), (6, 7)	1) $(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A1, r_2.A5, r_1.A6)}(\delta((r_1 >^{\theta} < r_2, A_7 = r_1.A_8), r_1.A_3 \neq c_2$ or $r_2.A_3 \neq c_3))$ .
4 2	для $r_1$ : (1, 1), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 1), (2, 2), (5, 7), (6, 6)	1) $(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A3, r_1.A4, r_2.A8)}(\delta((r_1 >^{\theta} < r_2, r_1.A_4 = r_2.A_4), r_1.A_7 > 2$ or $r_2.A_7 A_4 < 4))$ .
4 3	для $r_1$ : (1, 1), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 1), (2, 5), (5, 6), (6, 7)	1) $(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A3, r_2.A3, r_2.A8)}(\delta((r_1 >^{\theta} < r_2, r_1.A_7 < r_2.A_7), r_1.A_3 = c_2$ and $r_2.A_8 < 4))$ .
4 4	для $r_1$ : (1, 1), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 5), (2, 6), (5, 7), (6, 8)	1) $(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A3, r_1.A4, r_2.A8)}(\delta((r_1 >^{\theta} < r_2, r_1.A_7 < r_2.A_7), r_1.A_3 = c_3$ or $r_2.A_8 > 2))$ .
4 5	для $r_1$ : (1, 1), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 3), (2, 4), (5, 7), (6, 8)	1) $(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A1, r_2.A5, r_1.A6)}(\delta((r_1 >^{\theta} < r_2, r_1.A_7 \geq 2$ and $r_2.A_7 < 4), r_1.A_3 \neq c_1))$ .
4 6	для $r_1$ : (1, 1), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 3), (2, 5), (5, 7), (6, 8)	1) $(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A1, r_2.A2, r_2.A6)}(\delta((r_1 >^{\theta} < r_2, r_1.A_7 \neq r_2.A_7))$ .

	7), (6, 8)	$r_1.A_3 \neq c_2$ or $r_2.A_3 \neq c_3$ )).
4 7	для $r_1$ : (1, 1), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 2), (2, 5), (5, 7), (6, 8)	1) $(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A_1, r_2.A_5, r_1.A_6)}(\delta((r_1 >^\theta < r_2,$ $r_1.A_8 \neq r_2.A_8),$ $r_1.A_3 = c_2$ or $r_2.A_3 \neq c_3))$ .
4 8	для $r_1$ : (1, 1), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 2), (2, 4), (5, 7), (6, 8)	1) $(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A_3, r_2.A_3, r_2.A_8)}(\delta((r_1 >^\theta < r_2,$ $r_1.A_7 \neq r_2.A_7),$ $r_2.A_7 < 4$ and $r_1.A_3 \neq c_1))$ .
4 9	для $r_1$ : (1, 1), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 2), (2, 5), (5, 5), (6, 8)	1) $(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A_4, r_2.A_4, r_1.A_8)}(\delta((r_1 >^\theta < r_2,$ $r_1.A_7 \neq r_2.A_7),$ $r_1.A_4 = d_1$ and $r_2.A_4 \neq d_4))$ .
5 0	для $r_1$ : (1, 1), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 2), (2, 5), (5, 5), (6, 7)	1) $(r_1 \cup r_2)$ , 2) $(r_1 \cap r_2)$ , 3) $(r_1 \setminus r_2)$ , 4) $\pi_{(r_1.A_4, r_2.A_4, r_2.A_8)}(\delta((r_1 >^\theta < r_2,$ $r_1.A_4 \neq r_2.A_4),$ $r_1.A_3 \neq c_1$ and $r_2.A_3 \neq c_1))$ .

«Субъективизм – обычное дело  
при отыскании объективных причин».

Л.Кумор

«Те, кто верует слепо, - пути не найдут.  
Тех, кто мыслит, - сомнения вечно грызут».

Омар Хайям.

«Как бы ни была совершенна теория,  
она – только приближение к истине»

Александр Бутлеров.

## Глава 2. Неклассическая логика

Если классическая логика строго опирается на истинность высказывания и/или высказывательной функции, на достаточность законов «исключенного третьего» и непротиворечивости вывода, то неклассическая логика допускает множество рациональных значений истинности на интервале  $[0,1]$ , снимает ограничения законов классической логики. В настоящей главе выделены только два направления математической логики: нечёткой и модальной.

Нечёткая логика (fuzzi logic) оперирует с высказываниями, истинность которых может принимать любые рациональные значения на интервале  $[0, 1]$ . Даже неточные знания позволяют с помощью логической системы делать достаточно близкие к истине заключения. Например, мнения экспертов при формировании базы знаний экспертных систем могут быть противоречивыми, но использование логической системы позволяет снять эти ограничения и обеспечить пользователя нужными знаниями. Поэтому нечёткая логика определила развитие экспертных систем и систем искусственного интеллекта.

Модальная логика (modal logic) оперирует с так называемыми кванторами *модальности* высказываний (*необходимо* и *возможно*): того, что «должно быть», и того, что «может быть». Это определило развитие темпоральной (или временной) и алгоритмической логик.

Темпоральная логика (temporal logic) исследует модальные связи фактов во времени, описываемые в высказываниях: того, что «должно быть», и того, что «может быть» в прошлом и/или будущем. Эти знания позволяют с помощью логической системы организовать упорядоченное взаимодействие двух или нескольких процессов во времени. Например, параллельные вычисления на компьютере.

Алгоритмическая логика (algorithm logic) показывает: что и как надо делать для достижения очередного результата. Это позволяет организовывать доказательство правильности программ, используя логическую систему и раскрывая семантику языка программирования.

## 2.1. Нечёткая логика

### 2.1.1. Нечёткие множества

Нечёткая логика была предложена американским профессором Л. Заде, который для оценки частичной истинности или частичной лжи высказывания и/или суждения ввел множество рациональных чисел на интервале  $[0,1]$ .

Например, дано высказывание «Сидоров имеет рост 178см.» и дан предикат  $P(X):=$ « $X$  – высокий человек». Спрашивается: «Сидоров – высокий человек?» Некто установил значение истины высказывания  $P(\text{сидоров})=0,75$ .

Это означает, что высказывание «Сидоров – высокий человек» истинно на три четверти. Точно так же оно на одну четверть ложно.

Часто используют нечёткое описание объекта или его свойств. Например, высказывания: «большое входное сопротивление осциллографа» (какое?) или «большое число оборотов двигателя» (какое?). Такие высказывания не дают числовой характеристики атрибута, но с учетом опыта специалиста и описания сопутствующих фактов дают оценку истинности высказывания «иметь большое...».

Слабым моментом нечеткой логики является задание значения истинности на интервале  $[0,1]$ . Кто и как будет оценивать эти значения? Это - задачи высококвалифицированных специалистов в конкретной отрасли знаний (экспертов). И, как правило, при разработке экспертных систем используют опыт многих специалистов.

**Нечёткие множества элементов.** Пусть дано универсальное множество  $U$ . Если на этом множестве задать подмножество  $X'$ , имя которого недостаточно четко определено, то принадлежность элементов  $u \in U$  множеству  $X'$  может быть описана функцией принадлежности -  $\mu_{X'}(u)$ , как субъективная мера. Значение функции  $\mu_{X'}(u)$  называют степенью принадлежности и определяют на интервале  $[0, 1]$ , т.е.  $\mu_{X'}(u_i): U \rightarrow [0, 1]$ .

Поэтому модель нечёткого множества описывают так:

$$X' = \{ \mu_{X'}(u_1)/u_1, \mu_{X'}(u_2)/u_2, \dots, \mu_{X'}(u_n)/u_n \},$$

где  $\mu_{X'}(u_i) \in [0,1]$  – степень принадлежности каждого элемента  $u_i \in U$  нечёткому множеству  $X'$ .

*Носителем нечёткого множества*  $X'$  являются элементы четкого подмножества  $X \subseteq U$ , т. е.  $X = \{u_1, u_2, \dots, u_n\} \subseteq U$ , если для  $u_i \in X$   $\mu_{X'}(u_i) > 0$ . Если для  $u_i \in U$  имеем  $\mu_{X'}(u_i) = 1$ , то элемент четко принадлежит множеству  $X'$ . Если все элементы носителя  $X$  имеют значение  $\mu_{X'}(u_i) = 1$ , то дано четкое подмножество  $X' = X$  множества  $U$ . Если для  $u_i \in U$  имеем  $\mu_{X'}(u_i) = 0$ , то элемент четко не принадлежит множеству  $X'$ . Если все элементы носителя  $X$  имеют значение  $\mu_{X'}(u_i) = 0$ , то дано пустое множество, т. е.  $X' = \emptyset$ .

Пример 2.1. Дано 10 дискет. Эксперт должен сформировать множество подмножеств, удовлетворяющих предикату «выбрать несколько дискет» (как понимать слово «несколько»?). Множество всех подмножеств этих дискет содержит пустое множество, одно-, двух- трех- и т.д. до десятиэлементного подмножества. Следовательно, дано универсальное множество  $U$  по числу дискет в подмножестве, т.е.  $U = \{\emptyset, 1, 2, 3, \dots, 10\}$ .

Для подмножеств, содержащих нуль, один, два, половину или все дискеты, эксперт определил значение функции принадлежности равно нулю, так как проще сказать «возьмите одну дискету», «... две дискеты», «... половину дискет» или «... все дискеты». Для подмножеств, содержащих три, восемь или девять дискет, эксперт определил степень принадлежности 0,6, а для подмножеств, содержащих четыре, шесть или семь дискет, - 0,8. То есть эксперт сформировал нечёткое множество  $X'$  («выбрать несколько дискет») =  $\{0,6/3, 0,8/4, 0,8/6, 0,8/7, 0,6/8, 0,6/9\}$ , где над «/» указана степень принадлежности для выбранного

числа дискет, а под «/» - число дискет для данной степени принадлежности. Таково с учетом психических параметров эксперта оценка принадлежности каждого подмножества нечёткому высказыванию «выбрать несколько дискет».носителем нечёткого множества является  $X = \{3, 4, 6, 7, 8, 9\}$ .

Пример 2.2. Дан электрический двигатель, эксперт должен отнести значения скорости вращения работающего двигателя в четыре класса:  $X'_1$  - «нулевая скорость вращения»,  $X'_2$  - «малая»,  $X'_3$  - «средняя» и  $X'_4$  - «большая». Пусть скорость вращения двигателя изменяется в пределах от 0 до 3150 об/мин.

Эксперт разбил диапазон изменения скорости на восемь поддиапазонов, установил два уровня принадлежности скорости вращения двигателя к заданному классу (0,33 и 1,00) и составил таблицу истинности принадлежности каждому классу (см. табл. 2.1).

Например, в класс «средняя скорость вращения двигателя» эксперт выделил три значения скоростей в диапазоне от 1350 об/мин до 2250 об/мин. При этом для скорости 1350 об/мин эксперт установил степень истинности принадлежности этому классу равным 0,33, для скорости 1800 об/мин - 1 и для скорости 2250 об/мин - 0,33. Для других скоростей этого диапазона также можно установить степени истинности принадлежности, но это потребует уже более точного измерения числа оборотов двигателя.

В этом случае нечёткие множества описаны так:

$$X'_1 \text{ («нулевая») } = \{I/0, 0,33/450\},$$

$$X'_2 \text{ («малая») } = \{0,33/450, 1/900, 0,33/1350\},$$

$$X'_3 \text{ («средняя») } = \{0,33/1350, 1/1800, 0,33/2250\},$$

$$X'_4 \text{ («большая») } = \{0,33/2250, 1/2700, 1/3150\}.$$

Таблица 2.1

Скорость (об/мин)	Степень принадлежности			
	Нулевая	Малая	Средняя	Большая
0	1	0	0	0
450	0,33	0,33	0	0
900	0	1	0	0
1350	0	0,33	0,33	0
1800	0	0	1	0
2250	0	0	0,33	0,33
2700	0	0	0	1
3150	0	0	0	1

Забегая вперед, нужно дать определение прямого произведения нечётких множеств.

Если даны универсальное множество  $U = \{u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8, u_9\}$  и два нечётких подмножества  $X'_1 = \{0,6/u_1, 0,4/u_2, 0,8/u_3, 0,2/u_4, 1,0/u_5, 0,3/u_6\}$  и  $X'_2 = \{0,9/u_1, 0,4/u_2, 1,0/u_3, 0,7/u_7, 0,3/u_8, 0,5/u_9\}$ , то **прямое произведение** нечётких подмножеств есть множество  $X'$ , состоящее из пар  $(u_i, u_j)$ , первая компонента которых принадлежит нечёткому множеству  $X'_1$ , а вторая –  $X'_2$ , т. е.

$$X' = \{\mu_{X'}(u_i, u_j) / (u_i, u_j) \mid u_i \in X'_1 \text{ и } u_j \in X'_2\} = (X'_1 \otimes X'_2).$$

**Степень принадлежности**  $\mu_{X'}(u_i, u_j)$  пары  $(u_i, u_j)$ , равна минимальному значению степени принадлежности

$\mu_{X'1}(u_i)$  и  $\mu_{X'2}(u_j)$ , т.е

$$\mu_{X'}(u_i, u_j) = (\mu_{X'1}(u_i) \& \mu_{X'2}(u_j)) = \min\{\mu_{X'1}(u_i), \mu_{X'2}(u_j)\}.$$

Для множеств  $X'_1$  и  $X'_2$  прямое произведение представлено в таблица 2.2.

Таблица 2. 2

$C' = A' \otimes B'$	$u_1$	$u_2$	$u_3$	$u_7$	$u_8$	$u_9$
$u_1$	0,6	0,4	0,6	0,6	0,3	0,5
$u_1$	0,4	0,4	0,4	0,4	0,3	0,4
$u_3$	0,8	0,4	0,8	0,7	0,3	0,5
$u_4$	0,2	0,2	0,2	0,2	0,2	0,2
$u_5$	0,9	0,4	1,0	0,7	0,3	0,5
$u_6$	0,3	0,3	0,3	0,3	0,3	0,3

**Нечёткие множества отображений.** Если по какому-то нечёткому правилу для элементов множества  $X$  находят элементы множества  $Y$ , то формируют **нечёткое отображение**  $h': X \rightarrow Y$ , когда четко даны носители отображения  $x \in X$  и  $y \in Y$  и нечётко – их принадлежность отображению  $h'$ .

Значение **функции принадлежности**  $\mu_{h'}(x_i, y_j)$  пары  $(x_i, y_j) \in (X \otimes Y)$  есть **степень ее принадлежности** нечёткому отображению  $h'$ , т. е.

$$h' = \{\mu_{h'}(x_i, y_j) / (x_i, y_j) \mid x_i \in X, y_j \in Y\},$$

где над знаком «/» указана степень принадлежности нечёткому отображению  $\mu_{h'}(x_i, y_j)$  пары  $(x_i, y_j)$ , описанной под этим знаком.

Если область определения нечёткого отображения

дана на множестве  $X^n$ , т.е.  $h': X^n \xrightarrow{\mu} Y$ , то нечёткое отображение есть

$$h' = \{\mu_h(x_1, x_2, \dots, x_n, y) / (x_1, x_2, \dots, y) \mid x_i \in X^n, y \in Y\}.$$

Нечёткие отображения удобно представлять матрицами, элементы строки которых есть  $x_i \in X_i$  прообразы нечёткого отображения, а элементы столбца  $y \in Y$  – его образы. Тогда в каждую клеточку матрицы  $(x_i, y_j)$  – следует записать значение степени принадлежности  $\mu_h(x_i, y_j)$  для соответствующей пары  $(x_i, y_j)$ .

Пример 2.3. Пусть в пределах региона или города необходимо разместить двенадцать магазинов розничной торговли  $X = \{x_1, x_2, \dots, x_{11}, x_{12}\}$ , обслуживаемых фирмами «АЛТЫН», «ВИКТОРИЯ», «ВЕСТЕР» и «КОПЕЙКА» или  $Z = \{z_1, z_2, z_3, z_4\}$ .

Руководители фирм обратились к экспертам для определения рациональной структуры размещения магазинов в регионе. Пусть эксперты, обсуждая с населением региона, установили, что наибольшее влияние на размещение магазинов в регионе оказывают такие показатели: «доступность магазина» ( $y_1$ ), «высокое качество товара» ( $y_2$ ), «высокий уровень обслуживания» ( $y_3$ ) и «низкие цены» ( $y_4$ ), т.е.  $Y = \{y_1, y_2, y_3, y_4\}$ . Вспомните рекламные плакаты: «По-прежнему низкие цены!», «В трех шагах от дома!», «Высокое качество и сервис гарантированы!», «У нас отличное обслуживание!» и т.п. Эксперты, обсуждая с руководителями магазинов и фирм организацию торговли, установили их нечёткое понимание значимости предложенных населением региона показателей. Пусть учет мнений руководителей магазинов по набранным показателям

представлен в таблице 2.3, а мнений руководства фирм в таблице 2.4.

График нечёткого отображения мнений руководителей магазинов для каждого показателя есть

$$h'_1 = \{\mu_h(x_i, y_1)/(x_i, y_1), \mu_h(x_i, y_2)/(x_i, y_2), \mu_h(x_i, y_3)/(x_i, y_3), \mu_q(x_i, y_4)/(x_i, y_4)\}.$$

Анализ таблицы 2.3 показывает, что

- для показателя «доступность магазина» нечёткое множество мнений руководителей магазинов будет  $X'_{y_1}(x) = \{1,0/x_1, 1,0/x_5, 0,8/x_6, 0,7/x_7, 0,5/x_8, 0,5/x_9, 0,6/x_{10}, 0,1/x_{11}\}$ ,

- для руководителя магазина  $x_6$  отношение к этим показателям формирует нечёткое множество –  $Y'_{x_6}(y) = \{0,8/y_1, 0,4/y_2, 0,5/y_3, 0,9/y_4\}$ ,

- для руководителя магазином  $x_1$  значимым является только «доступность магазина», для  $x_2$  – «высокое качество товара», для  $x_3$  – «высокий уровень обслуживания», для  $x_4$  – «низкие цены», для  $x_5$  – важными являются все показатели, а для  $x_{11}$  – все они совершенно незначимы и т.д.

График нечёткого отображения мнений руководства фирм о каждом показателе есть

$$h'_2 = \{\mu_h(y_1, z_j)/(y_1, z_j), \mu_h(y_2, z_j)/(y_2, z_j), \mu_h(y_3, z_j)/(y_3, z_j), \mu_h(y_4, z_j)/(y_4, z_j)\}.$$

Таблица 2.3.

$h'_1$	$y_1$	$y_2$	$y_3$	$y_4$
$x_1$	1	0	0	0
$x_2$	0	1	0	0
$x_3$	0	0	1	0
$x_4$	0	0	0	1

Таблица 2.4.

$h'_2$	$z_1$	$z_2$	$z_3$	$z_4$
$y_1$	0,9	0,1	0,5	0,7
$y_2$	0,5	0,9	0,6	0,6
$y_3$	0,4	0,9	0,5	0,4
$y_4$	0,8	0,1	0,5	0,6

$x_5$	1	1	1	1
$x_6$	0,8	0,4	0,5	0,9
$x_7$	0,7	0,3	0,4	0,8
$x_8$	0,5	0,8	0,8	0,2
$x_9$	0,5	0,5	0,5	0,5
$x_{10}$	0,6	0,7	0,8	0,5
$x_{11}$	0,1	0,1	0,1	0,1
$x_{12}$	0	0	1	1

Анализ таблицы 2.4 показывает, что

- для показателя «доступность магазина» нечёткое множество мнений руководства фирм будет нечёткое множество  $Z'_{y_1}(z) = \{0,9/z_1, 0,1/z_2, 0,5/z_3, 0,7/z_4\}$ ,

- для руководства фирмой  $z_4$  отношение к этим показателям формирует нечёткое множество  $Y'_{z_4}(y) = \{0,7/y_1, 0,6/y_2, 0,4/y_3, 0,6/y_4\}$ ,

- для руководства фирмой  $z_1$  наиболее значимыми являются «доступность магазина» и «низкие цены», для  $z_2$  – «высокое качество товара» и «высокий уровень обслуживания», для  $z_3$  – все показатели слабо значимы, а для  $z_4$  – незначим показатель «высокий уровень обслуживания».

Для оценки согласия руководства фирм и магазинов о размещении магазинов в заданном регионе необходимо выполнить композицию  $h' = (h'_1 \cdot h'_2)$ . Эта операция позволит вычислить их согласие по заданным показателям и установить порог значимости их мнений для правильной организации торговли в регионе. Об этом более подробно в 2.1.4.

**Нечёткие множества отношений.** Если по какому-то нечёткому правилу между элементами множества  $X$  дано

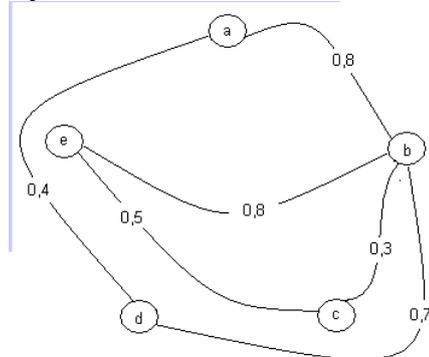
нечёткое отношение  $r': X \rightarrow X$ , когда четко даны элементы  $x_i, x_j \in X$ , а степень принадлежности  $\mu_{r'}(x_i, x_j)/(x_i, x_j)$  каждой пары  $(x_i, x_j) \in X \otimes X$  формирует нечёткое отношение:

$$r' = \{ \mu_{r'}(x_i, x_j)/(x_i, x_j) \mid x_i, x_j \in X \}.$$

Нечёткие отношения удобно описывать матрицами. В ее клетках  $(x_i, x_j)$  указаны степени принадлежности нечёткому отношению  $\mu_{r'}(x_i, x_j)$ .

Если дано  $n$ -арное отношение  $r': X^{n-1} \rightarrow X$ , то значение функции принадлежности должно быть найдено для каждого набора  $(x_{1i}, x_{2i}, \dots, x_{ni})$ , т.е.  $r' = \{ \mu_{r'}(x_1, x_2, \dots, x_n)/(x_1, x_2, \dots, x_n) \mid x_1, x_2, \dots, x_n \in X \}$ .

**Пример 2.4.** Пусть в результате стихийного бедствия нарушились транспортные связи между населенными пунктами  $a, b, c, d, e$ .



r	a	b	c	d	e
a	0	0,8	0	0,4	0
b	0,8	0	0,3	0,7	0,8
c	0	0,3	0	0	0,5
d	0,4	0,7	0	0	0
e	0	0,8	0,5	0	0

**Рис. 2.1.** Нечёткие отношения.

Эксперты МЧС установили степень принадлежности нечёткому отношению смежности между населенными пунктами как показано на рис.2.1 и в таблице 2.5.

Для нечётких отношений также могут быть установ-

лены свойства нечётких рефлексивности, симметричности и транзитивности, которые позволят формировать классы нечётких отношений. Об этом подробнее см. 2.1.2.3.

### 2.1.2. Нечёткая алгебра

Если универсальное множества  $U$  содержит нечёткие подмножества, то к ним применимы обычные теоретико-множественные операции: *объединение*  $X'=(X'_i \cup X'_j)$ , *пересечение*  $X'=(X'_i \cap X'_j)$  и *дополнение*  $\neg X'=U \setminus X'$ . Особенностью исполнения этих операций является определение степени принадлежности в результате исполнения операций для каждого элемента универсального множества.

Множество нечётких подмножеств универсального множества  $\Gamma'(U)=\{X'_1, X'_2, X'_3, \dots\}$  и множество операторов  $F=\{\neg, \cup, \cap\}$  представляют нечёткую алгебру:

$$A = \langle \Gamma'(U), F, \mu(u_i) \rangle,$$

где  $\mu(u_i)$  – функция принадлежности элементов  $(u_i)$  универсального множества  $U$  нечёткому множеству  $X'$ .

Любое нечёткое множество универсального множества есть *элементарная формула*, т. е.  $X'_i = F'_i$ . Правильную последовательность нечётких множеств и операторов, называют *формулой*, т.е. если  $F'_1$  и  $F'_2$  – формулы, то  $\neg F'$ ,  $(F'_1 \cup F'_2)$ ,  $(F'_1 \cap F'_2)$  – также формулы. Никаких других формул в нечёткой алгебре нет.

#### 2.1.2.1. Операции над нечёткими множествами

**Включение нечёткого множества  $X'_1$  в нечёткое множество  $X'_2$**  есть суждение «если  $'u' \in U$  принадлежит  $X'_1$ , то он принадлежит  $X'_2$ ».

Степень включения нечёткого множества  $X'_1$  в нечёткое множество  $X'_2$  равна конъюнкции степеней принадлежности результатов исполнения операций импликации для каждого элемента  $u_i \in U$ , т. е.

$$\begin{aligned}\mu_{(X'_1 \subseteq X'_2)} &= \&_i(\mu_{X'_1}(u_i) \rightarrow \mu_{X'_2}(u_i)) = \&_i(\mu_{\neg X'_1}(u_i) \vee \mu_{X'_2}(u_i)) = \\ &= \min\{\max\{1 - \mu_{X'_1}(u_i), \mu_{X'_2}(u_i)\}\}.\end{aligned}$$

Если  $\mu_{(X'_1 \subseteq X'_2)} \geq 0,5$ , то говорят, что множество  $X'_1$  нечётко включено в множество  $X'_2$ .

**Пример 2.5.** Даны  $U = \{u_1, u_2, u_3, u_4, u_5\}$ ,  $X'_1 = \{0,3/u_2, 0,6/u_3, 0,4/u_5\}$  и  $X'_2 = \{0,8/u_1, 0,5/u_2, 0,7/u_3, 0,6/u_5\}$ . Включено ли множество  $X'_1$  в множество  $X'_2$ ?

$$\begin{aligned}\mu_{(X'_1 \subseteq X'_2)} &= \min\{\max\{1/u_1, 0,8/u_1\}, \max\{0,7/u_2, 0,5/u_2\}, \\ &\max\{0,4/u_3, 0,7/u_3\}, \max\{1/u_4, 0/u_4\}, \max\{0,6/u_5, 0,6/u_5\}\} = \\ &= \min\{1/u_1, 0,7/u_2, 0,7/u_3, 1/u_4, 0,6/u_5\} = 0,6.\end{aligned}$$

Следовательно, нечёткое множество  $X'_1$  нечётко включено в нечёткое множество  $X'_2$ .

**Равенство нечётких множеств  $X'_1$  и  $X'_2$  есть суждение «если  $u \in U$  принадлежит  $X'_1$ , то он принадлежит  $X'_2$  и если он принадлежит  $X'_2$ , то принадлежит  $X'_1$ ».**

Степень равенства нечётких множеств  $X'_1$  и  $X'_2$  равна конъюнкции степеней принадлежности результатов исполнения операции эквивалентности для каждого элемента универсального множества, т. е.

$$\begin{aligned}\mu_{(X'_1 \equiv X'_2)} &= \&_i(\mu_{X'_1}(u_i) \leftrightarrow \mu_{X'_2}(u_i)) = \&_i((\mu_{\neg X'_1}(u_i) \vee \mu_{X'_2}(u_i)) \&_i(\mu_{\neg X'_2}(u_i) \vee \mu_{X'_1}(u_i))) = \\ &= \min\{\min\{\max\{1 - \mu_{X'_1}(u_i), \mu_{X'_2}(u_i)\}\}, \min\{\max\{1 - \mu_{X'_2}(u_i), \mu_{X'_1}(u_i)\}\}\}.\end{aligned}$$

Если  $\mu_{(X'_1 \equiv X'_2)} \geq 0,5$ , то множества  $X'_1$  и  $X'_2$  нечётко равны.

**Пример 2.6.** Даны  $U = \{u_1, u_2, u_3, u_4, u_5\}$ ,  $X'_1 = \{0,8/u_2, 0,6/u_3, 0,1/u_5\}$  и  $X'_2 = \{0,3/u_1, 0,6/u_2, 0,7/u_3, 0,2/u_4, 0,3/u_5\}$ . Равны ли нечётко множества  $X'_1$  и  $X'_2$ ?

$$\begin{aligned} \mu_{(X'_1 \equiv X'_2)} = & \min \{ \min \{ \max \{ 1/u_1, 0,3/u_1 \}, \max \{ 0/u_1, 0,7/u_1 \} \}, \\ & \min \{ \max \{ 0,2/u_2, 0,6/u_2 \}, \max \{ 0,8/u_2, 0,4/u_2 \} \}, \\ & \min \{ \max \{ 0,4/u_3, 0,7/u_3 \}, \max \{ 0,6/u_3, 0,3/u_3 \} \}, \min \{ \max \{ 1/u_4, \\ & 0,2/u_4 \}, \max \{ 0/u_4, 0,8/u_4 \} \}, \min \{ \max \{ 0,9/u_5, 0,3/u_5 \}, \\ & \max \{ 0,1/u_5, 0,7/u_5 \} \} = \min \{ \min \{ 1/u_1, 0,7/u_1 \}, \min \{ 0,6/u_2, \\ & 0,8/u_2 \}, \min \{ 0,7/u_3, 0,6/u_3 \}, \min \{ 1/u_4, 0,8/u_4 \}, \min \{ 0,9/u_5, \\ & 0,7/u_5 \} \} = \min \{ 0,7/u_1, 0,6/u_2, 0,6/u_3, 0,8/u_4, 0,7/u_5 \} = 0,6. \end{aligned}$$

Следовательно, множество  $X'_1$  нечётко равно множеству  $X'_2$ .

**Объединение нечётких множеств  $X'_1$  и  $X'_2$**  есть множество  $X'$ , состоящее из элементов множества  $U$ , которые принадлежат нечёткому множеству  $X'_1$  или  $X'_2$ , т. е.  $X' = (X'_1 \cup X'_2)$ .

Степень принадлежности элементов универсального множества нечёткому множеству  $X'$  равна дизъюнкции степеней принадлежности элементов универсального множества нечётким множествам  $X'_1$  и  $X'_2$ , т. е.

$$\mu_{X'}(u_i) = (\mu_{X'_1}(u_i) \vee \mu_{X'_2}(u_i)) = \max \{ \mu_{X'_1}(u_i), \mu_{X'_2}(u_i) \}.$$

**Пример 2.7.** Даны два нечётких множества  $X'_1 = \{0,6/u_1, 0,4/u_2, 0,8/u_3, 0,2/u_4, 1,0/u_5, 0,3/u_6\}$  и  $X'_2 = \{0,9/u_1, 0,4/u_2, 1,0/u_3, 0,7/u_7, 0,3/u_8, 0,5/u_9\}$ . Найти  $X' = (X'_1 \cup X'_2)$ .

$$X' = (X'_1 \cup X'_2) = \{0,9/u_1, 0,4/u_2, 1,0/u_3, 0,2/u_4, 1,0/u_5, 0,3/u_6, 0,7/u_7, 0,3/u_8, 0,5/u_9\}.$$

**Пример 2.8.** Даны нечёткие отображения множества  $X$  на множество  $Y$   $h'_1 = \{\mu_{h'_1}(x_i, y_j)/(x_i, y_j)\}$  и  $h'_2 = \{\mu_{h'_2}(x_i, y_j)/(x_i, y_j)\}$ .

Найти  $h' = (h'_1 \cup h'_2)$ .

Степень принадлежности элемента  $(x_i, y_j)$  объединению двух нечётких отображений есть

$$\mu_{h'}(x_i, y_j) = (\mu_{h'_1}(x_i, y_j) \vee \mu_{h'_2}(x_i, y_j)) = \max_{i,j} \{\mu_{h'_1}(x_i, y_j), \mu_{h'_2}(x_i, y_j)\}.$$

В таблицах приведены результаты исполнения этой операции

$h'_1$	$y_2$	$y_3$	$y_4$	$h'_2$	$y_2$	$y_3$	$y_4$	$h'$	$y_2$	$y_3$	$y_4$
$x_1$	0,2	0,4	0,6	$x_1$	0,4	0,2	0,8	$x_1$	0,4	0,4	0,8
$x_2$	0,3	0,5	0,7	$x_2$	0,5	0,7	0,3	$x_2$	0,5	0,7	0,7
$x_3$	0,2	0,5	0,4	$x_3$	0,5	0,2	0,6	$x_3$	0,5	0,5	0,6
$x_4$	0,3	0,6	0,9	$x_4$	0,4	0,7	0,8	$x_4$	0,4	0,7	0,9

**Пример 2.9.** Даны нечёткие отношения на множестве  $X$   $r'_1 = \{\mu_{r'_1}(x_i, x_j)/(x_i, x_j)\}$  и  $r'_2 = \{\mu_{r'_2}(x_i, x_j)/(x_i, x_j)\}$ . Найти  $r' = (r'_1 \cup r'_2)$ .

Степень принадлежности элемента  $(x_i, x_j)$  объединению двух нечётких отношений определяют по формуле:

$$\mu_{r'}(x_i, x_j) = (\mu_{r'_1}(x_i, x_j) \vee \mu_{r'_2}(x_i, x_j)) = \max \{\mu_{r'_1}(x_i, x_j), \mu_{r'_2}(x_i, x_j)\}.$$

В таблицах приведены результаты исполнения этой операции

$r'_1$	$x_1$	$x_2$	$x_3$	$x_4$	$r'_2$	$x_1$	$x_2$	$x_3$	$x_4$	$r'$	$x_1$	$x_2$	$x_3$	$x_4$
$x_1$	0,	0,	0,	0,	$x_1$	0,	0,	0,	0,	$x_1$	0,	0,	0,	0,
	2	4	6	3		4	2	8	9		4	4	8	9
$x_2$	0,	0,	0,	0,	$x_2$	0,	0,	0,	0,	$x_2$	0,	0,	0,	0,
	3	5	7	5		5	7	3	7		5	7	7	7
$x_3$	0,	0,	0,	0,	$x_3$	0,	0,	0,	0,	$x_3$	0,	0,	0,	0,

$$x_4 \begin{array}{c|cccc} 2 & 5 & 4 & 7 \\ \hline 0, & 0, & 0, & 0, \\ 3 & 6 & 9 & 9 \end{array} \quad x_4 \begin{array}{c|cccc} 5 & 2 & 6 & 5 \\ \hline 0, & 0, & 0, & 0, \\ 4 & 7 & 8 & 3 \end{array} \quad x_4 \begin{array}{c|cccc} 5 & 5 & 6 & 7 \\ \hline 0, & 0, & 0, & 0, \\ 4 & 7 & 9 & 9 \end{array}$$

**Пересечение нечётких множеств**  $X'_1$  и  $X'_2$  есть множество  $X'$ , состоящее из элементов множества  $U$ , которые принадлежат нечётким множествам  $X'_1$  и  $X'_2$ , т. е.  $X'=(X'_1 \cap X'_2)$ .

Степень принадлежности элементов универсального множества нечёткому множеству  $X'$  равна конъюнкции степени принадлежности элементов универсального множества нечётким множествам  $X'_1$  и  $X'_2$ , т.е.

$$\mu_{X'}(u_i) = (\mu_{X'_1}(u_i) \& \mu_{X'_2}(u_i)) = \min\{\mu_{X'_1}(u_i), \mu_{X'_2}(u_i)\}.$$

**Пример 2.10.** Даны нечёткие множества  $X'_1 = \{0,6/u_1, 0,4/u_2, 0,8/u_3, 0,2/u_4, 1,0/u_5, 0,3/u_6\}$  и  $X'_2 = \{0,9/u_1, 0,4/u_2, 1,0/u_3, 0,7/u_7, 0,3/u_8, 0,5/u_9\}$ . Найти  $X'=(X'_1 \cap X'_2)$ .  
 Ответ:  $X'=(X'_1 \cap X'_2) = \{0,6/u_1, 0,4/u_2, 0,8/u_3\}$ .

**Пример 2.11.** Даны нечёткие отображения  $h'_1 = \{\mu_{h'_1}(x_i, y_j)/(x_i, y_j)\}$  и  $h'_2 = \{\mu_{h'_2}(x_i, y_j)/(x_i, y_j)\}$ . Найти  $h'=(h'_1 \cap h'_2)$ .

Степень принадлежности элемента  $(x_i, y_j)$  пересечению нечётких отображений есть

$$\mu_{h'}(x_i, y_j) = (\mu_{h'_1}(x_i, y_j) \& \mu_{h'_2}(x_i, y_j)) = \min_{i,j}\{\mu_{h'_1}(x_i, y_j), \mu_{h'_2}(x_i, y_j)\}.$$

В таблицах приведены результаты исполнения этой операции.

$$\begin{array}{c|ccc} h_1 & y_2 & y_3 & y_4 \\ \hline x_1 & 0,2 & 0,4 & 0,6 \\ x_2 & 0,3 & 0,5 & 0,7 \end{array} \cap \begin{array}{c|ccc} h_2 & y_2 & y_3 & y_4 \\ \hline x_1 & 0,4 & 0,2 & 0,8 \\ x_2 & 0,5 & 0,7 & 0,3 \end{array} = \begin{array}{c|ccc} h' & y_2 & y_3 & y_4 \\ \hline x_1 & 0,2 & 0,2 & 0,6 \\ x_2 & 0,3 & 0,5 & 0,3 \end{array}$$

$x_3$	$\left  \begin{array}{ccc} 0,2 & 0,5 & 0,4 \end{array} \right.$
$x_4$	$\left  \begin{array}{ccc} 0,3 & 0,6 & 0,9 \end{array} \right.$

$x_3$	$\left  \begin{array}{ccc} 0,5 & 0,2 & 0,6 \end{array} \right.$
$x_4$	$\left  \begin{array}{ccc} 0,4 & 0,7 & 0,8 \end{array} \right.$

$x_3$	$\left  \begin{array}{ccc} 0,2 & 0,2 & 0,4 \end{array} \right.$
$x_4$	$\left  \begin{array}{ccc} 0,3 & 0,6 & 0,8 \end{array} \right.$

**Пример 2.12.** Даны нечёткие отношения  $r'_1 = \{\mu_{r'_1}(x_i, x_j)/(x_i, x_j)\}$  и  $r'_2 = \{\mu_{r'_2}(x_i, x_j)/(x_i, x_j)\}$ . Найти  $r' = (r'_1 \cap r'_2)$ .

Степень принадлежности элемента  $(x_i, x_j)$  пересечению нечётких отношений есть:

$$\mu_{r'}(x_i, x_j) = (\mu_{r'_1}(x_i, x_j) \& \mu_{r'_2}(x_i, x_j)) = \min_{i,j} \{\mu_{r'_1}(x_i, x_j), \mu_{r'_2}(x_i, x_j)\}.$$

В таблицах приведены результаты исполнения этой операции.

$r'_1$	$x_1$	$x_2$	$x_3$	$x_4$	$r'_2$	$x_1$	$x_2$	$x_3$	$x_4$	$r'$	$x_1$	$x_2$	$x_3$	$x_4$
$x_1$	0,	0,	0,	0,	$x_1$	0,	0,	0,	0,	$x_1$	0,	0,	0,	0,
	2	4	6	3		4	2	8	9		2	2	6	3
$x_2$	0,	0,	0,	0,	$x_2$	0,	0,	0,	0,	$x_2$	0,	0,	0,	0,
	3	5	7	5		5	7	3	7		3	5	3	5
$x_3$	0,	0,	0,	0,	$x_3$	0,	0,	0,	0,	$x_3$	0,	0,	0,	0,
	2	5	4	7		5	2	6	5		2	2	4	5
$x_4$	0,	0,	0,	0,	$x_4$	0,	0,	0,	0,	$x_4$	0,	0,	0,	0,
	3	6	9	9		4	7	8	3		3	6	8	3

**Дополнение нечёткого множества  $X'_1$**  есть нечёткое множество  $\neg X'_1$ , состоящее из всех элементов универсального множества  $U$ , которые не принадлежат нечёткому множеству  $X'_1$ .

Степень принадлежности элементов универсального множества нечёткому множеству  $\neg X'_1$  равна дополнению степени принадлежности элементов нечёткому множеству  $X'_1$  до степени принадлежности универсальному множеству  $U$ , т.е.

$$\mu_{\neg X^1}(u) = 1 - \mu_{X^1}(u).$$

Пример 2.13. Даны универсальное множество  $U = \{u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8, u_9\}$  и два нечётких подмножества  $X^1 = \{0,6/u_1, 0,4/u_2, 0,8/u_3, 0,2/u_4, 1,0/u_5, 0,3/u_6\}$  и  $X^2 = \{0,9/u_1, 0,4/u_2, 1,0/u_3, 0,7/u_7, 0,3/u_8, 0,5/u_9\}$ . Найти  $\neg X^1$  и  $\neg X^2$ .

Ответ:  $\neg X^1 = \{0,4/u_1, 0,6/u_2, 0,2/u_3, 0,8/u_4, 0,7/u_6, 1,0/u_7, 1,0/u_8, 1,0/u_9\}$ ,

$\neg X^2 = \{0,1/u_1, 0,6/u_2, 1,0/u_4, 1,0/u_5, 1,0/u_6, 0,3/u_7, 0,7/u_8, 0,5/u_9\}$ .

Пример 2.14. Дано нечёткое отображение  $h = \{\mu_h(x_i, y_j)/(x_i, y_j)\}$ .

Найти  $\neg h$ .

Степень принадлежности элементов  $(x_i, y_j)$  нечёткому отображению  $\neg h(x_i, y_j)$  равна дополнению степени принадлежности элементов нечёткого отображения  $h(x_i, y_j)$ , т.е.  $\mu_{\neg h}(x_i, y_j) = (1 - \mu_h(x_i, y_j))$ .

Результаты вычислений  $\neg h$  приведены в таблицах.

$h$	$y_2$	$y_3$	$y_4$		$\neg h$	$y_2$	$y_3$	$y_4$
$x_1$	0,2	0,4	0,6	$\Rightarrow$	$x_1$	0,8	0,6	0,4
$x_2$	0,3	0,5	0,7		$x_2$	0,7	0,5	0,3
$x_3$	0,2	0,5	0,4		$x_3$	0,8	0,5	0,6
$x_4$	0,0	0,0	0,0		$x_4$	1,0	1,0	1,0

$$4 \mid 3 \quad 6 \quad 9 \qquad \mid 7 \quad 4 \quad 1$$

**Пример 2.15.** Дано нечёткое отношение  $r' = \{\mu_{r'}(x_i, x_j) / (x_i, x_j)\}$ . Найти  $\neg r'$ .

Степень принадлежности элементов  $(x_i, x_j)$  нечёткому отношению  $\neg r'(x_i, x_j)$  равна дополнению степени принадлежности элементов нечёткого отношения  $r'(x_i, x_j)$ , т.е.  $\mu_{\neg r'}(x_i, x_j) = (1 - \mu_{r'}(x_i, x_j))$ .

$r'$	$x_1$	$x_2$	$x_3$	$x_4$	$\neg r'$	$x_1$	$x_2$	$x_3$	$x_4$
$x_1$	0,4	0,2	0,8	0,9	$x_1$	0,6	0,8	0,2	0,1
$x_2$	0,5	0,7	0,3	0,7	$x_2$	0,5	0,3	0,7	0,3
$x_3$	0,5	0,2	0,6	0,5	$x_3$	0,5	0,8	0,4	0,5
$x_4$	0,4	0,7	0,8	0,3	$x_4$	0,6	0,3	0,2	0,7

**Разность нечётких множеств**  $X'_1$  и  $X'_2$  есть множество  $X'$ , состоящее из тех элементов множества  $U$ , которые принадлежат нечёткому множеству  $X'_1$  и не принадлежат нечёткому множеству  $X'_2$ ,

т. е.  $X' = X'_1 \setminus X'_2 = X'_1 \cap \neg X'_2$ , где «\» - символ разности множеств.

Степень принадлежности элемента универсального множества нечёткому множеству  $X'$  равна конъюнкции степеней принадлежности нечёткому множеству  $X'_1$  и дополнению нечёткого множества  $X'_2$ , т.е.

$$\mu_{X'}(u_i) = (\mu_{X_1}(u_i) \& (1 - \mu_{X_2}(u_i))) = \min\{\mu_{X_1}(u_i), (1 - \mu_{X_2}(u_i))\}.$$

**Пример 2.16.** Даны универсальное множество  $U = \{u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8, u_9\}$  и два нечётких подмножества  $X'_1 = \{0,6/u_1, 0,4/u_2, 0,8/u_3, 0,2/u_4, 1,0/u_5, 0,3/u_6\}$  и  $X'_2 = \{0,9/u_1, 0,4/u_2, 1,0/u_3, 0,7/u_7, 0,3/u_8, 0,5/u_9\}$ . Найти  $X' = X'_1 \setminus X'_2$ .  
 Ответ:  $X' = X'_1 \setminus X'_2 = \{0,1/u_1, 0,4/u_2, 0,2/u_4, 1,0/u_5, 0,3/u_6\}$ .

**Пример 2.17.** Даны нечёткие отображения  $h'_1 = \{\mu_{h'_1}(x_i, y_j)/(x_i, y_j)\}$  и  $h'_2 = \{\mu_{h'_2}(x_i, y_j)/(x_i, y_j)\}$ . Найти  $h' = (h'_1 \setminus h'_2)$ .

Степень принадлежности элементов  $(x_i, y_j)$  нечёткому отображению  $h'(x_i, y_j)$  равна конъюнкции степеней принадлежности элементов нечёткому отображению  $h'_1(x_i, y_j)$  и дополнению нечёткого отображения  $\neg h'_2(x_i, y_j)$ , т.е.

$$\mu_{h'}(x_i, y_j) = (\mu_{h'_1}(x_i, y_j) \& \mu_{\neg h'_2}(x_i, y_j)) = \min\{\mu_{h'_1}(x_i, y_j), (1 - \mu_{h'_2}(x_i, y_j))\}.$$

В таблицах приведены результаты исполнения этой операции.

$h'_1$	$y_1$	$y_2$	$y_3$	$h'_2$	$y_1$	$y_2$	$y_3$	$h'$	$y_1$	$y_2$	$y_3$
$x_1$	0,2	0,4	0,6	$x_1$	0,4	0,2	0,8	$x_1$	0, 0, 0, 2 4 2		
$x_2$	0,3	0,5	0,7	$x_2$	0,5	0,7	0,3	$x_2$	0, 0, 0, 3 3 3		
$x_3$	0,2	0,5	0,4	$x_3$	0,5	0,2	0,6	$x_3$	0, 0, 0, 2 5 4		
$x_4$	0,3	0,6	0,9	$x_4$	0,4	0,7	0,8	$x_4$	0, 0, 0, 3 3 2		

**Пример 2.18.** Даны нечёткие отношения  $r'_1 = \{\mu_{r'_1}(x_i, x_j)/(x_i, x_j)\}$  и  $r'_2 = \{\mu_{r'_2}(x_i, x_j)/(x_i, x_j)\}$ . Найти  $r' = (r'_1 \setminus r'_2)$ .

Степень принадлежности элементов  $(x_i, x_j)$  нечёткому отношению  $\gamma'(x_i, x_j)$  равна конъюнкции степеней принадлежности элементов нечёткому отношению  $\gamma'_1(x_i, x_j)$  и дополнению нечёткого отношения  $\neg\gamma'_2(x_i, x_j)$ , т.е.

$$\mu_{\gamma'}(x_i, x_j) = (\mu_{\gamma'_1}(x_i, x_j) \& \mu_{\gamma'_2}(x_i, x_j)) = \min_{i,j} \{ \mu_{\gamma'_1}(x_i, x_j), (1 - \mu_{\gamma'_2}(x_i, x_j)) \}.$$

В таблицах приведены результаты исполнения этой операции.

$\gamma'_1$	$x_1$	$x_2$	$x_3$	$x_4$	$\gamma'_2$	$x_1$	$x_2$	$x_3$	$x_4$	$\gamma'$	$x_1$	$x_2$	$x_3$	$x_4$

**Симметрическая разность нечётких множеств  $X'_1$  и  $X'_2$**  есть нечёткое множество  $X'$ , состоящее из элементов универсального множества  $U$ , которые принадлежат нечётким множествам  $(X'_1 \& \neg X'_2)$  или  $(X'_2 \& \neg X'_1)$ , т. е.  $X' = (X'_1 \Delta X'_2) = (X'_1 \cap \neg X'_2) \cup (X'_2 \cap \neg X'_1)$ , где « $\Delta$ » - символ симметрической разности.

Степень принадлежности элементов универсального множества нечёткому множеству  $X'$  равна дизъюнкции степеней принадлежности нечетким множествам  $X'_1 \cap \neg X'_2$  и  $X'_2 \cap \neg X'_1$ , т.е.

$$\mu_{X'}(u) = (\mu_{X'_1}(u) \& \mu_{\neg X'_2}(u)) \vee (\mu_{X'_2}(u) \& \mu_{\neg X'_1}(u)) = \max \{ \min \{ \mu_{X'_1}(u), \mu_{\neg X'_2}(u) \}, \min \{ \mu_{X'_2}(u), \mu_{\neg X'_1}(u) \} \}.$$

**Пример 2.19.** Даны универсальное множество  $U = \{u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8, u_9\}$  и два нечётких подмножества  $X'_1 = \{0,6/u_1, 0,4/u_2, 0,8/u_3, 0,2/u_4, 1,0/u_5, 0,3/u_6\}$  и  $X'_2 = \{0,9/u_1, 0,4/u_2, 1,0/u_3, 0,7/u_7, 0,3/u_8, 0,5/u_9\}$ . Найти  $X' = (X'_1 \Delta X'_2)$ .

Ответ:  $X' = (X'_1 \Delta X'_2) = \{0,4/u_1, 0,4/u_2, 0,2/u_3, 0,2/u_4, 1,0/u_5, 0,3/u_6, 0,7/u_7, 0,3/u_8, 0,5/u_9\}$ .

**Пример 2.20.** Даны нечёткие отображения  $h'_1 = \{\mu_{h'_1}(x_i, y_j)/(x_i, y_j)\}$  и  $h'_2 = \{\mu_{h'_2}(x_i, y_j)/(x_i, y_j)\}$ . Найти  $h' = (h'_1 \Delta h'_2)$ .

Степень принадлежности элемента  $(x_i, y_j)$  нечеткому множеству симметрической разности двух нечётких отображений  $h'_1$  и  $h'_2$  есть

$$\mu_{h'}(x_i, y_j) = (\mu_{h'_1}(x_i, y_j) \& \mu_{\neg h'_2}(x_i, y_j)) \vee (\mu_{h'_2}(x_i, y_j) \& \mu_{\neg h'_1}(x_i, y_j)) = \max\{\min\{\mu_{h'_1}(x_i, y_j), (1 - \mu_{h'_2}(x_i, y_j))\}, \min\{\mu_{h'_2}(x_i, y_j), (1 - \mu_{h'_1}(x_i, y_j))\}\}$$

В таблицах приведены результаты исполнения этой операции.

$h_1$	$y_1$ $y_2$ $y_3$	$h_2$	$y_1$ $y_2$ $y_3$	$h'$	$y_1$ $y_2$ $y_3$
$x_1$	0,2 0,4 0,6	$x_1$	0,4 0,2 0,8	$x_1$	0,4 0,4 0,4
$x_2$	0,3 0,5 0,7 $\Delta$	$x_2$	0,5 0,7 0,3 =	$x_2$	0,5 0,5 0,3
$x_3$	0,2 0,5 0,4	$x_3$	0,5 0,2 0,6	$x_3$	0,5 0,5 0,6
$x_4$	0,3 0,6 0,9	$x_4$	0,4 0,7 0,8	$x_4$	0,4 0,4 0,2

**Пример 2.21.** Даны нечёткие отношения  $r'_1 = \{\mu_{r'_1}(x_i, x_j)/(x_i, x_j)\}$  и  $r'_2 = \{\mu_{r'_2}(x_i, x_j)/(x_i, x_j)\}$ . Найти  $r' = (r'_1 \Delta r'_2)$ .

Степень принадлежности элемента  $(x_i, x_j)$  нечеткого отношения симметрической разности двух нечётких отношений  $\Gamma_1$  и  $\Gamma_2$  есть

$$\mu_{\Gamma}(x_i, x_j) = (\mu_{\Gamma_1}(x_i, x_j) \& (1 - \mu_{\Gamma_2}(x_i, x_j))) \vee (\mu_{\Gamma_2}(x_i, x_j) \& (1 - \mu_{\Gamma_1}(x_i, x_j))) = \max\{\min\{\mu_{\Gamma_1}(x_i, x_j), (1 - \mu_{\Gamma_2}(x_i, x_j))\}, \min\{\mu_{\Gamma_2}(x_i, x_j), (1 - \mu_{\Gamma_1}(x_i, x_j))\}\}$$

В таблицах приведены результаты исполнения этой операции.

$\Gamma_1$	$x_1$	$x_2$	$x_3$	$x_4$	$\Gamma_2$	$x_1$	$x_2$	$x_3$	$x_4$	$\Gamma'$	$x_1$	$x_2$	$x_3$	$x_4$
$x_1$	0,2	0,4	0,6	0,3	$x_1$	0,4	0,2	0,8	0,9	$x_1$	0,4	0,4	0,4	0,7
$x_2$	0,3	0,5	0,7	0,5	$\Delta x_2$	0,5	0,7	0,3	0,7	$= x_2$	0,5	0,5	0,7	0,5
$x_3$	0,2	0,5	0,4	0,7	$x_3$	0,5	0,2	0,6	0,5	$x_3$	0,5	0,5	0,6	0,5
$x_4$	0,3	0,6	0,9	0,9	$x_4$	0,4	0,7	0,8	0,3	$x_4$	0,4	0,4	0,2	0,7

**Композиция нечётких отображений** есть нечёткое отображение  $h' = (h'_1 \circ h'_2) = \{\mu_{h'}(x_i, z_j) / (x_i, z_j)\}$ . Степень принадлежности нечёткому отображению  $h' = \{\mu_{h'}(x_i, z_j) / (x_i, z_j)\}$  существует тогда и только тогда, когда есть хотя бы один элемент  $y_j$ , принадлежащий нечётким отображениям  $h'_1 = \{\mu_{h'}(x_i, y_j) / (x_i, y_j)\}$  и  $h'_2 = \{\mu_{h'}(y_j, z_k) / (y_j, z_k)\}$ , и вычисляется операцией дизъюнкции для каждого значения  $y_j$ , т. е.

$$\mu_{h'}(x_i, z_k) = \vee (\mu_{h'_1}(x_i, y_j) \& \mu_{h'_2}(y_j, z_k)) = \max\{\min\{\mu_{h'_1}(x_i, y_j), \mu_{h'_2}(y_j, z_k)\}\}$$

**Пример 2.22.** Даны нечёткое отображение  $h': X' \rightarrow Y'$  и нечёткое множество  $X' = \{0,6/x_1, 0,9/x_4, 0,1/x_5\}$ . Найти нечёткое множество  $Y'$ .

Пусть  $x_1, x_2, x_3, x_4$  и  $x_5$  есть, соответственно, внешние признаки заболевания: высокая температура, хрипы в грудной клетке, головная боль, насморк, кашель, а  $y_1, y_2, y_3$  и  $y_4$  – возможные заболевания: мигрень, респираторное заболевание, бронхит, астма. Пусть нечеткая матрица  $h'$  составлена ведущими диагностами медицины, а  $X'$  – нечеткие данные о заболевании пациента, полученные в нестандартных условиях (в лесу, экспедиции и т.п.). Композиция  $(X' \cdot h')$  формирует вопрос о болезни пациента.

Степень принадлежности множеству  $Y'$  есть  $\mu_{Y'}(y_1) = \max\{\min\{0,6, 0\}, \min\{0,9, 0\}, \min\{0,1, 0\}\} = 0$ ,  
 $\mu_{Y'}(y_2) = \max\{\min\{0,6, 0,2\}, \min\{0,9, 0,3\}, \min\{0,1, 0,7\}\} = 0,3$ ,  
 $\mu_{Y'}(y_3) = \max\{\min\{0,6, 0\}, \min\{0,9, 0\}, \min\{0,1, 0,8\}\} = 0,1$ ,  
 $\mu_{Y'}(y_4) = \max\{\min\{0,6, 0\}, \min\{0,9, 0\}, \min\{0,1, 0\}\} = 0$ .

Результаты представлены в таблице.

		$h'$				$Y'$
		$v_1$	$v_2$	$v_3$	$v_4$	
$X'$	$x_1$	0	0	0	0	$v_1$
	$x_2$	0	0	0	0	$v_2$
	$x_3$	1	0	0	0	$v_3$
	$x_4$	0	0	0	0	$v_4$
	$x_5$	0	0	0	0	

Ответ:  $Y' = \{0, 3/v_3\}$ .

То есть по данным о внешних признаках у пациента, возможно, респираторное заболевание. Надежность диагноза не более 0,3.

**Пример 2.23.** Даны нечёткие отображения

$h'_1 = \{0,3/(x_1, y_1), 1,0/(x_1, y_3), 0,7/(x_2, y_1), 0,9/(x_2, y_2), 0,4/(x_2, y_3)\}$  и  $h'_2 = \{0,2/(y_1, z_1), 0,8/(y_1, z_3), 1,0/(y_1, z_4), 0,3/(y_2, z_1), 0,4/(y_2, z_4)\}$ . Найти  $h' = (h'_1 \cdot h'_2)$ .

Ответ:  $h' = \{0,2/(x_1, z_1), 0,3/(x_1, z_3), 0,3/(x_1, z_4), 0,3/(x_2, z_1), 0,7/(x_2, z_3), 0,7/(x_2, z_4)\}$ .

**Пример 2.24.** Продолжая пример по размещению магазинов в заданном регионе (см. таблицы 2.3 и 2.4), найдем композицию двух отображений  $h'=(h'_1 \circ h'_2)$ .

		Таб-				Например, $\mu_{h'}(x_{10}, z_2)=\max\{\min\{(\mu_{h'_1}(x_{10}, y_1), \mu_{h'_2}(y_1, z_2)\}, \min\{\mu_{h'_1}(x_{10}, y_2), \mu_{h'_2}(y_2, z_2)\}, \min\{\mu_{h'_1}(x_{10}, y_3), \mu_{h'_2}(y_3, z_2)\}, \min\{\mu_{h'_1}(x_{10}, y_4), \mu_{h'_2}(y_4, z_2)\}\} = \max\{\min\{0,6, 0,1\}, \min\{0,7, 0,9\}, \min\{0,8, 0,9\}, \min\{0,5, 0,1\}\} = \max\{0,1, 0,7, 0,8, 0,1\} = 0,8.$
$h'$	$z_1$	$z_2$	$z_3$	$z_4$		
$x_1$	0	0	0	0		
$x_2$	0	0	0	0		
$x_3$	0	0	0	0		
$x_4$	0	0	0	0		
$x_5$	0, 9	0, 9	0, 6	0, 7		
$x_6$	0	0	0	0		
$x_7$	0	0	0	0		
$x_8$	0	0	0	0		
$x_9$	0	0	0	0		
$x_{10}$	0	0	0	0		
$x_{11}$	0	0	0	0		
$x_{12}$	0	0	0	0		
$x_{13}$	0	0	0	0		
$x_{14}$	0, 0	0, 0	0, 0	0, 0		

Таблица 2.7a)

$h'$	$z_1$	$z_2$	$z_3$	$z_4$
$x_1$	0,	-	0,	0,
	9		5	7
$x_2$	0,	0,	0,	0,
	5	9	6	6
$x_3$	-	0,	0,	-
		9	5	
$x_4$	0,	-	0,	0,
	8		5	6
$x_5$	0,	0,	0,	0,
	9	9	6	7
$x_6$	0,	0,	0,	0,
	8	5	5	7
$x_7$	0,	-	0,	0,
	8		5	7

Таблица 2.7b)

$h'$	$z_1$	$z_2$	$z_3$	$z_4$
$x_1$	0,	-	-	0,
	9			7
$x_2$	-	0,	0,	0,
		9	6	6
$x_3$	-	0,	-	-
		9		
$x_4$	0,	-	-	0,
	8			6
$x_5$	0,	0,	0,	0,
	9	9	6	7
$x_6$	0,	-	-	0,
	8			7
$x_7$	0,	-	-	0,
	8			7

$x_8$	0,	0,	0,	0,	$x_8$	-	0,	0,	0,
	5	8	6	6			8	6	6
$x_9$	0,	0,	0,	0,	$x_9$	-	-	-	-
	5	5	5	5					
$x_{10}$	0,	0,	0,	0,	$x_{10}$	0,	0,	0,	0,
	6	8	6	6		6	8	6	6
$x_{11}$	-	-	-	-	$x_{11}$	-	-	-	-
$x_{12}$	0,	0,	0,	0,	$x_{12}$	0,	0,	-	0,
	8	9	5	6		8	9		6

Эта композиция позволяет вычислить нечёткое согласие руководителей магазинов и фирм по заданным показателям. Композиция  $h' = \{\mu_{h'}(x_i, z_j) / (x_i, z_j)\}$  представлена в таблице 2.6. Анализ таблицы показывает степень согласия руководителей фирм и магазинов.

Если установить порог согласия, например,  $\alpha \geq 0,5$ , то можно выделить зоны согласия руководителей фирм и магазинов. В таблице 2.7а) удалены значения степеней принадлежности менее 0,5.

Однако для объективного задания  $\alpha$  следует выполнить попарное сравнение всех фирм для каждого руководителя магазина, т.е. вычислить степени принадлежности для каждой пары  $(z_i, z_j)$ :

$$\mu(x_i, (z_s, z_t)) = \mu(x_i, (z_s \& z_t)) = \min_{s,k} \{\mu(x_i, z_s) \mu(x_i, z_t)\}.$$

Все вычисления сведены в таблицу 2.8. По найденным значениям  $\mu(x_i, (z_s, z_t))$  вычислить максимальное значение  $\mu$  для каждой пары  $(z_s, z_t)$  по формуле  $\mu(z_s, z_t) = \max\{\mu(x_i, (z_s, z_t))\}$ . Выбрать среди множества  $\mu(z_s, z_t) = \{0,9; 0,6; 0,7; 0,6; 0,7; 0,6\}$  минимальное значение и принять его равным границе, разделяющей значимые и незначимые степени принад-

лежности  $\alpha \geq \min\{\mu(x_i, z_j)\}=0,6$ . В таблице 2.7b) приведены результаты расчетов для  $\alpha \geq 0,6$ .

Степень принадлежности композиции мнений руководителей магазинов и фирм более 0,6 показывает:

- высокую заинтересованность руководства магазинов в связях с фирмами; например, почти все руководители магазинов заинтересованы в связях с фирмами  $z_1, z_2$  и  $z_4$ ,
- соперничество фирм в продвижении своих товаров; например, фирмы  $z_1$  и  $z_4$  конкурируют в продвижении товаров в магазинах  $x_1, x_6, x_7$ , фирмы  $z_1, z_2, z_4$  - в магазине  $x_5$  а фирмы  $z_1, z_2$  - в магазине  $x_{12}$ .

Таблица 2.8

$h'(x_i, (z_s, z_t))$	$ \mu(x_i, (z_1, z_2)) $	$ \mu(x_i, (z_1, z_3)) $	$ \mu(x_i, (z_1, z_4)) $	$ \mu(x_i, (z_2, z_3)) $	$ \mu(x_i, (z_2, z_4)) $	$ \mu(x_i, (z_3, z_4)) $
$x_1$	0,1	0,5	0,7	0,1	0,1	0,5
$x_2$	0,5	0,5	0,5	0,6	0,6	0,6
$x_3$	0,4	0,4	0,4	0,5	0,4	0,4
$x_4$	0,1	0,5	0,6	0,1	0,1	0,5
$x_5$	0,9	0,6	0,7	0,6	0,7	0,6
$x_6$	0,5	0,5	0,7	0,5	0,5	0,5
$x_7$	0,4	0,5	0,7	0,4	0,4	0,5
$x_8$	0,5	0,5	0,5	0,6	0,6	0,6
$x_9$	0,5	0,5	0,5	0,5	0,5	0,5
$x_{10}$	0,6	0,6	0,6	0,6	0,6	0,6
$x_{11}$	0,1	0,1	0,1	0,1	0,1	0,1
$x_{12}$	0,8	0,5	0,6	0,5	0,6	0,5
$\max \mu(z_s, z_t)$	0,9	0,6	0,7	0,6	0,7	0,6

**Композиция нечётких отношений** есть нечёткое отношение  $r'=(r'_1 \circ r'_2)=\{\mu_{r'}(x_{1i}, x_{2k})/(x_{1i}, x_{2k})\}$ . Степень при-

надлежности нечёткому отношению  $r' = \{\mu_{r'}(x_i, x_j)/(x_i, x_j)\}$  существует тогда и только тогда, когда есть хотя бы один элемент  $x_k$ , принадлежащий нечётким отношениям  $r'_1 = \{\mu_{r'_1}(x_i, x_k)/(x_i, x_k)\}$  и  $r'_2 = \{\mu_{r'_2}(x_k, x_j)/(x_k, x_j)\}$ , и вычисляется операцией дизъюнкции для каждого значения  $x_k$ , т. е.

$$\mu_{r'}(x_i, x_j) = \vee(\mu_{r'_1}(x_i, x_k) \& \mu_{r'_2}(x_k, x_j)) = \max_k \{\min\{\mu_{r'_1}(x_i, x_k), \mu_{r'_2}(x_k, x_j)\}\}.$$

**Пример 2.25.** Пусть даны нечёткие отношения  $r'_1$  и  $r'_2$ . Найти  $r' = (r'_1 \circ r'_2)$ .

В таблицах приведены результаты исполнения этой операции

$r'_1$	$x_1$	$x_2$	$x_3$	$x_4$	$r'_2$	$x_1$	$x_2$	$x_3$	$x_4$	$r'$	$x_1$	$x_2$	$x_3$	$x_4$
1														
$x_1$	0, 0, 0, 0,				$x_1$	0, 0, 0, 0,				$x_1$	0, 0, 0, 0,			
1	2 4 6 3					4 2 8 9					5 4 6 5			
$x_2$	0, 0, 0, 0,				$x_2$	0, 0, 0, 0,				$x_2$	0, 0, 0, 0,			
2	3 5 7 5					5 7 3 7					5 5 6 5			
$x_3$	0, 0, 0, 0,				$x_3$	0, 0, 0, 0,				$x_3$	0, 0, 0, 0,			
3	2 5 4 7					5 2 6 5					5 7 7 5			
$x_4$	0, 0, 0, 0,				$x_4$	0, 0, 0, 0,				$x_4$	0, 0, 0, 0,			
4	3 6 9 9					4 7 8 3					5 7 8 6			

**Пример 2.26.** Для поиска надежной связи между населенными пунктами (см. пример 2.4) следует воспользоваться матрицами достижимости  $q^{i+i} = I \cup r' \cup r'^2 \cup \dots \cup r'^i$ , степень принадлежности нечеткой матрице достижимости вычисляют по формуле:

$$\mu_{q^{i+i}}(x_i, x_k) = (\mu_{q^{i+i-1}}(x_i, x_j) \vee (\vee_k (\mu_{r'}(x_i, x_k) \& (\mu_{r'^{i-1}}(x_j, x_k)))) = \max_k \{\mu_{q^{i+i-1}}(x_i, x_j), \max_k \{\min\{\mu_{r'}(x_i, x_k), \mu_{r'^{i-1}}(x_j, x_k)\}\}\}.$$

В таблице 2.9 приведены результаты расчетов степени принадлежности только для  $q^{+2}$ . Анализ таблицы показывает:

- из пункта 'а' в пункт 'е' и наоборот можно проехать через пункт 'b', степень надёжности этого маршрута 0,8,
- из пункта 'd' в пункт 'е' и наоборот можно проехать через пункт 'b', степень надёжности этого маршрута 0,7,
- из пункта 'b' в пункт 'с' и наоборот можно проехать через пункт 'е', степень надёжность маршрута 0,5 и т.д.

Таблица 2.9

$r'$	a	b	c	d	e	$q^{+}$	a	b	c	d	e
a	0	0,8	0	0,4	0	a	1	0,8	0	0,4	0
b	0,8	0	0,3	0,7	0,8	b	0,8	1	0,3	0,7	0,8
c	0	0,3	0	0	0,5	c	0	0,3	1	0	0,5
d	0,4	0,7	0	0	0	d	0,4	0,7	0	1	0
e	0	0,8	0,5	0	0	e	0	0,8	0,5	0	1

$r'^2$	a	b	c	d	e	$q^{+2}$	a	b	c	d	e
a	0,8	0,4	0,3	0,7	0,8	a	1	0,8	0,3	0,7	0,8
b	0,4	0,8	0,5	0,7	0,8	b	0,8	1	0,5	0,7	0,8
c	0,3	0,5	0,5	0,3	0,3	c	0,3	0,5	1	0,3	0,5
d	0,7	0,4	0,3	0,7	0,7	d	0,7	0,7	0,3	1	0,7
e	0,8	0,3	0,3	0,7	0,,	e	0,8	0,8	0,5	0,7	1

### 2.1.2.2. Законы нечёткой алгебры

Две нечёткие формулы  $F'_i$  и  $F'_j$  называют равносильными и обозначают  $F'_i \equiv F'_j$ , если они имеют одинаковые степени принадлежности. Если две формулы равносильны,

то они эквивалентны, т.е.  $F'_i \leftrightarrow F'_j$ .

Законы нечёткой алгебры позволяют исполнять эквивалентные преобразования сложных алгебраических формул и находить степени принадлежности каждого элемента универсального множества в результате исполнения алгебраических операций. В таблице 2.10 приведены основные законы нечёткой алгебры.

Таблица 2.10

Имя закона	Равносильные формулы $F'_i \equiv F'_j$ и равносильные значения функций
Коммутативности	$(F'_1 \cup F'_2) \equiv (F'_2 \cup F'_1)$ , $\max\{\mu_{F'_1}(u_i), \mu_{F'_2}(u_i)\} \equiv \max\{\mu_{F'_2}(u_i), \mu_{F'_1}(u_i)\}$ , $(F'_1 \cap F'_2) \equiv (F'_2 \cap F'_1)$ ,
Ассоциативности	$F'_1 \cup (F'_2 \cup F'_3) \equiv (F'_1 \cup F'_2) \cup F'_3$ , $\max\{\mu_{F'_1}(u_i), \max\{\mu_{F'_2}(u_i), \mu_{F'_3}(u_i)\}\} \equiv$ $\equiv \max\{\max\{\mu_{F'_1}(u_i), \mu_{F'_2}(u_i)\}, \mu_{F'_3}(u_i)\}$ , $F'_1 \cap (F'_2 \cap F'_3) \equiv (F'_1 \cap F'_2) \cap F'_3$
Дистрибутивности	$F'_1 \cup (F'_2 \cap F'_3) \equiv (F'_1 \cup F'_2) \cap (F'_1 \cup F'_3)$ , $\max\{\mu_{F'_1}(u_i), \min\{\mu_{F'_2}(u_i), \mu_{F'_3}(u_i)\}\} \equiv$ $\equiv \min\{\max\{\mu_{F'_1}(u_i), \mu_{F'_2}(u_i)\},$ $\max\{\mu_{F'_1}(u_i), \mu_{F'_3}(u_i)\}\}$ ,
Идемпотентности	$F \cup F \equiv F$ $\max\{\mu_F(u_i), \mu_F(u_i)\} \equiv \mu_F(u_i)$ , $F \cap F \equiv F$ $\min\{\mu_F(u_i), \mu_F(u_i)\} \equiv \mu_F(u_i)$
Де Моргана	$\neg(F_1 \cup F_2) \equiv \neg F_1 \cap \neg F_2$ , $1 - \max\{\mu_{F'_1}(u_i), \mu_{F'_2}(u_i)\} \equiv \min\{(1 - \mu_{F'_1}(u_i)), (1 - \mu_{F'_2}(u_i))\}$ ,
Дополнения	$\neg(\neg F) \equiv F$ , $(1 - (1 - \mu_{F'}(u_i))) \equiv \mu_{F'}(u_i)$
Константы	$F \cup \emptyset \equiv F$ , $F \cup U \equiv U$ <span style="float: right;"><math>F \cap \emptyset \equiv \emptyset</math>, <math>F \cap U \equiv F</math></span>

## 2.1.2.3. Свойства нечётких отношений

Чтобы формировать классы нечётких отношений, необходимо знать свойства отношений. Но эти свойства также обладают нечёткостью, т.е. существуют нечёткая рефлексивность, нечёткая симметричность, нечёткая транзитивность.

Для определения нечётких свойств отношений нужно вычислять степени нечёткости для каждой позиции отношения.

**Степенью рефлексивности** нечёткого отношения  $\alpha(r')_{\text{ref}}$  называется величина, определяемая выражением:

$$\alpha(r')_{\text{ref}} = \& \mu_{r'}(x_i, x_i) = \min_i \{\mu_{r'}(x_i, x_i)\}.$$

Отношение называют **нечётко рефлексивным**, если  $\alpha(r')_{\text{ref}} \geq 0,5$ .

**Степенью антирефлексивности** нечёткого отношения  $\beta(r')_{\text{ref}}$  называется величина, определяемая выражением:

$$\beta(r')_{\text{ref}} = \& \mu_{\bar{r}}(x_i, x_i) = \min_i \{1 - \mu_{r'}(x_i, x_i)\}.$$

Отношение называют **нечётко антирефлексивным**, если  $\beta(r')_{\text{ref}} \geq 0,5$ .

**Степенью симметричности** нечёткого отношения  $\alpha(r')_{\text{sym}}$  называется величина, определяемая выражением:

$$\alpha(r')_{\text{sym}} = \& (\mu_{r'}(x_i, x_j) \rightarrow \mu_{r'}(x_j, x_i)) = \& ((\neg \mu_{r'}(x_i, x_j) \vee \mu_{r'}(x_j, x_i)) = \min_i \{\max_j \{(1 - \mu_{r'}(x_i, x_j), \mu_{r'}(x_j, x_i))\}\}.$$

Отношение называют **нечётко симметричным**, если  $\alpha(r')_{\text{sym}} \geq 0,5$ .

**Степенью антисимметричности** нечёткого отношения  $\beta(r')_{\text{sym}}$  называется величина, определяемая выражением

$$\beta(r')_{\text{sym}} = \&_i \neg((\mu_{r'}(x_i, x_j) \& \mu_{r'}(x_j, x_i)) = \&_i ((\neg \mu_{r'}(x_i, x_j) \vee \neg \mu_{r'}(x_j, x_i)) = \\ \min_i \{ \max_j \{ (1 - \mu_{r'}(x_i, x_j)), (1 - \mu_{r'}(x_j, x_i)) \} \}.$$

Отношение называют **нечётко антисимметричным**, если  $\beta(r')_{\text{sym}} \geq 0,5$ .

**Степенью транзитивности** нечёткого отношения  $\alpha(r')_{\text{tr}}$  называется величина, определяемая выражением

$$\alpha(r')_{\text{tr}} = \&_i ((\mu_{r'}(x_i, x_j) \& \mu_{r'}(x_j, x_k)) \rightarrow \mu_{r'}(x_i, x_k)) = \\ \&_i (\neg(\mu_{r'}(x_i, x_j) \& \mu_{r'}(x_j, x_k)) \vee \mu_{r'}(x_i, x_k)) = \\ \&_i (\neg \mu_{r'}(x_i, x_j) \vee \neg \mu_{r'}(x_j, x_k) \vee \mu_{r'}(x_i, x_k)) = \\ \min_i \{ \max_j \{ (1 - \mu_{r'}(x_i, x_j)), (1 - \mu_{r'}(x_j, x_k)), \mu_{r'}(x_i, x_k) \} \}$$

Отношение называют **нечётко транзитивным**, если  $\alpha(r')_{\text{tr}} \geq 0,5$ .

Отношение, для которого  $\alpha(r')_{\text{ref}} \geq 0,5$ ,  $\alpha(r')_{\text{sym}} \geq 0,5$  и  $\alpha(r')_{\text{tr}} \geq 0,5$ , есть **отношение нечёткой эквивалентности**. Степень нечёткой эквивалентности определяется выражением:

$$\eta(r') = \alpha(r')_{\text{ref}} \& \alpha(r')_{\text{sym}} \& \alpha(r')_{\text{tr}} = \min \{ \alpha(r')_{\text{ref}}, \alpha(r')_{\text{sym}}, \alpha(r')_{\text{tr}} \} \geq 0,5.$$

Отношение, для которого  $\alpha(r')_{\text{ref}} \geq 0,5$ ,  $\beta(r')_{\text{sym}} \geq 0,5$  и  $\alpha(r')_{\text{tr}} \geq 0,5$ , есть **отношение нечёткого нестрогого порядка**. Степень нечёткого нестрогого порядка определяется выражением:

$$\eta(r') = \alpha(r')_{\text{ref}} \& \beta(r')_{\text{sym}} \& \alpha(r')_{\text{tr}} = \min \{ \alpha(r')_{\text{ref}}, \beta(r')_{\text{sym}}, \alpha(r')_{\text{tr}} \} \geq 0,5.$$

Отношение, для которого  $\beta(r')_{\text{ref}} \geq 0,5$ ,  $\beta(r')_{\text{sym}} \geq 0,5$  и  $\alpha(r')_{\text{tr}} \geq 0,5$ , есть **отношение нечёткого строгого порядка**. Степень нечёткого строгого порядка определяется выражением:

$$\eta(r') = \beta(r')_{\text{ref}} \& \beta(r')_{\text{sym}} \& \alpha(r')_{\text{tr}} = \min\{\beta(r')_{\text{ref}}, \beta(r')_{\text{sym}}, \alpha(r')_{\text{tr}}\} \geq 0,5.$$

**Пример 2.26.** Даны отношения  $r'_1$  и  $r'_2$ . Определить тип отношений.

$r'_1$	$x_1$	$x_2$	$x_3$	$x_4$	$r'_2$	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$
$x_1$	0,8	0,2	0,7	0,2	$x_1$	0	0	0	0	0,2
$x_2$	0,4	0,6	0,6	0,2	$x_2$	0,7	0,3	0,6	0,8	0,9
$x_3$	0,6	0,7	0,8	0,3	$x_3$	0,7	0,4	0,2	0,8	0,9
$x_4$	0,3	0,1	0,3	0,7	$x_4$	0,8	0	0	0	0,3
					$x_5$	1	0	0,1	0,7	0

Для отношения  $r'_1$  имеем

$$\alpha(r'_1)_{\text{ref}} = \min\{0,8, 0,6, 0,8, 0,7\} = 0,6,$$

$$\beta(r'_1)_{\text{ref}} = \min\{0,2, 0,4, 0,2, 0,3\} = 0,2$$

$$\alpha(r'_1)_{\text{sym}} = \min\{0,8, 0,6, 0,8, 0,7, 0,8, 0,7\} = 0,6,$$

$$\beta(r'_1)_{\text{sym}} = \min\{0,8, 0,4, 0,8, 0,3, 0,9, 0,7\} = 0,3,$$

$$\alpha(r'_1)_{\text{tr}} = \min\{0,7, 0,7, 0,7, 0,6, 0,7, 0,7\} = 0,6,$$

откуда

$$\eta(r'_1) = \alpha(r')_{\text{ref}} \& \alpha(r')_{\text{sym}} \& \alpha(r')_{\text{tr}} = \min\{0,6, 0,6, 0,6\} = 0,6,$$

$$\eta(r'_1) = \alpha(r')_{\text{ref}} \& \beta(r')_{\text{sym}} \& \alpha(r')_{\text{tr}} = \min\{0,6, 0,3, 0,6\} = 0,3,$$

$$\eta(r'_1) = \beta(r')_{\text{ref}} \& \beta(r')_{\text{sym}} \& \alpha(r')_{\text{tr}} = \min\{0,2, 0,3, 0,6\} = 0,2.$$

Ответ:  $r'_1$  есть отношение нечёткой эквивалентности.

Для отношения  $r'_2$  имеем

$$\alpha(r'_2)_{\text{ref}} = \min\{0, 0,3, 0,2, 0, 0\} = 0,$$

$$\beta(r'_2)_{\text{ref}} = \min\{1, 0,7, 0,8, 1, 1\} = 0,7,$$

$$\alpha(r'_2)_{\text{sym}} = \min\{1, 1, 1, 1, 0,4, 0,2, 0,1, 0,2, 0,1, 0,7\} = 0,1,$$

$$\beta(r'_2)_{\text{sym}} = \min\{1, 1, 1, 0,8, 0,6, 1, 1, 1, 0,9, 0,7\} = 0,6,$$

$$\alpha(r'_2)_{\text{tr}} = 0,7,$$

откуда

$$\eta(r'_2) = \alpha(r')_{\text{ref}} \& \alpha(r')_{\text{sym}} \& \alpha(r')_{\text{tr}} = \min\{0, 0,1, 0,7\} = 0,$$

$$\eta(r'_2) = \alpha(r')_{\text{ref}} \& \beta(r')_{\text{sym}} \& \alpha(r')_{\text{tr}} = \min\{0, 0,6, 0,7\} = 0,$$

$$\eta(r'_2) = \beta(r')_{\text{ref}} \& \beta(r')_{\text{sym}} \& \alpha(r')_{\text{tr}} = \min\{0,7, 0,6, 0,7\} = 0,6.$$

Ответ:  $r'_2$  есть отношение нечёткого строгого порядка.

### 2.1.2. Нечёткое исчисление

**Нечёткие высказывания** суть предложения  $A'$ , *степень истинности* которых  $\rho(A')$  или ложности  $\neg\rho(A')$  принимает значение на интервале  $[0, 1]$ . Например, высказывание: «сегодня хорошая погода». По каким признакам и кто дал такую оценку? Ведь «у природы нет плохой погоды...».

**Нечёткие предикаты** есть нечёткие высказывательные функции, аргументами которых являются *предметные переменные и предметные постоянные*, степень истинности которых также принадлежит интервалу  $[0, 1]$ . Например, высказывание «Петров выполняет ответственное задание». Как понимать меру ответственности? Какова степень истинности ответственного задания?

**Лингвистические переменные.** Нечёткими предметными переменными являются слова и словосочетания ес-

тественного языка, которые называют *лингвистическими переменными*. Например, 'ребенок', 'юноша', 'часто', 'регулярно', 'близко', 'далеко' и т.п. Лингвистическая переменная служит для качественного описания явления, факта или события. Лингвистические переменные объединяются по какому-либо признаку в нечёткий классы, которые называют *терм-множествами* и обозначают  $T(x)$ . Такими классами, например, могут быть «возраст», «количество», «частота», «расположение» и т. п.

Например,  $T_1$ («возраст человека») = {'ребенок', 'подросток', 'юноша', 'молодой человек', 'человек средних лет', 'пожилой человек', ...},  $T_2$ («количество продуктов») = {'очень мало', 'мало', 'много', 'очень много', ...},  $T_3$ («частота события») = {'всегда', 'часто', 'регулярно', 'редко', 'иногда', 'никогда', ...},  $T_4$ («расположение объектов») = {'вплотную', 'близко', 'рядом', 'далеко', ...}.

Для того, чтобы согласовывать мнения различных экспертов о степени истинности лингвистической переменной, чаще всего формируют некоторую шкалу на каком-то заданном интервале. Это позволяет для каждого факта количественно оценивать степень истинности высказывания.

**Нечёткие формулы.** При описании факта сложными нечёткими высказываниями также используют логические связки отрицания, конъюнкции, дизъюнкции, импликации и эквивалентности. Так формируются нечёткие логические формулы.

Степень истинности нечёткой формулы определяется степенью принадлежности результатов логических операций над нечёткими множествами и обозначается символом

$\rho$ . Например,  $\rho(\neg A') = (1 - \rho(A'))$ ,  $\rho(A' \& B') = \min\{\rho(A'), \rho(B')\}$ ,  $\rho(A' \vee B') = \max\{\rho(A'), \rho(B')\}$ ,  $\rho(A' \rightarrow B') = \max\{(1 - \rho(A')), \rho(B')\}$ ,  $\rho(A' \leftrightarrow B') = \min\{\max\{(1 - \rho(A')), \rho(B')\}, \max\{(1 - \rho(B')), \rho(A')\}\}$ .

Обратите внимание, что законы противоречия и «третьего не дано» для нечётких высказываний не выполняются. Если для четких высказываний:  $(A \& \neg A) = \text{л}$ ,  $(A \vee \neg A) = \text{и}$ , то для нечётких высказываний:

$\rho(A' \& \neg A') = \min\{\rho(A'), (1 - \rho(A'))\}$  и  $\rho(A' \vee \neg A') = \max\{\rho(A'), (1 - \rho(\neg A'))\}$ .

**Нечёткие правила вывода.** Так же как для четких высказываний, в логике нечётких высказываний выводима теорема:

$$\vdash F_1' \& F_1' \& \dots \& F_1' \rightarrow B',$$

но формулы посылок и заключения имеют значение степени истинности в интервале  $[0, 1]$ .

Это удобно объяснить с помощью нечёткого высказывания «если  $A'$ , то  $B'$ , иначе  $C'$ ». Нечёткое высказывание «если  $A'$ , то  $B'$ » можно определить как нечёткое отображение нечёткого высказывания  $A'$  на нечёткое высказывание  $B'$ , т. е.  $(A' \otimes B')$ , а нечёткое высказывание «если не  $A'$ , то  $C'$ » – как нечёткое отображение нечёткого высказывания  $\neg A'$  на нечёткое высказывание  $C'$ , т. е.  $(\neg A' \otimes C')$ . Объединение этих двух отображений есть формула данного нечёткого высказывания  $((A' \rightarrow B'), C') = ((A' \otimes B') \cup (\neg A' \otimes C'))$ . Если даны степени истинности нечётких высказываний  $\rho(A')$ ,  $\rho(B')$  и  $\rho(C')$ , то истинность высказывания «если  $A'$ , то  $B'$ , иначе  $C'$ » может быть оп-

ределена по формуле:

$$\rho((A' \rightarrow B'), C') = \max\{\min\{\rho(A'), \rho(B')\}, \min\{\rho(\bar{A}'), \rho(C')\}\}.$$

Пример. «Если сегодня вечером будет дождь ( $A'$ ), то завтра будет солнечная погода ( $B'$ ), иначе завтра будет пасмурный день ( $C'$ )».

Пусть  $\rho(A')=0,3$ ,  $\rho(B')=0,5$  и  $\rho(C')=0,2$ . Тогда  $\rho((A' \rightarrow B'), C') = \max\{\min\{\rho(A'), \rho(B')\}, \min\{\rho(\bar{A}'), \rho(C')\}\} = \max\{\min\{0,3, 0,5\}, \min\{0,7, 0,2\}\} = \max\{0,3, 0,2\} = 0,3$ , т. е. истинность сложного высказывания при данных значениях простых нечётких высказываний не более 0,3.

Если допустить, что  $\rho(C')=1$ , т.е. высказывание  $C'$  истинно для любых значений истинности высказывания  $\bar{A}'$ , то  $((A' \rightarrow B'), C') = ((A' \otimes B') \cup \bar{A}')$ . Степень истинности такого высказывания есть  $\rho(A' \rightarrow B') = \max\{\min\{\rho(A'), \rho(B')\}, \rho(\bar{A}')\}$ .

Пример. «Если сегодня вечером будет дождь ( $A'$ ), то завтра будет солнечная погода ( $B'$ )».

Пусть  $\rho(A')=0,3$  и  $\rho(B')=0,5$ . Тогда  $\rho(A' \rightarrow B') = \max\{\min\{\rho(A'), \rho(B')\}, \rho(\bar{A}')\} = \max\{\min\{0,3, 0,5\}, 0,7\} = \max\{0,3, 0,7\} = 0,7$ , т.е. истинность сложного высказывания «если сегодня вечером будет дождь, то завтра будет солнечная погода» при данных значениях простых нечётких высказываний равна 0,7.

Если даны множества нечётких высказываний  $\{A' = \rho(u_i)/u_i\}$  и  $\{B' = \rho(v_j)/v_j\}$  о фактах  $\{u_1, u_2, u_3, u_4, u_5,$

$u_6\}$  и  $\{v_1, v_2, v_3, v_4, v_5, v_6\}$ , то истинность  $\rho(A' \rightarrow B')$  необходимо определять для каждой пары  $(u_i, v_j)$  по формуле:  $\rho(A' \rightarrow B') = \max\{\min\{\rho(u_i), \rho(v_j)\}, \neg\rho(u_i)\}$ .

Пример. Пусть даны нечёткие высказывания

$$\rho(A') = \{0,6/u_1, 0,4/u_2, 0,8/u_3, 0,2/u_4, 1,0/u_5, 0,3/u_6\},$$

$$\rho(B') = \{0,9/v_1, 0,4/v_2, 1,0/v_3, 0,7/v_4, 0,3/v_5, 0,5/v_6\}.$$

Для каждой позиции таблицы  $\rho(A' \rightarrow B')$  нужно вычислить значение  $\rho(u_i \rightarrow v_j) = \max\{\min\{\rho(u_i), \rho(v_j)\}, \neg\rho(u_i)\}$ . Например,  $\rho(u_4 \rightarrow v_2) = \max\{\min\{0,2, 0,4\}, 0,8\} = \max\{0,2, 0,8\} = 0,8$ . Все результаты вычислений  $\rho(u_i \rightarrow v_j)$  сведены в таблицу 2.11.

**Таблица 2.11**

$\rho(A' \rightarrow B')$	$v_1$	$v_2$	$v_3$	$v_4$	$v_5$	$v_6$
$u_1$	0,6	0,4	0,6	0,6	0,4	0,5
$u_2$	0,6	0,6	0,6	0,6	0,6	0,6
$u_3$	0,8	0,4	0,8	0,7	0,3	0,5
$u_4$	0,8	0,8	0,8	0,8	0,8	0,8
$u_5$	0,9	0,4	1,0	0,7	0,3	0,5
$u_6$	0,7	0,7	0,7	0,7	0,7	0,7

Основным правилом вывода нечётких высказываний также является правило *modus ponens* (m.p.). Однако, чтобы учесть истинность посылки  $\rho(A')$  и истинность импликации  $\rho(A' \rightarrow B')$ , нужно вычислить истинность заключения  $\rho(B')$  как композицию нечёткого высказывания  $\rho(A')$  и нечёткого суждения  $\rho(A' \rightarrow B')$ , т. е.

$$\rho(B') = (\rho(A') \circ \rho(A' \rightarrow B')).$$

В этом случае степень истинности высказывания  $B'$  есть

$$\rho(B') = \rho(A') \circ \rho(A' \rightarrow B') = \max \{ \min \{ \rho(A'), \rho(A' \rightarrow B') \} \}.$$

Пример. Пусть  $\rho(A' \rightarrow B')$  задано предыдущей таблицей, а

$$\rho(A') = \{0,36/u_1, 0,16/u_2, 0,64/u_3, 0,04/u_4, 1,0/u_5, 0,09/u_6\}.$$

Тогда истинность заключения:

$$\rho(B') = \rho(A') \bullet \rho(A' \rightarrow B') = \{ \max_{v_1} \{ \min \{ 0,36/u_1, 0,6/(u_1, v_1) \}, \min \{ 0,16/u_2, 0,6/(u_2, v_1) \}, \min \{ 0,64/u_3, 0,8/(u_3, v_1) \}, \min \{ 0,04/u_4, 0,8/(u_4, v_1) \}, \min \{ 1,0/u_5, 0,9/(u_5, v_1) \}, \min \{ 0,09/u_6, 0,7/(u_6, v_1) \} \}, \max_{v_2} \{ \min \{ 0,36/u_1, 0,4/(u_1, v_2) \}, \min \{ 0,16/u_2, 0,6/(u_2, v_2) \}, \min \{ 0,64/u_3, 0,4/(u_3, v_2) \}, \min \{ 0,04/u_4, 0,8/(u_4, v_2) \}, \min \{ 1,0/u_5, 0,4/(u_5, v_2) \}, \min \{ 0,09/u_6, 0,7/(u_6, v_2) \} \}, \max_{v_3} \{ \min \{ 0,36/u_1, 0,6/(u_1, v_3) \}, \min \{ 0,16/u_2, 0,6/(u_2, v_3) \}, \min \{ 0,64/u_3, 0,8/(u_3, v_3) \}, \min \{ 0,04/u_4, 0,8/(u_4, v_3) \}, \min \{ 1,0/u_5, 1,0/(u_5, v_3) \}, \min \{ 0,09/u_6, 0,7/(u_6, v_3) \} \} \dots = \max_{v_1} \{ 0,36, 0,16, 0,64, 0,04, 0,9, 0,09 \}, \max_{v_2} \{ 0,36, 0,16, 0,4, 0,04, 0,4, 0,09 \}, \max_{v_3} \{ 0,36, 0,16, 0,64, 0,04, 1,0, 0,09 \}, \dots = \{ 0,9/v_1, 0,4/v_2, 0,64/v_3, 0,7/v_4, 0,64/v_5, 0,5/v_6 \}.$$

#### 4.4.2. Экспертные системы

Экспертные системы относятся к категории интеллектуальных вычислительных систем, которые используют знания специалистов о некоторой специализированной предметной области, хранят и накапливают эти знания, предлагают и объясняют решения конкретных задач или консультируют на уровне профессионала.

Существует много экспертных систем различного назначения, построенных на различных платформах. На-

пример, экспертная система PROSPECT EXPLORER использует нечёткую логику и семантические сети для консультации геологов при обнаружении горных аномалий. Система INTERNIST предназначена для врачебной диагностики и лечения заболеваний внутренних органов. Система FLEX нашла применение в различных финансовых системах, так как использует прямой и обратный методы при нечётких формулировках вопросов и нечётких правилах вывода. Система реального времени COMDALE/C обрабатывает неточные данные в условиях производства и консультирует в условиях непрерывного производства. Экспертная система MYCIN использует систему нечётких продукций для консультации врачей при диагностике и лечении инфекционных заболеваний. Оболочка экспертной системы МИКРОЭКСПЕРТ нашла широкое применение в различных сферах человеческой деятельности.

Идеализированная экспертная система содержит пять основных компонент (см. рис. 2.2): базу знаний (факты и правила вывода), интерфейс пользователя, подсистему логического вывода, блок извлечения и пополнения знаний и блок объяснения.



Рис. 2.2 Структура экспертной системы.

База знаний хранит и накапливает информацию о фактах, явлениях, событиях предметной области и о правилах, используемых экспертом-профессионалом при принятии решений. Интерфейс служит для взаимодействия пользователя или эксперта-профессионала с компьютером на проблемно-ориентированном языке. В интерфейсе происходит трансляция предложений этого языка на внутренний язык компьютера.

Подсистема логического вывода - это *высокоуровневый интерпретатор знаний*. Знания могут быть представлены в виде правил продукций (например, в системе MYCIN), деревьев решений (например, в системе ACLS), семантических сетей (например, в системе PROSPECTOR), исчисления предикатов и т.п.

В большинстве экспертных систем формируется упорядоченный список правил (продукций) и, в простейшем варианте, управление логическим выводом начинается с начала списка этих правил. Последовательно проверяются правила, пока не будет найдено то, для которого соответствуют условия, предусмотренные знаниями о фактах. За-

тем система пытается найти следующее правило, вернувшись к началу списка или продолжив поиск с того места, где он был прерван.

Блок объяснения служит для аргументации причинно-следственных связей и используемых правил решений. Поэтому все истинные правила и необходимые условия группируются в стеки контекстов и представляются пользователю.

Блок пополнения и корректировки базы знаний предназначен только для эксперта-профессионала, так как знания имеют слабо формализуемый характер по причинно-следственным связям. Поэтому самым узким местом экспертной системы является наполнение и корректировка правил базы знаний.

Одной из первых экспертных систем была «Микро-эксперт», которая представляет собой систему с открытой архитектурой.

При создании системы для какой-либо предметной области, прежде всего, формируют базу данных и знаний, содержащую описание фактов и соответствующие им значения атрибутов, описание правил и условий для истинности заключения. Каждое правило и необходимые атрибуты имеют индексированные карточки, которые облегчают сборку стека контекста для вынесения суждения о цели консультации. Первым этапом любой консультации является выбор цели. В процессе консультации необходимо собрать и запомнить последовательность всех известных фактов данной цели. Так формируется стек контекста данной цели. Затем формируется стек целей. Атрибуты цели, находящейся в последней вершине стека, представляют

текущую цель. Сначала система пытается найти карточку с правилом, которое в своем заключении содержит атрибуты текущей цели. Если правило найдено, то следует перейти к оценке этого правила по условной части. Все утверждения, стоящие в условной части, сравниваются с атрибутами описываемых фактов.

В процессе сравнения может возникнуть одна из трех ситуаций:

- если в текущем правиле один или несколько атрибутов отсутствуют в стеке контекста, то формируется новая цель поиска неизвестного атрибута и она направляется в стек целей,
- если текущее правило является ложным, т. е. условия противоречат фактам в стеке контекста, то данное правило удаляется и выбирается следующее правило,
- если текущее правило отвечает текущей цели, т. е. все атрибуты условной части соответствуют элементам стека контекста, то заключительная часть истинна.

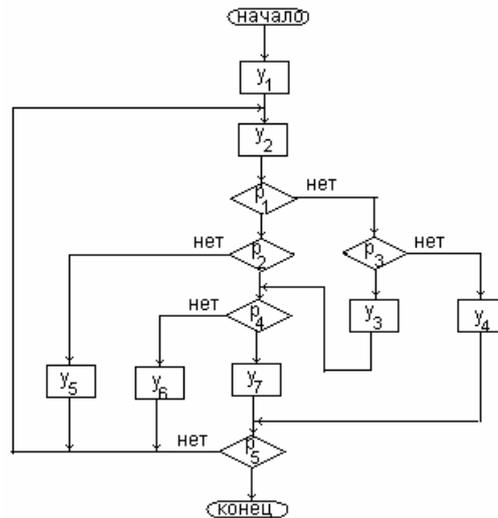


Рис. 2.3. Схема алгоритма ЭС «Микроэксперт».

Правило, которое было определено, как истинное, для текущей цели, помещается вместе с атрибутами в стек контекста решения для показа необходимых значений атрибутов. Далее следует переход к очередной цели. Если в стеке целей не осталось других целей, то задача решена.

На рис. 2.3 приведена укрупненная схема алгоритма логического вывода заключения экспертной системы «Микроэксперт», где

- $y_1$  – поместить цель в стек целей,
- $y_2$  – выбрать правило для оценки атрибутов цели,
- $y_3$  – поместить подсказку вместе с карточкой цели в стек контекста,
- $y_4$  – подсказка не найдена,
- $y_5$  – отложить карточку неиспользуемого правила,
- $y_6$  – поместить карточку неизвестного атрибута в стек целей вместе с карточкой правила,

$u_7$  – поместить карточки атрибутов из стека целей и карточку правила в стек контекста,

$p_1$  – правило выбрано?

$p_2$  – правило соответствует цели?

$p_3$  – есть ли подсказка в выборе правила?

$p_4$  – правило или подсказка содержит все атрибуты цели?

$p_5$  – стек целей пуст?

Другая экспертная система – MYCIN: диагностика и лечение инфекционных заболеваний - интересна правилом формирования заключения. База знаний содержит все свидетельства о заболевании пациента -  $F_i$  и правила диагностики. Данные о пациенте образуют так называемое контекстное дерево, структура которого унифицирована для всех больных. Для вывода заключения - гипотезы о заболевании  $V'$  - используют так называемый *коэффициент уверенности* КУ. Коэффициент уверенности – это разность между *мерой доверия* - МД и *мерой недоверия* - МНД в оценке истинности гипотезы:

КУ  $[(F' \rightarrow V'): F'] = \text{МД} [(F' \rightarrow V'): F'] - \text{МНД} [(F' \rightarrow V'): F']$ ,

где КУ  $[(F' \rightarrow V'): F']$  – коэффициент уверенности гипотезы  $(F' \rightarrow V')$  при истинности свидетельства  $F'$ ,

МД $[(F' \rightarrow V'): F']$  - мера доверия гипотезе  $(F' \rightarrow V')$  при заданной истинности свидетельства  $F'$ ,

МНД  $[(F' \rightarrow V'): F']$  - мера недоверия гипотезе  $(F' \rightarrow V')$  при заданной истинности свидетельства  $F'$ .

Меры доверия и недоверия определяются в диапазоне  $[0, 1]$ . Тогда КУ будет измеряться в диапазоне от -1 (абсолютная ложь) до +1 (абсолютная истина).

При наличии двух или нескольких свидетельств ( $F'_1$  и

$F'_2$ ) в системе MYCIN происходит уточнение мер доверия и недоверия гипотезе  $(F' \rightarrow B')$  при заданных значениях истинности двух свидетельств  $F'_1$  и  $F'_2$  по следующему правилу:

$\text{МД}[(F' \rightarrow B') : F'_1, F'_2] = \text{МД}[F'_1 \rightarrow B' : F'_1] + \text{МД}[F' \rightarrow B' : F'_2] \cdot (1 - \text{МД}[F' \rightarrow B' : F'_1]),$

$\text{МНД}[(F' \rightarrow B') : F'_1, F'_2] = \text{МНД}[(F' \rightarrow B') : F'_1] + \text{МНД}[(F' \rightarrow B') : F'_2] \cdot (1 - \text{МНД}[(F' \rightarrow B') : F'_1]).$  Смысл формулы состоит в том, что эффект второго свидетельства ( $F'_2$ ) на гипотезу  $(F' \rightarrow B')$  при заданном свидетельстве  $F'_1$  уточняет истинность гипотезы. По мере накопления свидетельств МД и МНД происходит постепенное уточнение гипотезы  $(F' \rightarrow B')$  до 1.

Вопросы и задачи

2.1.1 Дано универсальное множество  $U = \{u_1, u_2, u_3, u_4, u_5, u_6, u_7, u_8\}$  и множества  $A' = \{1/u_1, 0,1/u_2, 0,2/u_3, 0,3/u_4, 0,4/u_5\}$  и  $B' = \{0,1/u_1, 0,2/u_2, 0,3/u_6, 0,6/u_7, 0,8/u_8\}$ . Выполнить операции а) объединения, б) пересечения, с) дополнения, д) разности и е) симметрической разности.

2.1.2. Выполнить алгебраические операции а) объединения, б) пересечения, с) дополнения, д) разности и е) симметрической разности.

над нечёткими соответствиями  $h_1$  и  $h_2$ :

$h'_1$	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$	$h'_2$	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$
$x_1$	0,	0,	0,	0,	0,	$x_1$	0,	0,	0,	0,	0,
	2	4	6	2	4		4	2	8	2	4
$x_2$	0,	0,	0,	0,	0,	$x_2$	0,	0,	0,	0,	0,
	3	5	7	5	3		5	7	3	7	5
$x_3$	0,	0,	0,	0,	0,	$x_3$	0,	0,	0,	0,	0,
	2	5	4	5	2		5	2	6	2	5
$x_4$	0,	0,	0,	0,	0,	$x_4$	0,	0,	0,	0,	0,
	3	6	9	6	3		4	7	8	7	4

2.1.3. Выполнить композицию нечёткого множества  $X' = \{1/x_1, 0,1/x_2, 0,2/x_3, 0,3/x_4\}$  и соответствия  $h'$ , заданного таблицей:

$h'$	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$
$x_1$	0,2	0,4	0,6	0,2	0,4
$x_2$	0,3	0,5	0,7	0,5	0,3
$x_3$	0,2	0,5	0,4	0,5	0,2
$x_4$	0,3	0,6	0,9	0,6	0,3

2.1.4. Выполнить композицию двух нечётких соответствий  $h_1$  и  $h_2$ , заданных таблицами:

$h'_1$	$y_1$	$y_2$	$y_3$	$h'_2$	$z_1$	$z_2$	$z_3$	$z_4$
--------	-------	-------	-------	--------	-------	-------	-------	-------

$x_1$	1, 0, 0, 0 8 2	$y_1$	0, 0, 0, 0, 3 3 5 2
$x_2$	0, 1, 0, 2 0 4	$y_2$	0, 0, 0, 0, 2 2 4 3
$x_3$	0, 1, 0, 0 0 3	$y_3$	0, 0, 0, 0, 1 3 2 6
$x_4$	0, 0, 0, 2 9 5		
$x_5$	0, 0, 1, 3 7 0		

2.1.5. Выполнить алгебраические операции а) объединения, б) пересечения, с) дополнения, д) разности и е) симметрической разности и ф) композиции над  $r'_1$  и  $r'_2$ :

$r'_1$	$x_1$ $x_2$ $x_3$ $x_4$ $x_5$	$r'_2$	$x_1$ $x_2$ $x_3$ $x_4$ $x_5$
$x_1$	1 0,30,20,10,4	$x_1$	1 0,20,80,20,4
$x_2$	0,3 1 0,70,50,3	$x_2$	0, 1 0,80,70,2 5
$x_3$	0,20,5 1 0,50,2	$x_3$	0, 0,2 1 0,20,5 2
$x_4$	0,70,50,9 1 0,3	$x_4$	0, 0,70,8 1 0,4 4
$x_5$	0,60,70,20,3 1	$x_5$	0, 0,10,50,4 1 1

Расчетно-графическая работа

- 1) По отношениям  $r'_1$  и  $r'_2$  согласно заданию варианта нарисовать таблицы, содержащие четыре столбца и четыре строки, удалив из таблиц  $r'_1$  и  $r'_2$  четыре пары (столбец, строка),
- 2) выполнить операции  $(r'_1 \cup r'_2)$ ,  $(r'_1 \cap r'_2)$ ,  $(r'_1 \setminus r'_2)$ ,  $(r'_1 \circ r'_2)$ ,
- 3) включено ли отношение  $r'_1$  в отношение  $r'_2$  или наоборот?
- 2) какими свойствами обладают  $r'_1$  и  $r'_2$  и какому классу отношений они могут принадлежать?

$r'_1$	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	$r'_2$	$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	
$x_1$	0	0	0	0	0	0	0	0	$x_1$	0	0	0	0	0	0	0	0	7
$x_2$	0	0	0	0	0	0	0	0	$x_2$	0	0	0	0	0	0	0	0	4
$x_3$	0	0	0	0	0	0	0	0,	$x_3$	0	0	0	0	0	0	0	0	4
$x_4$	0	0	0	0	0	0	0	0	$x_4$	0	0	0	0	0	0	0	0	7
$x_5$	0	0	0	0	0	0	0	0	$x_5$	0	0	0	0	0	0	0	0	4
$x_6$	0	0	0	0	0	0	0	0	$x_6$	0	0	0	0	0	0	0	0	7
$x_7$	0,	0	0	0	0	0	0	0	$x_7$	0	0	0	0	0	0	0	0	8
$x_8$	0	0	0	0	0	0	0	0	$x_8$	0	0	0	0	0	0	0	0	7

Вари-	Удалить (столбец, строка)	Вари-	Удалить (столбец, строка)
1	для $r_1$ : (3, 1), (4, 2), (7, 5), (8, 6); для $r_2$ : (3, 1), (4, 2), (7, 5), (8, 6)	26	для $r_1$ : (1, 5), (4, 2), (7, 3), (8, 7); для $r_2$ : (1, 5), (4, 4), (7, 7), (8, 8)
2	для $r_1$ : (3, 1), (4, 3), (7, 7), (8, 8);	27	для $r_1$ : (1, 5), (4, 2), (7, 3), (8, 7);

	для $r_2$ : (3, 1), (4, 3), (7, 7), (8, 8)		для $r_2$ : (1, 5) (4, 4), (7, 6), (8, 8)
3	для $r_1$ : (3, 1), (4, 1), (7, 5), (8, 8); для $r_2$ : (3, 1), (4, 1), (7, 5), (8, 8)	28	для $r_1$ : (1, 5), (4, 2), (7, 3), (8, 7); для $r_2$ : (1, 5), (4, 4), (7, 5), (8, 8)
4	для $r_1$ : (3, 1), (4, 5), (7, 6), (8, 8); для $r_2$ : (3, 1), (4, 5), (7, 6), (8, 8)	29	для $r_1$ : (1, 5), (4, 2), (7, 3), (8, 7); для $r_2$ : (1, 5), (4, 4), (7, 5), (8, 7)
5	для $r_1$ : (3, 1), (4, 6), (7, 6), (8, 7); для $r_2$ : (3, 1), (4, 6), (7, 6), (8, 7)	30	для $r_1$ : (1, 5), (4, 2), (7, 3), (8, 7); для $r_2$ : (1, 5), (4, 4), (7, 5), (8, 6)
6	для $r_1$ : (3, 2), (4, 2), (7, 6), (8, 8); для $r_2$ : (3, 2), (4, 2), (7, 6), (8, 8)	31	для $r_1$ : (1, 4), (4, 2), (7, 3), (8, 6); для $r_2$ : (1, 4), (4, 4), (7, 7), (8, 8)
7	для $r_1$ : (3, 2), (4, 2), (7, 6), (8, 8); для $r_2$ : (3, 2), (4, 2), (7, 6), (8, 8)	32	для $r_1$ : (1, 4), (4, 2), (7, 3), (8, 6); для $r_2$ : (1, 4), (4, 4), (7, 6), (8, 8)
8	для $r_1$ : (3, 2), (4, 4), (7, 6), (8, 8); для $r_2$ : (3, 2), (4, 4), (7, 6), (8, 8)	33	для $r_1$ : (1, 4), (4, 2), (7, 3), (8, 6); для $r_2$ : (1, 4), (4, 4), (7, 5), (8, 8)
9	для $r_1$ : (3, 2), (4, 4), (7, 6), (8, 8); для $r_2$ : (3, 2), (4, 4), (7, 6), (8, 8)	34	для $r_1$ : (1, 4), (4, 2), (7, 3), (8, 6); для $r_2$ : (1, 4), (4, 4), (7, 5), (8, 7)

10	для $r_1$ : (3, 2), (4, 2), (7, 6), (8, 8); для $r_2$ : (3, 2), (4, 2), (7, 6), (8, 8)	35	для $r_1$ : (1, 4), (4, 2), (7, 3), (8, 6); для $r_2$ : (1, 4), (4, 4), (7, 5), (8, 6)
11	для $r_1$ : (3, 3), (4, 4), (7, 5), (8, 8); для $r_2$ : (3, 3), (4, 4), (7, 5), (8, 8)	36	для $r_1$ : (1, 3), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 8), (2, 5), (5, 6), (6, 8)
12	для $r_1$ : (3, 3), (4, 3), (7, 6), (8, 7); для $r_2$ : (3, 3), (4, 3), (7, 6), (8, 7)	37	для $r_1$ : (1, 8), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 8), (2, 5), (5, 3), (6, 8)
13	для $r_1$ : (3, 3), (4, 4), (7, 6), (8, 8); для $r_2$ : (3, 3), (4, 4), (7, 6), (8, 8)	38	для $r_1$ : (1, 8), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 8), (2, 2), (5, 3), (6, 8)
14	для $r_1$ : (3, 3), (4, 5), (7, 5), (8, 8); для $r_2$ : (3, 3), (4, 5), (7, 5), (8, 8)	39	для $r_1$ : (1, 8), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 8), (2, 2), (5, 1), (6, 8)
15	для $r_1$ : (3, 3), (4, 6), (7, 7), (8, 8); для $r_2$ : (3, 3), (4, 6), (7, 7), (8, 8)	40	для $r_1$ : (1, 8), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 8), (2, 2), (5, 1), (6, 7)
16	для $r_1$ : (3, 4), (4, 7), (7, 7), (8, 8); для $r_2$ : (3, 4), (4, 7), (7, 7), (8, 8)	41	для $r_1$ : (1, 7), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 7), (2, 2), (5, 3), (6, 7)
17	для $r_1$ : (3, 4), (4, 4), (7, 4), (8, 8);	42	для $r_1$ : (1, 7), (2, 2), (5, 7), (6, 8);

	для $r_2$ : (3, 4), (4, 4), (7, 4), (8, 8)		для $r_2$ : (1, 7), (2, 2), (5, 7), (6, 6)
18	для $r_1$ : (3, 4), (4, 4), (7, 5), (8, 8); для $r_2$ : (3, 4), (4, 4), (7, 5), (8, 8)	43	для $r_1$ : (1, 7), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 7), (2, 5), (5, 6), (6, 7)
19	для $r_1$ : (3, 4), (4, 7), (7, 5), (8, 8); для $r_2$ : (3, 4), (4, 7), (7, 5), (8, 8)	44	для $r_1$ : (1, 7), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 7), (2, 6), (5, 7), (6, 8)
20	для $r_1$ : (3, 4), (4, 2), (7, 4), (8, 8); для $r_2$ : (3, 4), (4, 2), (7, 4), (8, 8)	45	для $r_1$ : (1, 7), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 7), (2, 4), (5, 7), (6, 8)
21	для $r_1$ : (3, 5), (4, 8), (7, 3), (8, 8); для $r_2$ : (3, 5), (4, 8), (7, 3), (8, 8)	46	для $r_1$ : (1, 6), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 6), (2, 5), (5, 7), (6, 8)
22	для $r_1$ : (3, 5), (4, 8), (7, 6), (8, 8); для $r_2$ : (3, 5), (4, 8), (7, 6), (8, 8)	47	для $r_1$ : (1, 6), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 6), (2, 5), (5, 7), (6, 8)
23	для $r_1$ : (3, 5), (4, 8), (7, 5), (8, 8); для $r_2$ : (3, 5), 4, 8), (7, 5), (8, 8)	48	для $r_1$ : (1, 6), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 6), (2, 4), (5, 7), (6, 8)
24	для $r_1$ : (3, 5), (4, 4), (7, 3), (8, 8); для $r_2$ : (3, 5), (4, 4), (7, 3), (8, 8)	49	для $r_1$ : (1, 6), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 6), (2, 5), (5, 5), (6, 8)

25	для $r_1$ : (3, 5), (4, 4), (7, 5), (8, 6); для $r_2$ : (3, 5), (4, 4), (7, 5), (8, 6)	50	для $r_1$ : (1, 6), (2, 2), (5, 7), (6, 8); для $r_2$ : (1, 6), (2, 5), (5, 5), (6, 7)
----	---	----	---

## 2.2. Модальная логика

Если классическая логика оценивает высказывания, описывающие какую-то ситуацию, как истинные или ложные, то *модальная логика*<sup>\*</sup> указывает на *отношение субъекта* высказывания к истинности или ложности описываемой ситуации: что «должно быть», и что «может быть». Для этого вводят дополнительные операторы - **необходимо** и **возможно**.

Например, в высказывании «завтра должна быть хорошая погода» явно звучит отношение субъекта к ожидаемой на завтрашний день погоде: «**необходимо**, чтобы завтра была хорошая погода», Изменив чуть-чуть тот же пример, получим высказывание «если сегодня идет дождь, то, наверное, завтра будет хорошая погода». В этом высказывании уже звучит неуверенность субъекта: «**возможно** завтра или когда-нибудь будет хорошая погода». Или в высказывании «для проезда из Калининградской области в Россию на поезде необходимо иметь заграничный паспорт и визу иностранного государства» дано указание на **необходимость** приобрести соответствующий набор документов. В высказывании «проезд из Калининградской области в Россию возможен самолетом или паромом, минуя пересечение границ иностранного государства» дано перечисление **возможных** средств передвижения. Или в высказывании «для того, чтобы был хороший урожай необходимо выполнить своевременно все агрохимические и оросительные работы». В этом высказывании утверждается **необходимость** исполнения работ в прошлом времени для

---

\* - по материалам [6] и [22].

получения нужного результата в настоящем и будущем. Или в высказывании «неудовлетворительная оценка на экзаменах есть результат не исполнения лабораторных практикумов и не изучения дополнительной литературы». В этом высказывании указаны причины в прошлом времени, **возможно** определившие неудовлетворительную оценку знаний в настоящем.

Если при описании факта, имеющего место в настоящий момент времени, ввести понятие *Мир* как набор данных и условий и обозначить его символом « $r$ », то для множества фактов можно ввести понятие *множества Миров*, как множество наборов данных и множество условий и обозначить его символом « $W$ ». Если на множестве  $W$  ввести *отношение достижимости*  $q$  из настоящего Мира  $r \in W$  в Мир будущий или прошлый  $w \in W$  и два модальных оператора:  $\Box$  - «необходимо» и  $\Diamond$  - «возможно», то высказывание  $\Box A$  интерпретируется как истинное и читается «для всех  $w \in W$ , достижимых из  $r \in W$  (в прошлом или будущем), **необходимо**  $A$ », а высказывание  $\Diamond A$  интерпретируется как истинное и читается «для  $w \in W$ , достижимых из  $r \in W$  (в прошлом или будущем), **возможно**  $A$ ».

Если в классической логике высказывание  $A$  есть формула, то в модальной логике  $\Box A$  и  $\Diamond A$  также являются формулами.

Если ввести символ модальной импликации « $\Rightarrow$ », который для высказывания  $(A \Rightarrow B)$  означает  $\Box(A \rightarrow B)$ , и символ модальной эквивалентности « $\Leftrightarrow$ », который для высказывания  $(A \Leftrightarrow B)$  означает  $\Box(A \leftrightarrow B)$ , то для модальной логики справедливы правила и аксиомы [25]:

$$1 \quad \vdash \Box A \rightarrow A \qquad | \quad q \quad \vdash \Box(A \vee B) \leftrightarrow (\Box A \vee \Box B)$$

2. $\vdash \Box A \Leftrightarrow \neg \Box \neg A,$ 3. $\vdash \Box A \Leftrightarrow \neg \Box \neg A,$ 4. $\vdash \neg \Box A \Leftrightarrow \Box \neg A,$ 5. $\vdash \neg \Box A \Leftrightarrow \Box \neg A,$ 6. $\vdash \Box(A \& B) \Leftrightarrow$ $(\Box A \& \Box B),$ 7. $\vdash \Box(A \& B) \Rightarrow$ $(\Box A \& \Box B),$ 8. $\vdash (\Box A \vee \Box B)$ $\Rightarrow \Box(A \vee B),$	10. $\vdash \Box(A \& B) \Rightarrow \Box A,$ 11. $\vdash (A \Rightarrow B) \Rightarrow (\Box A \Rightarrow \Box B),$ 12. $\vdash (A \Rightarrow B) \Rightarrow (\Box A \Rightarrow \Box B),$ 13. $\vdash (A \Leftrightarrow B) \Rightarrow (\Box \Leftrightarrow \Box B),$ 14. $\vdash (A \Leftrightarrow B) \Rightarrow (\Box \Leftrightarrow \Box B),$ 15. $\vdash (A \Rightarrow B) \& \Box A \Rightarrow \Box B,$ 16. $\vdash (A \Rightarrow B) \& \Box A \Rightarrow \Box B.$
---	---

Истинность высказываний модальной логики доказывается законами классической логики с учетом приведенных правил и аксиом операторов модальности. Обратите внимание на правила 15 и 16. Это – *modus ponens* для операторов модальности. Правила 2, 3, 4 и 5 напоминают законы отрицания кванторов всеобщности и существования в исчислении предикатов (см. таблицу 1.11). Правило 10 напоминает правило П2 исчисления высказываний.

Модальный оператор «необходимо» ведет себя как квантор всеобщности для миров, достижимых из мира  $p$ , а оператор «возможно» - как квантор существования для некоторых миров, достижимых из мира  $p$ .

Например,  $(p \vdash \Box A) \Leftrightarrow (\forall_w((pqw) \Rightarrow w)) \vdash A$  есть высказывание «для настоящего Мира  $p$  выводимо высказывание **необходимо**  $A$  тогда и только тогда, когда для любого из Миров  $w \in W$ , если  $w$  достижимо из  $p$ , т.е.  $\forall_w((pqw) \Rightarrow w)$ , то  $A$  выводимо из  $w$ », а

$(p \vdash \Box A) \Leftrightarrow (\exists_w((pqw) \& w)) \vdash A$  есть высказывание «для настоящего Мира  $p$  выводимо высказывание **возможно**  $A$  тогда и только тогда, когда существуют Миры  $w \in W$ , дос-

тижимые из  $p$ , т.е.  $\exists_w((pqw)$ , и для этих  $w \in W$  выводимо  $A$ ».

Обратите внимание, оператор « $\square$ » требует истинности импликации в формуле  $\forall_w((pqw) \Rightarrow w)$ , а оператор « $\square$ » – истинности конъюнкции в формуле  $\exists_q((pqw) \& w)$ .

### 2.2.1. Темпоральная (или временная) логика\*.

Если Миры модальной логики понимать как срезы одного и того же Мира, но в разные моменты времени, то отношение достижимости Миров и выводимости высказываний можно разместить на временной оси.

Например,

$\square$  - «в следующий момент будет...»,  $\square$  – «когда-то будет...»,

$\square$  - «в предыдущий момент было...»,  $\square$  – «когда-то было...» и т. п.

В зависимости от интерпретации времени существуют логики «линейного времени» (linear time temporal logic) и «ветвящегося времени» (branching time temporal logic). В первом случае каждое состояние имеет ровно одного предшественника (или предшественника), а во втором – несколько. Так на основе модальной логики возникла **темпоральная (или временная) логика**.

Пусть дана программа  $\alpha$ , для которой высказывание  $A$  описывает начальные условия и входные данные. Назовем это высказывание **предусловием** программы  $\alpha$ . В результате исполнения программы  $\alpha$  формируется высказывание  $B$ , которое описывает результаты исполнения программы и

---

\* по материалам [6]

выходные данные. Назовем его *постусловием* программы  $\alpha$ . Тогда программа  $\alpha$  будет правильной по отношению к  $A$  и  $B$ , когда перед выполнением программы было истинным предусловие  $A$ , а после её исполнения стало истинным постусловие  $B$ . В этом случае справедлива запись  $\{A\}\alpha\{B\}$ , что на языке модальной логики есть  $A \vdash \Box B$ .

Проверка правильности программы - это задачи тестирования, а ее доказательство - это задачи верификации.

**Тестирование** – «способ семантической отладки программы, заключающийся в том, что программе предоставляется для обработки последовательность различных контрольных наборов данных (тестов), результат обработки которых известен ...» [17].

**Верификация программы** – «способ доказательства правильности программы, когда формулируются условия на входные и выходные данные, выбираются контрольные точки программы и формулируются утверждения о состоянии переменных в этих точках, ... затем каждому пути в программе между соседними точками сопоставляют так называемые условия правильности и доказывают истинность всех таких условий, а также завершаемость программы на данных, удовлетворяющих входным условиям» [17].

Если рассмотреть преемственность двух модулей программы  $\alpha$ , используя отношение достижимости  $q$ , то формула  $(AqB)$  есть высказывание «сейчас исполняется модуль  $A$ , а затем, т. е. в следующий (или предшествующий) момент времени, - модуль  $B$ ». При этом время имеет единственный начальный момент «сейчас». На языке модальной логики это есть высказывание  $(\Box A \Rightarrow \Box B) :=$  «вслед за

«А непосредственно следует В» или «А непосредственно предшествует В». Так цепочка  $(A_1 \supset (A_2 \supset (\dots (A_n \supset B)))$  моделирует последовательность исполнения модулей программы в логике с «линейным временем».

Логика с «ветвящимся временем» имеет более сложную структуру вывода заключения, но имеет широкое применение в проверке правильности параллельных программ, взаимодействующих между собой в процессе вычисления со ссылками на временные точки и/или временные интервалы.

Для логики с «ветвящимся временем» были приняты аксиомы:

$$\begin{array}{l} \vdash ((A_1 \vee A_2) \supset (B_1 \vee B_2)) \Leftrightarrow ((A_1 \supset B_1) \vee (A_1 \supset B_2) \vee (A_2 \supset B_1) \vee (A_2 \supset B_2)), \\ \vdash ((A \supset B_1) \& (A \supset B_2)) \Rightarrow (A \supset (B_1 \& B_2)), \\ \vdash A \Leftrightarrow ((A \supset (B \vee \neg B))), \\ \vdash \neg(A \supset (B \& \neg B)). \end{array}$$

При параллельных вычислениях, когда программа состоит из перемешанных между собой вычислительных операторов или модулей, необходимо различать «где» и «когда» *возможно* и/или *необходимо* переходить от исполнения операций одного вычислительного процесса к исполнению операций другого вычислительного процесса. То есть параллельные процессы могут включать секции, содержащие команды, критические для кооперирования этих процессов. Пока один процесс находится в своей критической секции и выполняет команды этой секции, второй процесс не должен входить в свою критическую секцию, поскольку изменения в ячейках памяти, происходящие при выполнении команд первого процесса, исказят вычисления, на которые ориентирован второй процесс.

Второй процесс должен ожидать пока содержимое используемых им ячеек не окажется востребованным. Ожидание формируется особыми командами программы, называемыми *семафорными инструкциями*. Так моделируется последовательность исполнения модулей вычислительных процессов с ветвящимся временем.

**Пример:** В параллельной программе имеются две критических секции ( $KC_1$  и  $KC_2$ ). Семафорной инструкцией является оператор «wait ... then ...» с семафорной переменной  $x_4$ . Пусть  $x_4:=1$  и вычисления начинает первый процесс. При выполнении оператора «wait ... then ...» значение  $x_4$  уменьшилось до 0. Тогда процесс 2 будет остановлен, так как семафорная инструкция под меткой  $m_0$  требует  $x_4>0$ . Следовательно, компьютер будет выполнять команды  $l_1, l_2, l_3$  процесса 1 и только после исполнения команды  $l_3$  компьютер может перейти к исполнению команд процесса 2.

Процесс 1		Процесс 2	
	$l_0: \text{wait } x_4>0$ $\text{then } x_4:=x_4-1;$		$m_0: \text{wait } x_4>0$ $\text{then } x_4:=x_4-1;$
$KC_1$	$l_1: t_1:=x_3*x_1;$ $l_2: x_3:=t_1;$ $l_3: x_4:=x_4+1;$	$KC_2$	$m_1: t_2:=x_3/x_2;$ $m_2: x_3:=t_2;$ $m_3: x_4:=x_4+1$
	$l_4: \text{end}$		$m_4: \text{end}$

Поэтому для нескольких параллельных процессов вводят позиционные переменные « $atl_i$ », которые

пробегают по выражениям вида « $i$ -ый процесс находится в метке  $l_i$ », а последовательность команд записывают теоремой временной логики:

$\vdash A_i \Rightarrow \Box \neg (at_{l_1} \& at_{l_2} \& \dots \& at_{l_i} \& \dots \& at_{l_n}) \& B_i$  и  $B_i \Rightarrow A_{i+1}$  где  $A_i$  и  $A_{i+1}$  – предусловия, а  $B_i$  – постусловие  $i$ -го процесса.

Стандарт SQL/92 предусмотрел возможное использование темпоральной логики и ввел типы данных «время-дата» и «интервал времени». Обычные таблицы реляционных баз являются плоскими, т. е. каждая строка таблицы соответствует текущему состоянию объекта предметной области. Однако на практике часто возникает потребность хранить в одной таблице информацию о состоянии разнообразных объектов в разные моменты времени и знать об этом (дата поступления товара и срок реализации и т.п.) для принятия соответствующих решений. До появления в языке SQL/92 данных темпорального типа это можно было сделать только одним способом: ввести дополнительный нестандартизованный и неунифицированный атрибут кортежа база данных, в котором хранились бы временные метки соответствующих записей. Однако, если нет их унифицированного представления, то вся логика обработки информации переходит в прикладную программу. В современных темпоральных реляционных СУБД состояния любого объекта, созданного в момент времени  $t_1$  и закончившего жизненный цикл в момент времени  $t_2$ , сохраняются и доступны на всем временном интервале  $[t_1, t_2]^*$ .

\* см. <http://zeus.sei.msu.ru:7000/database/sqlbook/>

### 2.2.2. Алгоритмическая логика\*\*

В 70-х годах XX века назрела проблема верификации программных комплексов. Необходимо было так описать семантику языка программирования, чтобы писать программу вместе с доказательством ее верности, а для этого необходимо построить логическую систему, опирающуюся на аксиоматику и правила вывода каждой конструктивной единицы языка программирования. Эти утверждения следовали из временной логики  $\{A\}\alpha\{B\}$ , когда «если предусловие  $A$  истинно перед выполнением  $\alpha$ , то постусловие  $B$  также будет истинным после её исполнения». Так начала формироваться *алгоритмическая логика*.

Пусть память компьютера  $L_0$  разделена на ячейки, каждая из которых имеет имя (идентификатор) и содержит натуральное число. Тогда состояние памяти до исполнения оператора есть предусловие  $A$  для исполнения оператора  $S$ , а после его исполнения состояние памяти компьютера есть постусловие  $B$  или «если до исполнения оператора  $S$  предусловие  $A$  было истинным, то после его исполнения - постусловие  $B$  также должно быть истинным».

Если дан оператор присваивания, конструкция которого есть  $S=(x:=\varepsilon)$ , где  $x$  – содержимое ячейки, имеющей имя  $x$ ,  $\varepsilon$  – значение арифметического выражения, находящееся в ячейке  $\varepsilon$ ,  $A$  – формула формальной арифметики, то состояние памяти до исполнения оператора должно быть  $A=A(x|\varepsilon)$ , а после его исполнения -  $B=A(x)$  или «если до исполнения оператора  $S$  предусловие  $A=A(x|\varepsilon)$  истинно,

---

\*\* по материалам [6]

то после его исполнения постусловие  $B=A(x)$  также истинно», т. е.  $\{A\}S\{B\}$  или  $\{A(x|\varepsilon)\}x:=\varepsilon\{A(x)\}$ . Условие  $(x|\varepsilon)$  означает замену переменной  $x$  на терм  $\varepsilon$ . Например, если дано  $A(x)=(x<2)$  и  $S=(x:=x+1)$ , то  $A(x|x+1)=(x+1<2)$ . То есть для того, чтобы после исполнения оператора присваивания стало истинным значение формулы  $A(x)=(x<2)$  необходимо и достаточно, чтобы до его исполнения было истинным значение формулы  $A(x|\varepsilon) = (x+1<2)$ , что возможно только при  $x=0$ .

Так формируется условие проверки оператора присваивания:

$$\{A(x|\varepsilon)\}x:=\varepsilon\{A(x)\}.$$

Если даны два оператора, которые выполняются последовательно, то этому соответствует логическая формула:

$$\frac{\{A_1\}S_1\{B_1\}, \{A_2\}S_2\{B_2\}, \{B_1\}\Rightarrow\{A_2\}}{\{A_1\}S_1S_2\{B_2\}},$$

Если дан условный оператор, конструкция которого есть

$IF A_1 \rightarrow S_1 \nabla A_2 \rightarrow S_2 \nabla \dots \nabla A_n \rightarrow S_n FI$ , где  $A_1, A_2, \dots, A_n$  – логические выражения, построенные при помощи логических связок на формулах формальной арифметики и отношениях  $\{=, \neq, <, \leq, \}$ ,  $S_i$  – операторы,  $(A_i \rightarrow S_i)$  – правила исполнения оператора  $S_i$  при истинном значении  $A_i$ ,  $\nabla$  – разделительный символ правил, то проверяется каждое логическое выражение  $A_i$  при настоящем состоянии памяти. Если ни одно из логических выражений не истинно, то фиксируется ошибка. Если некоторые  $A_i$  истинны, то выбирается одна из них (неважно как) и выполняется соответствующая последовательность операторов. Если каждая

из  $S_i$  есть  $\{A_i\}S_i\{B_i\}$  и логические выражения  $A_i$  истинны, то условный оператор описывается так:

$$\{(A_1 \Rightarrow A_i) \& (A_2 \Rightarrow A_2) \& \dots \& \{(A_n \Rightarrow A_n)\} \quad \text{IF} \quad A_1 \rightarrow S_1 \nabla \\ A_2 \rightarrow S_2 \nabla \dots \nabla A_n \rightarrow S_n \text{FI} \{B_i\}.$$

Если дан оператор цикла, конструкция которого есть  $DO A_1 \rightarrow S_1 \nabla A_2 \rightarrow S_2 \nabla \dots \nabla A_n \rightarrow S_n \text{OUT } B_1 \rightarrow T_1 \nabla B_2 \rightarrow T_2 \nabla \dots \nabla B_m \rightarrow T_m \text{OD}$ , где  $A_i$  и  $B_j$  – логические выражения, построенные при помощи логических связок на формулах формальной арифметики и отношениях  $\{=, \neq, <, \leq, \}$ ,  $S_i$  и  $T_j$  – операторы,  $A_i \rightarrow S_i$  и  $B_j \rightarrow T_j$  – правила исполнения операторов  $S_i$  и  $T_j$  при истинных значениях  $A_i$  и  $B_j$ , то прежде всего проверяются формулы  $A_1, A_2, \dots, A_n, B_1, B_2, \dots, B_m$  при настоящем состоянии памяти. Если ни одна из них не истинна, то фиксируется ошибка. Если некоторые из них истинны, то выбирается одна из них (неважно как). Если выбрана  $A_i$ , то выполняется соответствующая последовательность операторов  $S_i$  и выполнение цикла возобновляется. Если выбрана  $B_j$ , то выполняется соответствующая последовательность операторов  $T_j$  и выполнение цикла завершается.

Если каждая из  $S_i$  есть  $\{U_i\}S_i\{C_i\}$ , а  $T_j$  есть  $\{D_j\}T_j\{B_j\}$ , то выполнение одного цикла описывается логической формулой:

$$\{(A_1 \Rightarrow U_1) \& (A_2 \Rightarrow U_2) \& \dots \& (A_n \Rightarrow U_n) \& (B_1 \Rightarrow D_1) \& (B_2 \Rightarrow D_2) \& \\ \dots \& (B_m \Rightarrow D_m) \& \\ \& (A_1 \vee A_2 \vee \dots \vee A_n \vee B_1 \vee B_2 \vee \dots \vee B_m)\} \text{IF } A_1 \rightarrow S_1 \nabla A_2 \rightarrow S_2 \nabla \dots \nabla A_n \\ \rightarrow S_n \nabla B_1 \rightarrow T_1 \nabla B_2 \rightarrow T_2 \nabla \dots \nabla B_m \rightarrow T_m \text{FI} \{B\}.$$

Так были найдены логические условия для различных операторов и конструкций языка Pascal: от оператора присваивания до операторов формирования циклов, от вызова

процедур до сопрограмм, были доказаны аксиомы и правила вывода. Это позволило создать автоматическую систему верификации Pascal-программ.

### 2.2.3. Алгоритмическая логика Хоара\*

В трудах английского ученого Чарльза Хоара показано, что определение семантики языка программирования не в терминах программирования, а в терминах доказательства ее правильности, упрощает процесс построения программы. Основой для вывода правильных программ служат аксиомы Хоара (правила верификации). Особенностью этих аксиом является то, что они допускают интерпретации в терминах программных конструкций.

Например, такие:

$A_1$ : «если  $x$  после исполнения оператора присваивания будет содержать  $\varepsilon$ , то  $A$  будет истинно после его исполнения, если результат подстановки  $\varepsilon$  вместо  $x$  в  $A$  истинно перед его исполнением, т. е. истинна логическая формула:  $\{A(x|\varepsilon)\}x:=\varepsilon\{A(x)\}$ »,

$A_2$ : «если после исполнения оператора  $S$  из условия  $P$  логически следует  $B$ , значение которого истинно, то истинна логическая формула:  $\{A\}S\{P\} \& (P \Rightarrow B) \Rightarrow \{A\}S\{B\}$ »,

$A_3$ : «если до исполнения оператора  $S$  истинное значение условия  $P$  логически следует из  $A$ , то истинна логическая формула:  $\{P\}S\{B\} \& (A \Rightarrow P) \Rightarrow AS\{B\}$ »,

---

\* по материалам [6].

$A_4$ : «если  $S$  — это последовательность двух операторов  $S_1$  и  $S_2$ , то истинна логическая формула:  $\{A\}S_1\{P\} \& \{P\}S_2\{B\} \Rightarrow \{ASB\}$ ,

$A_5$ : «если дана краткая форма условного оператора  $S$ , то истинна логическая формула:

$\{P \& A\}S\{B\} \& (P \& \neg A \Rightarrow B) \Rightarrow \{P\} \text{ if } A \text{ then } S \{B\}$ »,

$A_6$ : «если дан альтернативный оператор, содержащий  $S_1$  и  $S_2$  (полная форма условного оператора), то истинна логическая формула:

$(\{P \& A\}S_1\{B\}) \& (\{P \& \neg A\}S_2\{B\}) \Rightarrow \{P\} A \text{ then } S_1 \text{ else } S_2 \{B\}$ »,

$A_7$ : «если дан оператор цикла until (до), то истинна логическая формула:  $\{A \& B\}S\{A\} \Rightarrow \{A\} \text{ repeat } S \text{ until } B\{A \& \neg B\}$ », предикат  $A$  истинный перед и после выполнения каждого шага цикла, называют инвариантом цикла,

$A_8$ : «если дан оператор цикла while (пока), то истинна логическая формула:  $\{A \& B\}S\{A\} \Rightarrow \{A\} \text{ while } B \text{ do } S\{A \& \neg B\}$ », предикат  $A$  истинный перед и после выполнения каждого шага цикла, также называют инвариантом цикла.

Аксиомы можно использовать для проверки согласованности передачи данных от оператора к оператору и для анализа структурных свойств текстов программ и для установления правильности программы.



## Ответы и решения

1.1.1a) Если ввести обозначения:

«подготовка специалистов высокой квалификации» - А,  
«подготовка специалистов на базе развития вузовской науки» - В,

«подготовка специалистов на базе усиления связи вузовской, академической и отраслевой науки» - С,

«подготовка специалистов на базе обеспечения единства научной и учебной работы» - D,

«подготовка специалистов на базе широкого привлечения студентов к научным исследованиям» - E, то суждение:

«подготовка специалистов высокой квалификации возможна лишь (...тогда и только тогда, когда – прим. авт.) на базе всемерного развития вузовской науки, усиления связи вузовской, академической и отраслевой науки, обеспечения единства научной и учебной работы, широкого привлечения студентов к научным исследованиям» имеет формулу  $F=(A\leftrightarrow V\&C\&D)$ .

1.1.1b) Если ввести обозначения:

«хлеба уцелеют в различных климатических условиях» -  $A_1$ ,

«хлеба уцелеют в различных погодных условиях» -  $A_2$ ,

«на полях выполнены все мелиоративные работы» - В,

«фермеры обанкротятся» -  $C_1$ ,

«фермеры оставят фермы» -  $C_2$ , то суждение:

«хлеба уцелеют в различных климатических и погодных условиях тогда и только тогда, когда на полях будут вы-

полнены все мелиоративные работы, а если хлеба не уцелеют, то фермеры обанкротятся и оставят фермы...»

имеет формулу

$$F = (((A_1 \& A_2) \leftrightarrow B) \& ((\neg A_1 \vee \neg A_2)) \rightarrow (C_1 \& C_2))).$$

1.1.1c) Если ввести обозначения:

«я поеду автобусом» - A,

«автобус опоздает» - B,

«я опоздаю на работу» - C,

«я стану огорчаться» - D,

«я сделаю в срок важную работу – E», то суждение

«если я поеду автобусом и автобус опоздает, то я опоздаю на работу, а если я опоздаю на работу, то я не сделаю в срок важную работу, а если я не сделаю в срок важную работу, то я стану огорчаться; следовательно, если я не поеду автобусом, то я сделаю в срок важную работу и не стану огорчаться»

имеет формальную запись: 
$$\frac{(A \& B \rightarrow C) \& (C \rightarrow \neg E) \& (\neg E \rightarrow D)}{(\neg A \rightarrow E \& \neg D)}.$$

1.1.1d) Если ввести обозначения:

«обвиняемый - исполнитель совершенного преступления» - A,

«обвиняемый - организатор совершенного преступления» - B»,

то суждение «обвиняемый может быть либо исполнителем, либо организатором совершенного преступления, но не тем и другим. Обвиняемый является организатором преступления. Следовательно, он не является исполнителем»

имеет формальную запись:  $\frac{\neg(A \leftrightarrow B), B}{\neg A}$ .

1.1.1e) Если ввести обозначения:

«Цезарь был суеверным» - A,

«Цезарь был осторожным» - B,

«Цезарь уступил просьбам Кальпурнии не идти в сенат» - C,

«Цезарь удалил Брута» - D, то суждение:

«если бы Цезарь был суеверен, то он уступил бы просьбам Кальпурнии не идти в сенат. Если бы он был осторожен, он удалил бы Брута. Но Цезарь не уступил просьбам Кальпурии, не удалил Брута» имеет формулу:  $(A \rightarrow C) \& (B \rightarrow D) \& (\neg C \& \neg D)$ .

1.1.2a)  $(A \vee B) \& (A \vee \neg B) \equiv A$ .

Решение:

- по закону дистрибутивности:

$$(A \vee B) \& (A \vee \neg B) = ((A \vee B) \& A) \vee ((A \vee B) \& \neg B),$$

$$((A \vee B) \& A) \vee ((A \vee B) \& \neg B) = ((A \& A \vee B \& A) \vee (A \& \neg B) \vee B \& \neg B),$$

- по законам идемпотентности и противоречия:

$$((A \& A \vee B \& A) \vee (A \& \neg B) \vee B \& \neg B) = A \vee B \& A \vee A \& \neg B,$$

- по закону поглощения:

$$A \vee B \& A \vee A \& \neg B = A \vee A \& \neg B = A,$$

т.е.  $A \equiv A$ . - *ч.т.д*

1.1.2b)  $(A \vee B) \& (B \vee C) \& (C \vee A) \equiv (A \& B) \vee (B \& C) \vee (C \& A)$ ,

Решение:

- по закону дистрибутивности:

$$(A \vee B) \& (B \vee C) \& (C \vee A) = A \& (B \vee C) \vee B \& C,$$

$$A \& (B \vee C) \vee B \& C = A \& B \vee A \& C \vee B \& C, \text{ т.е.}$$

$$A \& B \vee A \& C \vee B \& C = A \& B \vee A \& C \vee B \& C. - \text{ч.т.д.}$$

1.1.2с)

$$(A \vee B) \& (A \vee C) \& (B \vee D) \& (C \vee D) \equiv ((A \& D) \vee (B \& C)).$$

Решение:

- по закону дистрибутивности:
 
$$(A \vee B) \& (A \vee C) \& (B \vee D) \& (C \vee D) = (A \vee (B \& C)) \& (D \vee (B \& C)),$$

$$(A \vee (B \& C)) \& (D \vee (B \& C)) = ((A \& D) \vee (B \& C)), \text{ т.е.}$$

$$((A \& D) \vee (B \& C)) = ((A \& D) \vee (B \& C)). - \text{ч.т.д.}$$

1.1.2d)

$$(A \vee B) \& (B \vee C) \& (C \vee A)$$

$$\equiv ((A \& B) \vee (B \& C) \vee (C \& A)).$$

Решение:

- по закону дистрибутивности:
 
$$(A \vee B) \& (B \vee C) \& (C \vee A) = (A \& B \vee A \& C \vee B \& B \vee B \& C) \& (C \vee A) =$$

$$A \& B \& C \vee A \& A \& B \& \vee A \& C \& C \vee A \& A \& C \vee B \& B \& C \vee B \& A \vee$$

$$B \& C \& C \vee B \& C \& A,$$
- по законам идемпотенции и поглощения:
 
$$A \& B \& C \vee A \& A \& B \& \vee A \& C \& C \vee A \& A \& C \vee B \& B \& C \vee B \& A \vee$$

$$B \& C \& C \vee B \& C \& A =$$

$$A \& B \vee B \& C \vee C \& A. - \text{ч.т.д.}$$

1.1.2e)

$$(A \vee B \vee C) \& (B \vee C \vee D) \& (C \vee D \vee A)$$

$$\equiv ((A \& B) \vee (A \& D) \vee (B \& D) \vee C).$$

Решение:

- по закону дистрибутивности:
 
$$(A \& B \& C \vee B \& B \& C \& \vee C \& B \& C) \vee (A \& C \& C \vee B \& C \& C \vee C \& C$$

$$\begin{aligned} & \&C) \vee (A \&D \&C \vee \\ & B \&D \&C \vee C \&D \&C) \vee (A \&B \&D \vee B \&B \&D \vee C \&B \&D) \vee (A \&C \& \\ & D \vee B \&C \&D \vee \\ & C \&C \&D) \vee (A \&D \&D \vee B \&D \&D \vee C \&D \&D) \vee (A \&B \&A \vee B \&B \\ & \&A \vee C \&B \&A) \vee \\ & (A \&C \&A \vee B \&C \&A \vee C \&C \&A) \vee (A \&D \&A \vee B \&D \&A \vee C \&D \& \\ & A), \end{aligned}$$

- по законам идемпотенции и поглощения:

$$((A \&B) \vee (A \&D) \vee (B \&D) \vee C). - \text{ч.т.д.}$$

1.1.2f)  $(A \&B) \vee ((A \vee B) \& (\neg A \vee \neg B)) \equiv (A \vee B).$

Решение:

- по закону дистрибутивности:

$$\begin{aligned} & (A \&B) \vee ((A \vee B) \& (\neg A \vee \neg B)) = \\ & (A \&B) \vee (A \&\neg A) \vee (A \&\neg B) \vee (B \&\neg A) \vee (B \&\neg B) = \\ & A \&(B \vee \neg B) \vee (B \&\neg A) = A \vee (B \&\neg A) = \\ & (A \vee B) \&(A \vee \neg A) = (A \vee B) - \text{ч.т.д.} \end{aligned}$$

1.1.3a)  $((A \rightarrow B) \rightarrow (C \rightarrow \neg A)) \rightarrow (\neg B \rightarrow \neg C).$

Решение:

- удалить логические связки « $\rightarrow$ »:

$$(((A \rightarrow B) \rightarrow (C \rightarrow \neg A)) \rightarrow (\neg B \rightarrow \neg C)) = (\neg(\neg(\neg(A \vee B) \vee (\neg C \vee \neg A)) \vee (B \vee \neg C))),$$

- по закону де Моргана:

$$\begin{aligned} & (\neg(\neg(\neg(A \vee B) \vee (\neg C \vee \neg A)) \vee (B \vee \neg C))) = (\neg((A \&\neg B) \vee (\neg C \vee \neg A)) \\ & \vee (B \vee \neg C)) = \\ & (\neg(A \&\neg B) \&\neg(\neg C \vee \neg A)) \vee (B \vee \neg C) = (\neg A \vee B) \&C \&A \vee B \vee \neg C \end{aligned}$$

- по закону дистрибутивности:

$$(\neg A \vee B) \& C \& A \vee B \vee \neg C = \neg A \& C \& A \vee B \& C \& A \vee B \vee \neg C,$$

- по законам противоречия и дистрибутивности:

$$\neg A \& C \& A \vee B \& C \& A \vee B \vee \neg C = B \& C \& A \vee B \vee \neg C =$$

$$= (B \vee B \vee \neg C) \& (C \vee B \vee \neg C) \& (A \vee B \vee \neg C),$$

- по законам идемпотентности и исключенного третьего:

$$(B \vee B \vee \neg C) \& (C \vee B \vee \neg C) \& (A \vee B \vee \neg C) = (B \vee \neg C) \& (A \vee B \vee \neg C),$$

по закону поглощения:

$(B \vee \neg C) \& (A \vee B \vee \neg C) = (B \vee \neg C)$ . Это – элементарная дизъюнкция КНФ.

$$1.1.3b) (((((A \rightarrow B) \rightarrow \neg A) \rightarrow \neg B) \rightarrow \neg C) \rightarrow C).$$

Решение:

- устранить логические связки « $\rightarrow$ »:

$$(\neg(\neg(\neg(\neg(\neg A \vee B) \vee \neg A) \vee \neg B) \vee \neg C) \vee C),$$

- по закону де Моргана:

$$(\neg(\neg(\neg((A \& \neg B) \vee \neg A) \vee \neg B) \vee \neg C) \vee C) = (\neg(\neg(\neg(\neg B \vee \neg A) \vee \neg B) \vee \neg C) \vee C) =$$

$$= (\neg(\neg((B \& A) \vee \neg B) \vee \neg C) \vee C) = (\neg(\neg(A \vee \neg B) \vee \neg C) \vee C) = (\neg((\neg A \& B) \vee \neg C) \vee C) =$$

$$= (\neg(\neg A \& B) \& C) \vee C = (A \vee \neg B) \& C \vee C,$$

- по закону дистрибутивности:

$(A \vee \neg B) \& C \vee C = (A \& C \vee \neg B \& C \vee C)$ . Это – три элементарных конъюнкции ДНФ.

- по закону поглощения  $(A \& C \vee \neg B \& C \vee C) = C$ .

$$1.1.3c) (A \rightarrow (B \rightarrow C)) \rightarrow (A \rightarrow \neg C) \rightarrow (A \rightarrow \neg B).$$

Решение:

- устранить логические связки « $\rightarrow$ »:

$$\neg(\neg A \vee (\neg B \vee C)) \vee (\neg(\neg A \vee \neg C) \vee (\neg A \vee \neg B)),$$

- по закону де Моргана:

$(A \& \neg(\neg B \vee C)) \vee A \& C \vee (\neg A \vee \neg B) = (A \& B \& \neg C) \vee (A \& C) \vee \neg A \vee \neg B$ . Это – четыре элементарных конъюнкции ДНФ.

$$1.1.3d) (\neg(A \& (B \vee C)) \rightarrow ((A \& B) \vee C)).$$

Решение:

- устранить логические связки « $\rightarrow$ »:

$$((A \& (B \vee C)) \vee ((A \& B) \vee C)),$$

- по законам дистрибутивности, идемпотентности и поглощения:

$$A \& B \vee A \& C \vee A \& B \vee C = A \& B \vee C,$$

- по закону дистрибутивности:

$(A \vee C) \& (B \vee C)$ . Это – две элементарных дизъюнкции КНФ.

$$1.1.3e) (C \rightarrow A) \rightarrow (\neg(B \vee C) \rightarrow A).$$

Решение:

- устранить логические связки « $\rightarrow$ » и « $\neg$ »:

$$\neg(\neg C \vee A) \vee ((B \vee C) \vee A) = C \& \neg A \vee B \vee C \vee A,$$

- по законам дистрибутивности и третьего не дано:

$(C \vee B \vee C \vee A) \& (\neg A \vee B \vee C \vee A) = (A \vee B \vee C)$ . Это – одна элементарная дизъюнкция КНФ.

$$1.1.4a) \frac{\int_{A}^{B \& C} (A \vee B)}{(B \& C \vee B)} = B.$$

$$1.1.4b) \frac{\int_{-B \rightarrow \neg A}^{B \vee C} \int_A^{-B \rightarrow \neg A} (A \rightarrow (B \vee C))}{\int_{-B \rightarrow \neg A}^{B \vee C} (-B \rightarrow \neg A) \rightarrow (B \vee C) = (B \vee C) \rightarrow (B \vee C) = \text{и.}}$$

$$1.1.4c) \frac{\int_A^B (A \rightarrow B) \rightarrow (-B \rightarrow \neg A)}{(B \rightarrow B) \rightarrow (-B \rightarrow \neg B) = \neg(\neg B \vee B) \vee (B \vee \neg B) = \text{и.}}$$

$$1.1.4d) \frac{\int_C^B (A \vee B \vee C) \& (B \vee C \vee D) \& (A \vee C \vee D)}{(A \vee B \vee B) \& (B \vee B \vee D) \& (A \vee B \vee D) = (A \vee B) \& (B \vee D)}$$

$$1.1.4e) \frac{\int_{(C \rightarrow \neg A)}^{B \rightarrow C} (((A \rightarrow B) \rightarrow (C \rightarrow \neg A)) \rightarrow (\neg B \rightarrow \neg C))}{(((A \rightarrow B) \rightarrow (B \rightarrow C)) \rightarrow (\neg B \rightarrow \neg C)) = ((A \rightarrow C) \rightarrow (C \rightarrow B)) = (A \rightarrow B)}$$

$$1.1.5a) \frac{(A \vee B), (A \rightarrow C), (B \rightarrow D)}{(C \vee D)}$$

доказательство методом дедукции (см. рис. О.1):

1.  $(A \vee B)$  – посылка,
2.  $(A \rightarrow C)$  – посылка,
3.  $(B \rightarrow D)$  – посылка,
4.  $(\neg A \rightarrow B)$  – заключение по посылке 1 и правилам эквивалентных преобразований,
5.  $(\neg B \rightarrow A)$  - заключение по формуле 4 и правилу Пб и.в.,
6.  $(\neg B \rightarrow C)$  - заключение по формулам 2 и 5 и правилу П9 и.в.,
7.  $(\neg C \rightarrow B)$  - заключение по формуле 6 и правилу Пб и.в.,

8.  $(\neg C \rightarrow D)$  - заключение по формулам 3 и 7 и правилу П9 и.в.,
9.  $(C \vee D)$  – заключение по формуле 8 и правилам эквивалентных преобразований. - *ч.т.д.*.

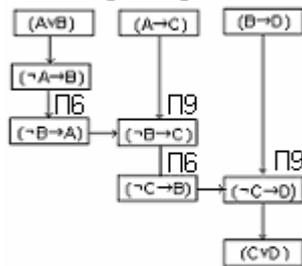


Рис. О.1. Граф дедуктивного вывода.

доказательство по принципу резолюции (см. рис. О.2):

- множество дизъюнктов посылок и отрицания заключения:  
 $K = \{(A \vee B), (\neg A \vee C), (\neg B \vee D), \neg C, \neg D\}$ ,
- вывод резольвент:  $\neg C \vee \neg A \vee C = \neg A$  - резольвента,  
 $\neg A \vee A \vee B = B$  - резольвента,  $B \vee \neg B \vee D = D$  – резольвента,  
 $D \vee \neg D = \square$  - пустая резольвента. - *ч.т.д.*

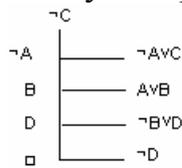


Рис. О.2. Граф вывода по принципу резолюции.

$$1.1.5b) \frac{((A \vee B) \rightarrow C), (C \rightarrow (D \vee E)), (E \rightarrow F), (\neg D \& \neg F)}{\neg A \& \neg C}$$

доказательство методом дедукции (см. рис. О.3):

1.  $(A \vee B) \rightarrow C$  – посылка,
2.  $C \rightarrow (D \vee E)$  – посылка,

3.  $E \rightarrow F$  – посылка,
4.  $\neg D \& \neg F$  – посылка
5.  $\neg D$  – заключение по посылке 4 и правилу П2 и.в.,
6.  $\neg F$  – заключение по посылке 4 и правилу П2 и.в.,
7.  $\neg E$  – заключение по посылке 3, формуле 6 и правилу m.t.,
8.  $\neg C$  – заключение по посылке 2, формулам 5, 7 и правилу m.t.,
9.  $(\neg D \& \neg E)$  – заключение по формулам 5 и 7 и правилу П1 и.в.,
10.  $(A \vee B) \rightarrow (D \vee E)$  – заключение по 1 и 2 и правилу П9 и.в.,
11.  $\neg(A \vee B)$  – заключение по формуле 9 и правилу m.t.,
12.  $(\neg A \& \neg B)$  – заключение по формуле 10 и закону де Моргана,
13.  $\neg A$  – заключение по формуле 11 и правилу П2 и.в.
14.  $\neg A \& \neg C$  – *ч.т.д.*

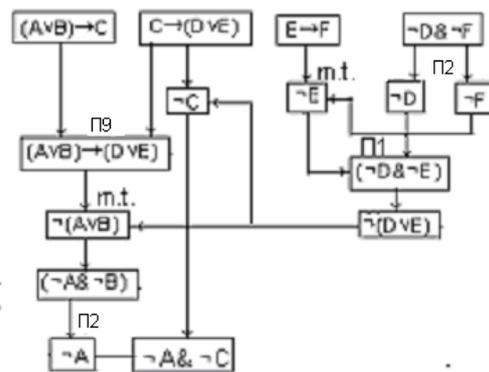


Рис. О.3. Граф дедуктивного вывода.

доказательство по принципу резолюции (см. рис. О.4):

- множество дизъюнктов посылок и отрицания заключения:

$K = \{(\neg A \vee C), (\neg B \vee C), (\neg C \vee D \vee E), \neg E \vee F, \neg D, \neg F, (A \vee C)\}$ ,  
вывод резольвент:  $A \vee \neg A \vee C = C$  - резольвента,  
 $C \vee \neg C \vee D \vee E = D \vee E$  - резольвента,  $D \vee E \vee \neg D = E$  - резольвента,  
 $E \vee \neg E \vee F = F$  резольвента,  $F \vee \neg F = \square$  - пустая резольвента. - *ч.т.д.*

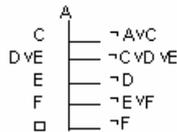


Рис.О.4. Граф вывода по принципу резолюции.

$$1.1.5c) \frac{(A \vee B), (A \rightarrow B), (B \rightarrow A)}{A \& B.}$$

доказательство методом дедукции (см. рис. О.5):

1.  $(A \vee B)$  – посылка,
2.  $(A \rightarrow B)$  – посылка,
3.  $(B \rightarrow A)$  – посылка,
4.  $(\neg A \rightarrow B)$  - заключение по посылке 1,
5.  $(\neg A \rightarrow A)$  - заключение по посылке 3, формуле 4 и правилу П9 и.в.,
6.  $(A \vee A)$  - заключение по формуле 5,
7.  $A$  - заключение по формуле 4 и закону идемпотентности,
8.  $(\neg B \rightarrow \neg A)$  – заключение по посылке 2 и правилу П6,
9.  $(\neg B \rightarrow B)$  – заключение по посылке 2, формуле 8 и правилу П9 и.в.,
10.  $(B \vee B)$  - заключение по формуле 9,

11. В - заключение по формуле 10 и закону идемпотентности,  
 12. (A&B) – заключение по формулам 7 и 11 и правилу П1 и.в. - *ч.т.д.*

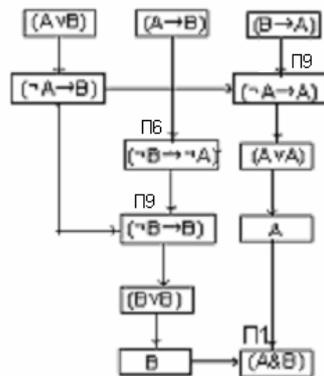


Рис. О.5. Граф дедуктивного вывода.

доказательство по принципу резолюции (см. рис. О.6):

- множество дизъюнктов посылок и отрицания заключения:

$$K = \{(A \vee B), (\neg A \vee B), (\neg B \vee A), (\neg A \vee \neg B)\},$$

вывод резольвент:  $(\neg A \vee \neg B) \vee (\neg B \vee A) = \neg B \vee \neg B = \neg B$  - резольвента,

$\neg B \vee (\neg A \vee B) = \neg A$  - резольвента,  $\neg A \vee (A \vee B) = B$  - резольвента,

$B \vee \neg B = \square$  - пустая резольвента. - *ч.т.д.*

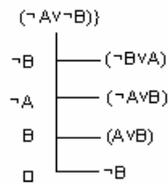


Рис. О.6. Граф вывода по принципу резолюции.

$$1.1.5d) \frac{((A \vee B) \rightarrow C \& D), ((D \vee B) \rightarrow F)}{(A \rightarrow F)}.$$

доказательство методом дедукции (см. рис. О.7):

1.  $((A \vee B) \rightarrow C \& D)$  – посылка,
2.  $((D \vee B) \rightarrow F)$  – посылка,
3.  $(\neg(A \vee B) \vee C \& D) = (\neg A \& \neg B) \vee (C \& D) =$   
 $(\neg A \vee C) \& (\neg B \vee C) \& (\neg A \vee D) \& (\neg B \vee D)$  – заключение по формуле 1,
4.  $(\neg A \rightarrow D) = (A \rightarrow D)$  - заключение по формуле 3 и правилу П2 и.в.,
5.  
 $(\neg(D \vee B) \vee F) = (\neg D \& \neg B) \vee F = (\neg D \vee F) \& (\neg B \vee F) = (D \rightarrow F) \& (B \rightarrow F)$  - заключение по формуле 2,
6.  $(D \rightarrow F)$  - заключение по формуле 6 и правилу П2 и.в.,
7.  $(A \rightarrow F)$  - заключение по формулам 4, 6 и правилу П9 и.в.  
 - *ч.т.д.*

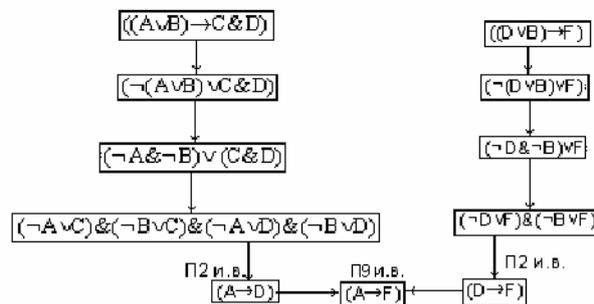


Рис.О.7. Граф дедуктивного вывода.

доказательство по принципу резолюции (см. рис. О.8):

- множество дизъюнктов посылок и отрицания заключения:  
 $K = \{(\neg A \vee C), (\neg B \vee C), (\neg A \vee D), (\neg B \vee D), (\neg D \vee F), (\neg B \vee F), A, \neg F\}$   
 вывод резольвент:  $(A \vee (\neg A \vee D)) = D$  - резольвента,

$(D \vee (\neg D \vee F)) = F$  – резольвента,  $F \vee \neg F = \square$  – пустая резольвента. - *ч.т.д.*

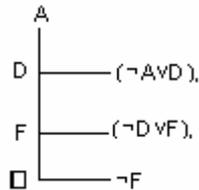


Рис. О.8. Граф вывода по принципу резолюции.

$$1.1.5e) \frac{(A \rightarrow B), (C \rightarrow D), (A \vee C), (A \rightarrow \neg D), (C \rightarrow \neg B)}{(D \leftrightarrow \neg B)}$$

доказательство методом дедукции (см. рис. О.9):

1.  $(A \rightarrow B)$  – посылка,
2.  $(C \rightarrow D)$  – посылка,
3.  $(A \vee C)$  – посылка,
4.  $(A \rightarrow \neg D)$  - посылка,
5.  $(C \rightarrow \neg B)$  - посылка,
6.  $(\neg B \rightarrow \neg A)$  - заключение по формуле 1 и правилу П6 и.в.,
7.  $(\neg A \rightarrow C)$  - заключение по формуле 3,
8.  $(\neg B \rightarrow C)$  - заключение по формулам 6, 7 и правилу П9 и.в.,
9.  $(\neg B \rightarrow D)$  - заключение по формулам 2, 8 и правилу П9 и.в.,
10.  $(D \rightarrow \neg A)$  - заключение по формуле 4 и правилу П6 и.в.,
11.  $(D \rightarrow C)$  - заключение по формулам 7, 10 и правилу П9 и.в.,
12.  $(D \rightarrow \neg B)$  - заключение по формулам 5, 11 и правилу П9 и.в.,

$(D \rightarrow \neg B) \& (D \rightarrow \neg B) = (D \leftrightarrow \neg B)$  - заключение по формулам 9, 12 и правилу П1 и.в. - *ч.т.д.*

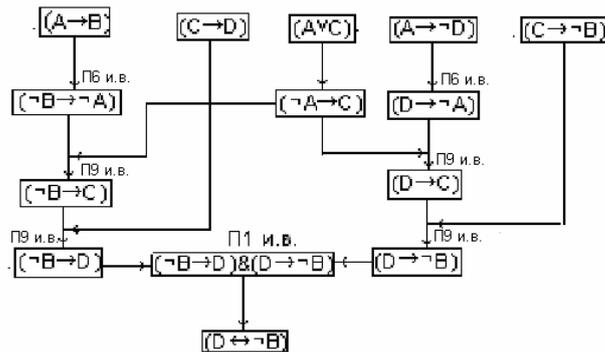


Рис.О.9. Граф дедуктивного вывода.

доказательство по принципу резолюции (см. рис.

О.10):

- множество дизъюнктов посылок и отрицания заключения:

$$K = \{(\neg A \vee B), (\neg C \vee D), (A \vee C), (\neg A \vee \neg D), (\neg C \vee \neg B), (D \vee \neg B), (B \vee \neg D)\}$$

- вывод резольвент:

$$(B \vee \neg D) \vee (\neg C \vee \neg B) = (\neg D \vee \neg C) \text{ - резольвента,}$$

$$(\neg D \vee \neg C) \vee (D \vee \neg B) = (\neg C \vee \neg B) \text{ - резольвента,}$$

$$(\neg C \vee \neg B) \vee (A \vee C) = (\neg B \vee A) \text{ - резольвента,}$$

$$\neg B \vee A) \vee (\neg A \vee B) = \square \text{ - пустая резольвента. - ч.т.д.}$$

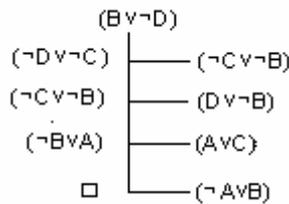


Рис. О.10. Граф вывода по принципу резолюции.

1.2.1a) Если ввести предикаты:

$P_1(x)$ : «быть судьёй»,

$P_2(x)$ : «быть юристом», то суждение:

«все судьи - юристы, но не все юристы – судьи»

имеет формулу:  $(\forall x(P_1(x) \rightarrow P_2(x))) \& (\neg \forall x(P_2(x) \rightarrow P_1(x)))$ .

1.2.1b) Если ввести предикаты:

$P_1(x)$ : «быть судьёй»,

$P_2(x)$ : «быть родственником потерпевшего»,

$P_3(x)$ : «x участвует в рассмотрении дела», то суждение:

«судья, являющийся родственником потерпевшего, не может участвовать в рассмотрении дела»

имеет формулу:  $\forall x(P_1(x) \& P_2(x) \rightarrow \neg P_3(x))$ .

1.2.1c) Если ввести предикаты:

$P_1(x)$ : «x привлекается к уголовной ответственности

$P_2(x, y)$ : «x совершил тайное хищение личного имущества у»,  $P_3(x)$ : «x есть обвиняемый», то суждение:

«к уголовной ответственности привлекаются лица, совершившие тайное похищение личного имущества граждан. Обвиняемый не совершал тайного похищения личного имущества граждан. Следовательно, обвиняемый x не может быть привлечен к уголовной ответственности» имеет формальную запись:

$$\frac{\exists x \forall y (P_2(x, y) \rightarrow P_1(x) \& P_3(x)), \exists x (P_3(x) \& \neg P_2(x))}{\exists x (P_3(x) \& \neg P_2(x) \rightarrow \neg P_1(x))}$$

1.2.1d) Если ввести предикаты:

$P_1(x)$ : «иск предъявлен дееспособным лицом»,

$P_2(x)$  := «суд рассматривает иск», то суждение:

«если иск предъявлен недееспособным лицом, то суд оставляет иск без рассмотрения. Иск предъявлен недееспособным лицом. Следовательно, суд оставляет иск без рассмотрения» имеет формальную запись:

$$\frac{\exists x (\neg P_1(x) \rightarrow \neg P_2(x)), \neg P_1(x)}{\neg P_1(x)}$$

1.2.1e) Если ввести предикаты:

$P_1(x)$  := «человек может быть вполне беспристрастным»,

$P_2(x)$  := «человек может быть юристом», то суждение:

«ни один человек не может быть вполне беспристрастным. Каждый юрист – человек. Следовательно, ни один юрист не может быть вполне беспристрастным» имеет формальную запись:

$$\frac{\neg \exists x (P_1(x)), \exists x (P_2(x))}{\neg \exists x (P_2(x) \rightarrow P_1(x))}$$

1.2.2a)  $\exists x (\forall y (P_1(x, y))) \& \exists x (\forall y (P_2(x, y)))$ ,

- заменить связанную левым квантором  $\exists x$  переменную  $x=v$ :

$$\exists v (\forall y (P_1(v, y))) \& \exists x (\forall y (P_2(x, y))),$$

- заменить связанную левым квантором  $\forall y$  переменную  $y=w$ :

$$\exists v (\forall w (P_1(v, w))) \& \exists x (\forall y (P_2(x, y))),$$

- вынести кванторы в префикс:

$$\exists v \forall w \exists x \forall y (P_1(v, w) \& P_2(x, y)). - \text{ч.т.д.}$$

1.2.2b)  $\exists x (\forall y (P_1(x, y))) \vee \exists x (\forall y (P_2(x, y)))$ ,

- применить закон дистрибутивности:

$$\exists_x(\forall_y(P_1(x, y)) \vee \forall_y(P_2(x, y))).$$

- заменить связанную левым квантором  $\forall_y$  переменную  $y=w$ :

$$\exists_x(\forall_w(P_1(x, w)) \vee \forall_y(P_2(x, y))),$$

- вынести кванторы  $\forall_w$  и  $\forall_y$  в префикс:

$$\exists_x \forall_w \forall_y (P_1(x, w) \vee P_2(x, y)). - \text{ч.т.д.}$$

$$1.2.2c) \exists_x(\forall_y(P_1(x, y))) \rightarrow \exists_x(\forall_y(P_2(x, y))),$$

- удалить « $\rightarrow$ »:

$$\exists_x(\neg \forall_y(P_1(x, y))) \vee \exists_x(\forall_y(P_2(x, y))),$$

- опустить « $\neg$ » по закону де Моргана:

$$\exists_x(\exists_y(\neg P_1(x, y))) \vee \exists_x(\forall_y(P_2(x, y))),$$

- применить закон дистрибутивности:

$$\exists_x(\exists_y(\neg P_1(x, y)) \vee \forall_y(P_2(x, y))),$$

- заменить связанную левым квантором  $\exists_y$  переменную  $y=w$ :

$$\exists_x(\exists_w(\neg P_1(x, w)) \vee \forall_y(P_2(x, y))),$$

- вынести кванторы  $\exists_w$  и  $\forall_y$  в префикс:

$$\exists_x \exists_w \forall_y (\neg P_1(x, w) \vee P_2(x, y)). - \text{ч.т.д.}$$

$$1.2.2d) \forall_x(\forall_y(P_1(x, y))) \rightarrow \exists_x(\exists_y(P_2(x, y))).$$

- удалить « $\rightarrow$ »:

$$\neg \forall_x(\forall_y(P_1(x, y))) \vee \exists_x(\exists_y(P_2(x, y))),$$

- опустить « $\neg$ » по закону де Моргана:

$$\exists_x(\neg \forall_y(P_1(x, y))) \vee \exists_x(\exists_y(P_2(x, y))) = \exists_x(\exists_y(\neg P_1(x, y))) \vee \exists_x(\exists_y(P_2(x, y))),$$

- применить закон дистрибутивности:

$$\exists_x(\exists_y(\neg P_1(x, y)) \vee \exists_y(P_2(x, y))),$$

- применить закон дистрибутивности:

$\exists_x \exists_y (\neg P_1(x, y) \vee P_2(x, y))$ . - *ч.т.д.*

1.2.2e)  $\forall_x (P_1(x) \vee (P_2(x) \rightarrow (\forall_x (P_1(x)) \vee \forall_y (P_2(y))))$ .

- удалить « $\rightarrow$ » по законам алгебры логики:

$\neg \forall_x (P_1(x) \vee (P_2(x) \vee \forall_x (P_1(x)) \vee \forall_y (P_2(y))))$ ,

- опустить « $\neg$ » по закону де Моргана:

$\exists_x \neg (P_1(x) \vee (P_2(x) \vee \forall_x (P_1(x)) \vee \forall_y (P_2(y))))$ ,

$\exists_x (\neg P_1(x) \& \neg P_2(x) \vee \forall_x (P_1(x)) \vee \forall_y (P_2(y)))$ ,

- заменить связанную левым квантором  $\exists_x$  переменную

$x=v$ :

$\exists_v (\neg P_1(v) \& \neg P_2(v) \vee \forall_x (P_1(x)) \vee \forall_y (P_2(y)))$ ,

- вынести кванторы  $\exists_v$ ,  $\forall_x$ , и  $\forall_y$  в префикс:

$\exists_v \forall_x \forall_y (\neg P_1(v) \& \neg P_2(v) \vee P_1(x) \vee P_2(y))$ ,

- применить закон дистрибутивности:

$\exists_v \forall_x \forall_y (\neg P_1(v) \vee P_1(x) \vee P_2(y)) \& (\neg P_2(v) \vee P_1(x) \vee P_2(y))$ . - *ч.т.д.*

1.2.3a)  $\exists_x \forall_y (P_1(x, y)) \& (\forall_x \exists_y (P_2(x, y)))$ .

- подстановка  $\int_x^a \exists_y \forall_y (P(x, y))$  и  $\int_y^{f(x)} \forall_x \exists_y (P_2(x, y))$ :

$\forall_y (P_1(a, y)) \& \forall_x (P_2(x, f(x)))$ ,

- вынести квантор  $\forall_x$  в префикс:

$\forall_y \forall_x (P_1(a, y) \& P_2(x, f(x)))$ . - *ч.т.д.*

1.2.3b)  $(\exists_x \forall_y \exists_z \forall_w (P(x, y, z, w)))$

- подстановка  $\int_x^a \int_z^{f(y)} \exists_y \forall_w (P(x, y, z, w))$ :

$\forall_y \forall_w (P(a, y, f(y), w))$ . - *ч.т.д.*

1.2.3c)  $\exists_x (P_1(x)) \& \exists_x (P_2(x)) \rightarrow \exists_x (P_1(x) \& P_2(x))$ .

- удалить « $\leftrightarrow$ »:  
 $\neg(\exists_x(P_1(x)) \& \exists_x(P_2(x))) \vee \exists_x(P_1(x) \& P_2(x)),$
- опустить « $\neg$ » по закону де Моргана:  
 $\forall_x(\neg P_1(x)) \vee \forall_x(\neg P_2(x)) \vee \exists_x(P_1(x) \& P_2(x)),$
- заменить связанную левым квантором  $\forall_x$  переменную  $x=v$ :  
 $\forall_v(\neg P_1(v)) \vee \forall_x(\neg P_2(x)) \vee \exists_x(P_1(x) \& P_2(x)),$
- заменить связанную левым квантором  $\forall_x(\neg P_2(x))$  переменную  $x=w$ :  
 $\forall_v(\neg P_1(v)) \vee \forall_w(\neg P_2(w)) \vee \exists_x(P_1(x) \& P_2(x)),$
- вынести кванторы  $\forall_v, \forall_w$  и  $\exists_x$  в префикс:  
 $\forall_v \forall_w \exists_x(\neg P_1(v) \vee \neg P_2(w) \vee P_1(x) \& P_2(x)),$
- подстановка  $\int_x^{f(v,w)} (\neg P_1(v) \vee \neg P_2(w) \vee P_1(x) \& P_2(x))$ :  
 $\forall_v \forall_w(\neg P_1(v) \vee \neg P_2(w) \vee P_1(f(v,w)) \& P_2(f(v,w))).$
- применить закон дистрибутивности  
 $\forall_v \forall_w((\neg P_1(v) \vee \neg P_2(w) \vee P_1(f(v,w))) \& (\neg P_1(v) \vee \neg P_2(w) \vee P_2(f(v,w))))$ . - *ч.т.д.*

1.2.3d)  $\forall_x \forall_y (P_1(x, y) \rightarrow \exists_x \exists_y (P_2(x, y)))$ .

- удалить символ « $\rightarrow$ »:  
 $\neg(\forall_x \forall_y (P_1(x, y)) \vee \exists_x \exists_y (P_2(x, y))) = \exists_x \exists_y (\neg P_1(x, y)) \vee \exists_x \exists_y (P_2(x, y)),$
- применить закон дистрибутивности:  
 $\exists_x \exists_y (\neg P_1(x, y) \vee P_2(x, y)),$
- заменить предметные переменными предметными постоянными  $x=a, y=b$   $\neg P_1(a, b) \vee P_2(a, b)$ . - *ч.т.д.*

1.2.4a)  $\exists_x (P_1(x)) \& \exists_x (P_2(x)) \rightarrow \exists_x (P_1(x) \& P_2(x))$  не тождественно истинная, так как  $\exists_x (P_1(x)) \& \exists_x (P_2(x)) \neq \exists_x (P_1(x) \& P_2(x))$ .

1.2.4b)  $\forall_y (P_1(y) \& P_2(y)) \rightarrow \forall_y (P_1(y) \& P_2(y))$  тождественно истинная, так как  $\forall_y (P_1(y) \& P_2(y)) = \forall_y (P_1(y) \& P_2(y))$ .

1.2.4c)  $\exists_x (P_1(x) \vee P_2(x)) \rightarrow (\exists_x (P_1(x)) \vee \exists_x (P_2(x)))$  тождественно истинная, так как  $\exists_x (P_1(x) \vee P_2(x)) = (\exists_x (P_1(x)) \vee \exists_x (P_2(x)))$ .

1.2.4d)  $\forall_y (P_1(y) \vee P_2(y)) \rightarrow (\forall_y (P_1(y)) \vee \forall_y (P_2(y)))$  не тождественно истинная, так как  $\forall_y (P_1(y) \vee P_2(y)) \neq (\forall_y (P_1(y)) \vee \forall_y (P_2(y)))$ .

$$1.2.5a) \frac{\forall_x (P_1(x) \rightarrow \neg P_2(x)), \forall_x (P_3(x) \rightarrow P_1(x))}{\forall_x (P_3(x) \rightarrow \neg P_2(x))}$$

доказательство методом дедукции (см. рис. О.11):

1.  $\forall_x (P_1(x) \rightarrow \neg P_2(x))$  – посылка,
2.  $\forall_x (P_3(x) \rightarrow P_1(x))$  – посылка,
3.  $(P_1(t) \rightarrow \neg P_2(t))$  – заключение по 1. и правилу П1 и.п.,
4.  $(P_3(t) \rightarrow P_1(t))$  – заключение по 2 и правилу П1 и.п.,
5.  $(P_3(t) \rightarrow \neg P_2(t))$  – заключение по 3. и 4. и правилу П9 и.в.
6.  $\forall_x (P_3(x) \rightarrow \neg P_2(x))$  – заключение по 5 и правилу П2 и.п. *ч.т.д.*

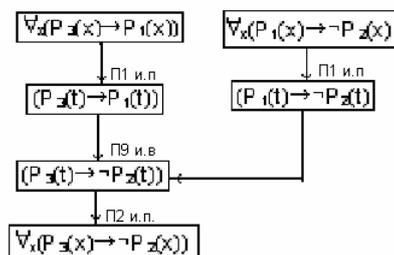


Рис. О.11. Граф дедуктивного вывода.

доказательство по принципу резолюции (см. рис. О.12):

• выписать множество дизъюнктов посылок и отрицания заключения:

$$K = \{(\neg P_3(t) \vee P_1(t)), (\neg P_1(t) \vee \neg P_2(t)), P_3(t), P_2(t)\}.$$

• вывод резольвент:  $P_3(t) \vee (\neg P_3(t) \vee P_1(t)) = P_1(t)$  - резольвента,  $P_1(t) \vee (\neg P_1(t) \vee \neg P_2(t)) = \neg P_2(t)$  - резольвента,  $\neg P_2(t) \vee P_2(t) = \square$  - пустая резольвента. - *ч.т.д.*

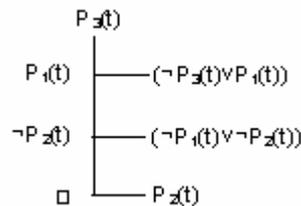


Рис. О.12. Граф вывода по принципу резолюции.

$$1.2.5b) \frac{\forall_x (P_1(x, y) \rightarrow P_2(x) \& P_3(x)), \exists_x (P_1(x, y) \& P_4(x))}{\exists_x (P_4(x) \& P_3(x))}.$$

доказательство методом дедукции (см. рис. О.13):

1.  $\forall_x (P_1(x, y) \rightarrow P_2(x) \& P_3(x))$  - посылка,
2.  $\exists_x (P_1(x, y) \& P_4(x))$  - посылка,
3.  $(P_1(t, y) \rightarrow P_2(t) \& P_3(t))$  - заключение по 1 и правилу П1 и.п.,
4.  $(P_1(a, y) \& P_4(a))$  - заключение по 2 и по правилу П3 и.п.,
5.  $P_1(a, y)$  - заключение по 4 и правилу П2 и.в.,
6.  $P_4(a)$  - заключение по 4 и правилу П2 и.в.,
7.  $P_2(a) \& P_3(a)$  - заключение по 3 и 5 и правилу т.р. при подстановке  $t=a$ ,
8.  $P_2(a)$  - заключение по 7 и правилу П2 и.в.,
9.  $P_3(a)$  - заключение по 7 и правилу П2 и.в.,
10.  $P_3(a) \& P_4(a)$  - заключение по 6 и 9 и правилу П1 и.в.,

11.  $\exists_x (P_3(x) \& P_4(x))$  - заключение по 10 и правилу ПЗ и.п. - *ч.т.д.*

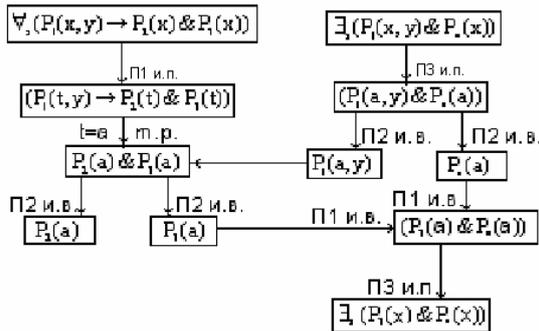


Рис. О.13. Граф дедуктивного вывода.

доказательство по принципу резолюции (см. рис. О.14):

- **выписать множество дизъюнктов посылок и отрицания заключения:**

$$K = \{(\neg P_1(t, y) \vee P_2(t)), (\neg P_1(t, y) \vee P_3(t)), P_1(a, y), P_4(a), (\neg P_4(a) \vee \neg P_3(a))\}.$$

- **вывод резольвент:**  $(\neg P_4(a) \vee \neg P_3(a)) \vee P_4(a) = \neg P_3(a)$  – резольвента,  $\neg P_3(a) \vee (\neg P_1(a, y) \vee P_3(a)) = \neg P_1(a, y)$  – резольвента при подстановке  $t=a$ ,  $\neg P_1(a, y) \vee P_1(a, y) = \square$  - пустая резольвента. - *ч.т.д.*

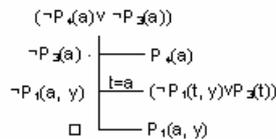


Рис. О.14. Граф вывода по принципу резолюции.

$$1.2.5c) \frac{\exists_x (P_1(x) \& \forall_y (P_2(x) \rightarrow P_3(x, y))), \forall_x (P_1(x) \rightarrow \forall_y (P_4(y) \rightarrow \neg P_3(x, y)))}{\forall_y (P_2(y) \rightarrow \neg P_4(y))}.$$

доказательство методом дедукции (см. рис. О.15):

1.  $\exists_x (P_1(x) \& \forall_y (P_2(x) \rightarrow P_3(x, y)))$  – посылка,
2.  $\forall_x (P_1(x) \rightarrow \forall_y (P_4(y) \rightarrow \neg P_3(x, y)))$  - посылка,
3.  $(P_1(a) \& \forall_y (P_2(a) \rightarrow P_3(a, y)))$  - заключение по 1 и правилу ПЗ,
4.  $\forall_y (P_1(a) \& (P_2(a) \rightarrow P_3(a, y)))$  - заключение по 3 и правилу П5,
5.  $(P_1(a) \& (\neg P_2(a) \vee P_3(a, t)))$  - заключение по 4 и правилу П1,
6.  $(\neg P_2(a) \vee P_3(a, t))$  - заключение по 5 и правилу П2 и.в.,
7.  $(P_2(a) \rightarrow P_3(a, t))$  - заключение по 6 ,
8.  $P_1(a)$  - заключение по 7 и правилу П2 и.в.,
9.  $P_1(a) \rightarrow \forall_y (P_4(y) \rightarrow \neg P_3(a, y))$  - заключение по 2 при подстановке  $x=a$  и правилу П2 и.п,
10.  $\forall_y (P_4(y) \rightarrow \neg P_3(a, y))$  - заключение по 8, 9 и т.р.,
11.  $(\neg P_3(a, t) \vee \neg P_4(t))$  - заключение по 10 и правилу П1 и.п.,
12.  $(P_3(a, t) \rightarrow \neg P_4(t))$  - заключение по 11,
13.  $(P_2(t) \rightarrow \neg P_4(t))$  - заключение по 7, 12 и правилу П9 и.в.
14.  $\forall_y (P_2(y) \rightarrow \neg P_4(y))$  - заключение по 13 и правилу П2 и.п. - *ч.т.д.*

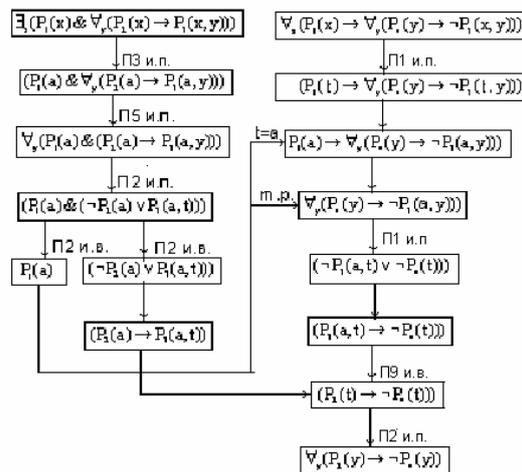


Рис. О.15. Граф дедуктивного вывода.

доказательство по принципу резолюции (см. рис. О.16):

• выписать множество дизъюнктов посылок и отрицания заключения:

$K = \{(\neg P_2(a) \vee P_3(a, t)), (\neg P_3(a, t) \vee \neg P_4(t)), P_1(a), P_2(y), P_4(y)\}$ .

• вывод резольвент:  $P_2(y) \vee (\neg P_2(a) \vee P_3(a, t)) = P_3(a, t)$  при  $y=a$  - резольвента,  $P_3(a, t) \vee (\neg P_3(a, t) \vee \neg P_4(t)) = \neg P_4(t)$  - резольвента,  $\neg P_4(t) \vee P_4(t) = \square$  - пустая резольвента. - *ч.т.д.*

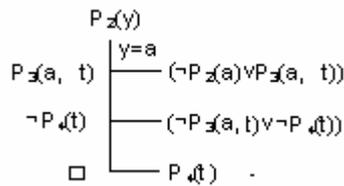


Рис. О.16. Граф вывода по принципу резолюции.

	$r = (r_1 \cup r_2)$				
	1	2	3	4	5
1.3.1a) $r = \text{union}(r_1, r_2) = \{t \mid t \in r_1 \text{ or } t \in r_2\}$ ,	b	c <sub>1</sub>	d	1	4
$r = \{t \mid \forall_x \forall_y (r_1(x) \& r_2(y) \& ((t=x) \vee (t=y)))\}$ ,	1	1			
SELECT $r_1.A_1, r_1.A_2, r_1.A_3,$	b	c <sub>2</sub>	d	2	3
$r_1.A_4, r_1.A_5$	2	2			
FROM $r_1$ UNION	b	c <sub>3</sub>	d	3	2
	3	3			
SELECT $r_2.A_1, r_2.A_2, r_2.A_3,$	b	c <sub>4</sub>	d	4	1
$r_2.A_4, r_2.A_5$	4	4			
FROM $r_2$ .	b	c <sub>3</sub>	d	2	3
	2	4			
	b	c <sub>2</sub>	d	4	1
	1	3			

1.3.1b)  $r = \text{minus}(r_3, r_4) = \{t \mid t \in r_3 \wedge r = (r_3 \setminus r_4) \wedge t \notin r_4\}$ ,  
 $r = \{t \mid \exists x \exists y (r_3(x) \wedge r_4(y) \wedge ((t=x) \wedge \text{not}(t=y)))\}$ ,  
 SELECT  $r_3.A_1, r_3.A_2, r_3.A_3, r_3.A_4,$   
 $r_3.A_5$   
 FROM  $r_3$  MINUS  
 SELECT  $r_4.A_1, r_4.A_2, r_4.A_3, r_4.A_4,$   
 $r_4.A_5$   
 FROM  $r_4$ .

A	A	A	A	A
1	2	3	4	5
b	c	d	3	2
4	3	2		
b	c	d	4	1
1	2	3		

1.3.1c)  $r = \text{intersection}(r_2, r_3) = \{t \mid r = (r_2 \cap r_3) \wedge t \in r_2 \wedge t \in r_3\}$ ,  
 $r = \{t \mid \exists x \exists y (r_2(x) \wedge r_3(y) \wedge (t=x) \wedge (t=y) \wedge (x=y))\}$ ,  
 SELECT  $r_2.A_1, r_2.A_2, r_2.A_2,$   
 $r_2.A_4, r_2.A_5$   
 FROM  $r_2$  INTERSECTION  
 SELECT  $r_3.A_1, r_3.A_2, r_3.A_3,$   
 $r_3.A_4, r_3.A_5$   
 FROM  $r_3$ .

A	A	A	A	A
1	2	3	4	5
b <sub>3</sub>	c <sub>3</sub>	d <sub>3</sub>	3	2
b <sub>1</sub>	c <sub>2</sub>	d <sub>3</sub>	4	1

1.3.1d)  $r = \text{select}(r_3, A_5 > 2) = \{t \mid r = \delta(r_3, A_5 > 2) \wedge t(A_5) > 2\}$ ,  
 $r = \{t \mid \exists x (r_3(x) \wedge (t=x) \wedge (x(A_5) > 2))\}$

A	A	A	A	A
1	2	3	4	5
b	c <sub>3</sub>	d	3	2
4	2			

$\}},$ SELECT A <sub>1</sub> , A <sub>2</sub> , A <sub>3</sub> , A <sub>4</sub> , A <sub>5</sub> FROM r <sub>3</sub> WHERE A <sub>5</sub> >2.	<table style="border-collapse: collapse;"> <tr><td style="padding: 2px 5px;">b</td><td style="padding: 2px 5px;">c<sub>3</sub></td><td style="padding: 2px 5px;">d</td><td style="padding: 2px 5px;">3</td><td style="padding: 2px 5px;">2</td></tr> <tr><td style="padding: 2px 5px;">3</td><td style="padding: 2px 5px;">3</td><td></td><td></td><td></td></tr> <tr><td style="padding: 2px 5px;">b</td><td style="padding: 2px 5px;">c<sub>2</sub></td><td style="padding: 2px 5px;">d</td><td style="padding: 2px 5px;">1</td><td style="padding: 2px 5px;">4</td></tr> <tr><td style="padding: 2px 5px;">1</td><td style="padding: 2px 5px;">3</td><td></td><td></td><td></td></tr> </table>	b	c <sub>3</sub>	d	3	2	3	3				b	c <sub>2</sub>	d	1	4	1	3			
b	c <sub>3</sub>	d	3	2																	
3	3																				
b	c <sub>2</sub>	d	1	4																	
1	3																				

1.3.1e)  $r = \text{product}(r_1, r_2) = \{(t_1 t_2) \mid t_1 \in r_1, t_2 \in r_2\}$ ,  
 $r = \{t = (t_1 t_2) \mid \exists x \exists y (r_2(x) \& r_4(y) \& (t_1 = x) \& (t_2 = y))\} = \exists x \exists y (r_2(x) \& r_4(y) \& (t_1[1] = x[A_1]) \& (t_1[2] = x[A_2]) \& (t_1[3] = x[A_3]) \& (t_1[4] = x[A_4]) \& (t_1[5] = x[A_5]) \& (t_1[6] = y[A_1]) \& (t_1[7] = y[A_2]) \& (t_1[8] = y[A_3]) \& (t_1[9] = y[A_4]) \& (t_1[10] = y[A_5]))\}$ ,  
 SELECT r<sub>2</sub>.A<sub>1</sub>, r<sub>2</sub>.A<sub>2</sub>, r<sub>2</sub>.A<sub>3</sub>, r<sub>2</sub>.A<sub>4</sub>, r<sub>2</sub>.A<sub>5</sub>, r<sub>4</sub>.A<sub>1</sub>, r<sub>4</sub>.A<sub>2</sub>, r<sub>4</sub>.A<sub>3</sub>, r<sub>4</sub>.A<sub>4</sub>, r<sub>4</sub>.A<sub>5</sub>  
 FROM r<sub>1</sub> PRODUCT r<sub>2</sub>.

r=(r <sub>2</sub> ⊗r <sub>4</sub> )	r <sub>2</sub> . A <sub>1</sub>	r <sub>2</sub> . A <sub>2</sub>	r <sub>2</sub> . A <sub>3</sub>	r <sub>2</sub> . A <sub>4</sub>	r <sub>2</sub> . A <sub>5</sub>	r <sub>4</sub> . A <sub>1</sub>	r <sub>4</sub> . A <sub>2</sub>	r <sub>4</sub> . A <sub>3</sub>	r <sub>4</sub> . A <sub>4</sub>	r <sub>4</sub> . A <sub>5</sub>
	b <sub>2</sub>	c <sub>3</sub>	d <sub>4</sub>	2	3	b <sub>3</sub>	c <sub>2</sub>	d <sub>1</sub>	3	2
	b <sub>2</sub>	c <sub>3</sub>	d <sub>4</sub>	2	3	b <sub>3</sub>	c <sub>3</sub>	d <sub>3</sub>	3	2
	b <sub>2</sub>	c <sub>3</sub>	d <sub>4</sub>	2	3	b <sub>1</sub>	c <sub>2</sub>	d <sub>3</sub>	1	4
	b <sub>2</sub>	c <sub>3</sub>	d <sub>4</sub>	2	3	b <sub>2</sub>	c <sub>2</sub>	d <sub>2</sub>	2	3
	b <sub>3</sub>	c <sub>3</sub>	d <sub>3</sub>	3	2	b <sub>3</sub>	c <sub>2</sub>	d <sub>1</sub>	3	2
	b <sub>3</sub>	c <sub>3</sub>	d <sub>3</sub>	3	2	b <sub>3</sub>	c <sub>3</sub>	d <sub>3</sub>	3	2
	b <sub>3</sub>	c <sub>3</sub>	d <sub>3</sub>	3	2	b <sub>1</sub>	c <sub>2</sub>	d <sub>3</sub>	1	4
	b <sub>3</sub>	c <sub>3</sub>	d <sub>3</sub>	3	2	b <sub>2</sub>	c <sub>2</sub>	d <sub>2</sub>	2	3
	b <sub>4</sub>	c <sub>4</sub>	d <sub>4</sub>	4	1	b <sub>3</sub>	c <sub>2</sub>	d <sub>1</sub>	3	2
	b <sub>4</sub>	c <sub>4</sub>	d <sub>4</sub>	4	1	b <sub>3</sub>	c <sub>3</sub>	d <sub>3</sub>	3	2

b <sub>4</sub>	c <sub>4</sub>	d <sub>4</sub>	4	1	b <sub>1</sub>	c <sub>2</sub>	d <sub>3</sub>	1	4
b <sub>4</sub>	c <sub>4</sub>	d <sub>4</sub>	4	1	b <sub>2</sub>	c <sub>2</sub>	d <sub>2</sub>	2	3
b <sub>1</sub>	c <sub>2</sub>	d <sub>3</sub>	4	1	b <sub>3</sub>	c <sub>2</sub>	d <sub>1</sub>	3	2
b <sub>1</sub>	c <sub>2</sub>	d <sub>3</sub>	4	1	b <sub>3</sub>	c <sub>3</sub>	d <sub>3</sub>	3	2
b <sub>1</sub>	c <sub>2</sub>	d <sub>3</sub>	4	1	b <sub>1</sub>	c <sub>2</sub>	d <sub>3</sub>	1	4
b <sub>1</sub>	c <sub>2</sub>	d <sub>3</sub>	4	1	b <sub>2</sub>	c <sub>2</sub>	d <sub>2</sub>	2	3

1.3.1f)  $r = \text{join}(r_1, r_4, r_1.A_i = r_4.A_i) = \{(t_1 t_2) \mid t_1 \in r_1, t_2 \in r_4,$

$r_1.A_i = r_4.A_i, \text{rel}(r) = (\text{rel}(r_1) \cup \text{rel}(r_4))\},$

$r = \{t = (t_1 t_2) \mid \exists x \exists y (r_1(x) \& r_4(y) \& (x[A_4] = y[A_4]) \& (t_1[1] = x[A_1]) \& (t_1[2] = x[A_2]) \&$

$\& (t_1[3] = x[A_3]) \& (t_1[4] = x[A_4]) \& (t_1[5] = x[A_5]) \& (t_2[6] = y[A_1]) \& (t_2[7] = y[A_2]) \& (t_2[8] = y[A_3]) \& (t_2[9] = y[A_5]))\},$

SELECT  $r_1.A_1, r_1.A_2, r_1.A_3, A_4, r_1.A_5, r_4.A_1, r_4.A_2, r_4.A_3,$   
 $r_4.A_5$

FROM  $r_1$  INNER JOIN  $r_4$  ON  $r_1.A_4 = r_4.A_4.$

$r = (r_1 \times r_4)$	$r_1.A_1$	$r_1.A_2$	$r_1.A_3$	$A_4$	$r_1.A_5$	$r_4.A_1$	$r_4.A_2$	$r_4.A_3$	$r_4.A_5$
	b <sub>1</sub>	c <sub>1</sub>	d <sub>1</sub>	1	4	b <sub>1</sub>	c <sub>2</sub>	d <sub>3</sub>	4
	b <sub>2</sub>	c <sub>2</sub>	d <sub>2</sub>	2	3	b <sub>2</sub>	c <sub>2</sub>	d <sub>2</sub>	3
	b <sub>3</sub>	c <sub>3</sub>	d <sub>3</sub>	3	2	b <sub>3</sub>	c <sub>2</sub>	d <sub>1</sub>	2
	b <sub>3</sub>	c <sub>3</sub>	d <sub>3</sub>	3	2	b <sub>3</sub>	c <sub>3</sub>	d <sub>3</sub>	2

1.3.1g)  $r = \text{join}(r_2, r_3, r_2.A_4 > r_3.A_5) = \{(t_1 t_2) \mid t_1 \in r_2, t_2 \in r_3, r_2.A_4 > r_3.A_5,$

$\text{rel}(r) = (\text{rel}(r_2), \text{rel}(r_3))\},$

$r = \{t = (t_1 t_2) \mid \exists x \exists y (r_2(x) \& r_3(y) \& (x[A_4] > y[A_5]) \& (t_1[1] = x[A_1]) \& (t_1[2] = x[A_2]) \&$

$\& (t_1[3] = x[A_3]) \& (t_1[4] = x[A_4]) \& (t_1[5] = x[A_5]) \& (t_2[6] = y[A_1]) \& (t_2[7] = y[A_2]) \&$

$(t_2[8] = y[A_3]) \& (t_2[9] = y[A_4]) \& (t_2[10] = y[A_5]))\},$

$r=(r_2 >^0 < r_3)$	$r_2.A_1$	$r_2.A_2$	$r_2.A_3$	$r_2.A_4$	$r_2.A_5$	$r_3.A_1$	$r_3.A_2$	$r_3.A_3$	$r_3.A_4$	$r_3.A_5$
	$b_2$	$c_3$	$d_4$	2	3	$b_1$	$c_2$	$d_3$	4	1
	$b_3$	$c_3$	$d_3$	3	2	$b_4$	$c_3$	$d_2$	3	2
	$b_3$	$c_3$	$d_3$	3	2	$b_3$	$c_3$	$d_3$	3	2
	$b_3$	$c_3$	$d_3$	3	2	$b_1$	$c_2$	$d_3$	4	1
	$b_4$	$c_4$	$d_4$	4	1	$b_4$	$c_3$	$d_2$	3	2
	$b_4$	$c_4$	$d_4$	4	1	$b_3$	$c_3$	$d_3$	3	2
	$b_4$	$c_4$	$d_4$	4	1	$b_1$	$c_2$	$d_3$	4	1
	$b_1$	$c_2$	$d_3$	4	1	$b_4$	$c_3$	$d_2$	3	2
	$b_1$	$c_2$	$d_3$	4	1	$b_3$	$c_3$	$d_3$	3	2
	$b_1$	$c_2$	$d_3$	4	1	$b_1$	$c_2$	$d_3$	4	1

2.1.1a) Если  $\mu_{C'}(u_i) = \max\{\mu_{A'}(u_i), \mu_{B'}(u_i)\}$ , то  $C' = (A' \cup B') = \{1/u_1, 0,2/u_2, 0,2/u_3, 0,3/u_3, 0,3/u_4, 0,4/u_5, 0,3/u_6, 0,6/u_7, 0,8/u_8\}$ ,

2.1.1b) если  $\mu_{C'}(u_i) = \min\{\mu_{A'}(u_i), \mu_{B'}(u_i)\}$ , то  $C' = (A' \cap B') = \{0,1/u_1, 0,1/u_2\}$ .

2.1.1c) если  $\mu_{\neg A'}(u_i) = (1 - \mu_{A'}(u_i))$ , то  $\neg A' = \{0,9/u_2, 0,8/u_3, 0,7/u_4, 0,6/u_5, 1/u_6, 1/u_7, 1/u_8\}$ ,

если  $\mu_{\neg B'}(u_i) = (1 - \mu_{B'}(u_i))$ , то  $\neg B' = \{0,9/u_1, 0,8/u_2, 1/u_3, 1/u_4, 1/u_5, 0,7/u_6, 0,4/u_7, 0,2/u_8\}$ ,

2.1.1d) если  $\mu_{A' \setminus B'}(u_i) = \min\{\mu_{A'}(u_i), (1 - \mu_{B'}(u_i))\}$ , то  $C' = (A' \setminus B') = (A' \cap \neg B') = \{0,9/u_1, 0,1/u_2, 0,2/u_3, 0,3/u_4, 0,4/u_5\}$ ,

если  $\mu_{B' \setminus A'}(u_i) = \min\{\mu_{B'}(u_i), (1 - \mu_{A'}(u_i))\}$ , то  $C' = (B' \setminus A') = (B' \cap \neg A') = \{0,1/u_1, 0,2/u_2, 0,3/u_6, 0,6/u_7, 0,8/u_8\}$ ,

2.1.1е) если

$$\mu_{A \sqcup B'}(u_i) = \max\{\min\{\mu_{A'}(u_i), (1 - \mu_{B'}(u_i))\}, \min\{\mu_{B'}(u_i), (1 - \mu_{A'}(u_i))\}\},$$

то  $C' = (A' \Delta B') = ((A' \cap \neg B') \cup (B' \cap \neg A')) = \{ 0,9/u_1, 0,2/u_2, 0,2/u_3, 0,3/u_4, 0,4/u_5, 0,3/u_6, 0,6/u_7, 0,8/u_8 \}$ .

2.1.2.a) Для  $h' = (h'_1 \cup h'_2)$

$$\mu_{h'}(x_i, y_j) = (\mu_{h'_1}(x_i, y_j) \vee \mu_{h'_2}(x_i, y_j)) = \max_{i,j} \{\mu_{h'_1}(x_i, y_j), \mu_{h'_2}(x_i, y_j)\}.$$

$h'$	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$	$h'$	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$	$h'_1 \cup h'$	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$
1						2						2					
$x_1$	0,0	0,0	0,0	0,0	0,0	$\cup$	$x_1$	0,0	0,0	0,0	0,0	$x_1$	0,0	0,0	0,0	0,0	0,0
	2	4	6	2	4		4	2	8	2	4		4	4	8	2	4
$x_2$	0,0	0,0	0,0	0,0	0,0		$x_2$	0,0	0,0	0,0	0,0	=	$x_2$	0,0	0,0	0,0	0,0
	3	5	7	5	3		5	7	3	7	5		5	7	7	7	5
$x_3$	0,0	0,0	0,0	0,0	0,0		$x_3$	0,0	0,0	0,0	0,0		$x_3$	0,0	0,0	0,0	0,0
	2	5	4	5	2		5	2	6	2	5		5	5	6	5	5
$x_4$	0,0	0,0	0,0	0,0	0,0		$x_4$	0,0	0,0	0,0	0,0		$x_4$	0,0	0,0	0,0	0,0
	3	6	9	6	3		4	7	8	7	4		4	7	9	7	4

2.1.2b) для  $h' = (h'_1 \cap h'_2)$

$$\mu_{h'}(u_i) = (\mu_{x_1}(u_i) \& \mu_{x_2}(u_i)) = \min_i \{\mu_{x_1}(u_i), \mu_{x_2}(u_i)\}.$$

$h'$	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$	$h'$	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$	$h'_1 \cap h'$	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$
1						2						2					
$x_1$	0,0	0,0	0,0	0,0	0,0	$\cap$	$x_1$	0,0	0,0	0,0	0,0	$x_1$	0,0	0,0	0,0	0,0	0,0
	2	4	6	2	4		4	2	8	2	4		2	2	6	2	4
$x_2$	0,0	0,0	0,0	0,0	0,0		$x_2$	0,0	0,0	0,0	0,0	=	$x_2$	0,0	0,0	0,0	0,0
	3	5	7	5	3		5	7	3	7	5		3	5	3	5	3
$x_3$	0,0	0,0	0,0	0,0	0,0		$x_3$	0,0	0,0	0,0	0,0		$x_3$	0,0	0,0	0,0	0,0
	2	5	4	5	2		5	2	6	2	5		2	2	4	2	2
$x_4$	0,0	0,0	0,0	0,0	0,0		$x_4$	0,0	0,0	0,0	0,0		$x_4$	0,0	0,0	0,0	0,0
	3	6	9	6	3		4	7	8	7	4		3	6	8	6	3

2.1.2c) для  $\neg h'_1$ 

$$\mu_{\neg h'_1}(x_i, y_j) = (1 - \mu_{h'_1}(x_i, y_j)).$$

$h'_1$	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$	$\Rightarrow$	$\neg h'_1$	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$
$x_1$	0,	0,	0,	0,	0,		$x_1$	0,	0,	0,	0,	0,
	2	4	6	2	4			8	6	4	8	6
$x_2$	0,	0,	0,	0,	0,		$x_2$	0,	0,	0,	0,	0,
	3	5	7	5	3			7	5	3	5	7
$x_3$	0,	0,	0,	0,	0,		$x_3$	0,	0,	0,	0,	0,
	2	5	4	5	2			8	5	6	5	8
$x_4$	0,	0,	0,	0,	0,		$x_4$	0,	0,	0,	0,	0,
	3	6	9	6	3			7	4	1	4	7

для  $\neg h'_2$ 

$$\mu_{\neg h'_2}(x_i, y_j) = (1 - \mu_{h'_2}(x_i, y_j)).$$

$h'_2$	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$	$\Rightarrow$	$\neg h'_2$	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$
$x_1$	0,	0,	0,	0,	0,		$x_1$	0,	0,	0,	0,	0,
	4	2	8	2	4			6	8	2	8	6
$x_2$	0,	0,	0,	0,	0,		$x_2$	0,	0,	0,	0,	0,
	5	7	3	7	5			5	3	7	3	5
$x_3$	0,	0,	0,	0,	0,		$x_3$	0,	0,	0,	0,	0,
	5	2	6	2	5			5	8	4	8	5
$x_4$	0,	0,	0,	0,	0,		$x_4$	0,	0,	0,	0,	0,
	4	7	8	7	4			6	3	2	3	6

2.1.2d) для  $h' = h'_1 \setminus h'_2 = h'_1 \cap \neg h'_2$ 

$$\mu_{h'}(x_i, y_j) = (\mu_{h'_1}(x_i, y_j) \& \mu_{\neg h'_2}(x_i, y_j)) = \min_{i,j} \{ \mu_{h'_1}(x_i, y_j), (1 - \mu_{h'_2}(x_i, y_j)) \}.$$

$h'_1$	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$	$\neg h'_2$	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$	$h'$	$y_1$	$y_2$	$y_3$	$y_4$	$y_5$
1						2											

$x_1$	0, 0, 0, 0, 0, 2 4 6 2 4	$\cap$	$x_1$	0, 0, 0, 0, 0, 6 8 2 8 6	$=$	$x_1$	0, 0, 0, 0, 0, 2 4 2 2 4
$x_2$	0, 0, 0, 0, 0, 3 5 7 5 3		$x_2$	0, 0, 0, 0, 0, 5 3 7 3 5		$x_2$	0, 0, 0, 0, 0, 3 3 7 3 3
$x_3$	0, 0, 0, 0, 0, 2 5 4 5 2		$x_3$	0, 0, 0, 0, 0, 5 8 4 8 5		$x_3$	0, 0, 0, 0, 0, 2 5 4 5 2
$x_4$	0, 0, 0, 0, 0, 3 6 9 6 3		$x_4$	0, 0, 0, 0, 0, 6 3 2 3 6		$x_4$	0, 0, 0, 0, 0, 3 3 2 3 3

для  $h' = h'_2 \setminus h'_1 = h'_2 \cap \neg h'_1$

$$\mu_{h'}(x_i, y_j) = (\mu_{h'_2}(x_i, y_j) \& \mu_{\neg h'_1}(x_i, y_j)) = \min_{i,j} \{ \mu_{h'_2}(x_i, y_j), (1 - \mu_{h'_1}(x_i, y_j)) \}.$$

$h'$	$y_1$ $y_2$ $y_3$ $y_4$ $y_5$	$\neg h'_1$	$y_1$ $y_2$ $y_3$ $y_4$ $y_5$	$h'$	$y_1$ $y_2$ $y_3$ $y_4$ $y_5$		
2		1					
$x_1$	0, 0, 0, 0, 0, 4 2 8 2 4	$\cap$	$x_1$	0, 0, 0, 0, 0, 8 6 4 8 6	$=$	$x_1$	0, 0, 0, 0, 0, 4 2 4 2 4
$x_2$	0, 0, 0, 0, 0, 5 7 3 7 5		$x_2$	0, 0, 0, 0, 0, 7 5 3 5 7		$x_2$	0, 0, 0, 0, 0, 5 5 3 5 4
$x_3$	0, 0, 0, 0, 0, 5 2 6 2 5		$x_3$	0, 0, 0, 0, 0, 8 5 6 5 8		$x_3$	0, 0, 0, 0, 0, 5 2 6 2 5
$x_4$	0, 0, 0, 0, 0, 4 7 8 7 4		$x_4$	0, 0, 0, 0, 0, 7 4 1 4 7		$x_4$	0, 0, 0, 0, 0, 4 4 1 4 4

2.1.2e) для  $h' = h^? = (h'_1 \Delta h'_2) = (h'_1 \cap \neg h'_2) \cup (h'_2 \cap \neg h'_1)$

$$\mu_{h'}(x_i, y_j) = (\mu_{h'_1}(x_i, y_j) \& \mu_{\neg h'_2}(x_i, y_j)) \vee (\mu_{h'_2}(x_i, y_j) \& \mu_{\neg h'_1}(x_i, y_j)) = \max \{ \min \{ \mu_{h'_1}(x_i, y_j), (1 - \mu_{h'_2}(x_i, y_j)) \}, \min \{ \mu_{h'_2}(x_i, y_j), (1 - \mu_{h'_1}(x_i, y_j)) \} \}.$$

$h'$	$y_1$ $y_2$ $y_3$ $y_4$ $y_5$	$h'$	$y_1$ $y_2$ $y_3$ $y_4$ $y_5$	$h'_1 \Delta h'_2$	$y_1$ $y_2$ $y_3$ $y_4$ $y_5$	
1		2		2		
$x_1$	0, 0, 0, 0, 0, 2 4 6 2 4	$\Delta$	$x_1$	0, 0, 0, 0, 0, 4 2 8 2 4	$x_1$	0, 0, 0, 0, 0, 4 4 4 2 4

$$\begin{array}{l}
 x_2 \left| \begin{array}{l} 0, 0, 0, 0, 0, \\ 3 \ 5 \ 7 \ 5 \ 3 \end{array} \right. \\
 x_3 \left| \begin{array}{l} 0, 0, 0, 0, 0, \\ 2 \ 5 \ 4 \ 5 \ 2 \end{array} \right. \\
 x_4 \left| \begin{array}{l} 0, 0, 0, 0, 0, \\ 3 \ 6 \ 9 \ 6 \ 3 \end{array} \right.
 \end{array}
 \quad
 \begin{array}{l}
 x_2 \left| \begin{array}{l} 0, 0, 0, 0, 0, \\ 5 \ 7 \ 3 \ 7 \ 5 \end{array} \right. \\
 x_3 \left| \begin{array}{l} 0, 0, 0, 0, 0, \\ 5 \ 2 \ 6 \ 2 \ 5 \end{array} \right. \\
 x_4 \left| \begin{array}{l} 0, 0, 0, 0, 0, \\ 4 \ 7 \ 8 \ 7 \ 4 \end{array} \right.
 \end{array}
 =
 \begin{array}{l}
 x_2 \left| \begin{array}{l} 0, 0, 0, 0, 0, \\ 5 \ 5 \ 7 \ 5 \ 4 \end{array} \right. \\
 x_3 \left| \begin{array}{l} 0, 0, 0, 0, 0, \\ 5 \ 5 \ 6 \ 5 \ 5 \end{array} \right. \\
 x_4 \left| \begin{array}{l} 0, 0, 0, 0, 0, \\ 4 \ 4 \ 2 \ 4 \ 4 \end{array} \right.
 \end{array}$$

2.1.3. Для  $Y'=(X' \circ h)$ 

$$\mu_{Y'}(y_i) = \vee (\mu_{X'}(x_i) \& \mu_h(y_j, x_i)) = \max_j \{ \min \{ \mu_{X'}(x_i), \mu_h(y_j, x_i) \} \}$$

$$\begin{array}{c}
 \begin{array}{c|cccc}
 & x_1 & x_2 & x_3 & x_4 \\
 \hline
 x_1 & 1 & 0,1 & 0,2 & 0,3 \\
 x_2 & 0,1 & 0,2 & 0,3 & 0,3 \\
 x_3 & 0,2 & 0,3 & 0,4 & 0,3 \\
 x_4 & 0,3 & 0,3 & 0,3 & 0,3
 \end{array}
 \circ
 \begin{array}{c|ccccc}
 h' & y_1 & y_2 & y_3 & y_4 & y_5 \\
 \hline
 x_1 & 0,2 & 0,4 & 0,6 & 0,2 & 0,4 \\
 x_2 & 0,3 & 0,5 & 0,7 & 0,5 & 0,3 \\
 x_3 & 0,2 & 0,5 & 0,4 & 0,5 & 0,2 \\
 x_4 & 0,3 & 0,6 & 0,9 & 0,6 & 0,3
 \end{array}
 =
 \begin{array}{c|ccccc}
 & y_1 & y_2 & y_3 & y_4 & y_5 \\
 \hline
 y_1 & 0,3 & & & & \\
 y_2 & 0,4 & & & & \\
 y_3 & 0,6 & & & & \\
 y_4 & 0,3 & & & & \\
 y_5 & 0,4 & & & & 
 \end{array}
 \end{array}$$

2.1.4. Для  $h=(h_1 \circ h_2)$ 

$$\mu_h(x_i, z_k) = \vee (\mu_{h_1}(x_i, y_j) \& \mu_{h_2}(y_j, z_k)) = \max_j \{ \min \{ \mu_{h_1}(x_i, y_j), \mu_{h_2}(y_j, z_k) \} \}$$

$$\begin{array}{c}
 \begin{array}{c|ccc}
 h_1 & y_1 & y_2 & y_3 \\
 \hline
 x_1 & 1, 0, 0, \\ & 0 \ 8 \ 2 \\
 x_2 & 0, 1, 0, \\ & 2 \ 0 \ 4 \\
 x_3 & 0, 1, 0, \\ & 0 \ 0 \ 3 \\
 x_4 & 0, 0, 0, \\ & 2 \ 9 \ 5 \\
 x_5 & 0, 0, 1, \\ & 3 \ 7 \ 0
 \end{array}
 \circ
 \begin{array}{c|cccc}
 h_2 & z_1 & z_2 & z_3 & z_4 \\
 \hline
 y_1 & 0, 0, 0, 0, \\ & 3 \ 3 \ 5 \ 2 \\
 y_2 & 0, 0, 0, 0, \\ & 2 \ 2 \ 4 \ 3 \\
 y_3 & 0, 0, 0, 0, \\ & 1 \ 3 \ 2 \ 6
 \end{array}
 =
 \begin{array}{c|cccc}
 h_1 \circ h_2 & z_1 & z_2 & z_3 & z_4 \\
 \hline
 x_1 & 0, 0, 0, 0, \\ & 3 \ 3 \ 5 \ 3 \\
 x_2 & 0, 0, 0, 0, \\ & 2 \ 3 \ 4 \ 4 \\
 x_3 & 0, 0, 0, 0, \\ & 2 \ 3 \ 4 \ 3 \\
 x_4 & 0, 0, 0, 0, \\ & 2 \ 3 \ 4 \ 5 \\
 x_5 & 0, 0, 0, 0, \\ & 3 \ 3 \ 4 \ 6
 \end{array}
 \end{array}$$

2.1.5a) Для  $r'=(r'_1 \cup r'_2)$

$$\mu_{r'}(x_i, x_j) = (\mu_{r'_1}(x_i, x_j) \vee \mu_{r'_2}(x_i, x_j)) = \max_{i,j} \{\mu_{r'_1}(x_i, x_j), \mu_{r'_2}(x_i, x_j)\}.$$

$r'_1$	$x_1$ $x_2$ $x_3$ $x_4$ $x_5$	$r'_2$	$x_1$ $x_2$ $x_3$ $x_4$ $x_5$	$r'_1 \cup r'_2$	$x_1$ $x_2$ $x_3$ $x_4$ $x_5$		
1	1 0, 0, 0, 0, 0, 3 2 1 4	2	1 0, 0, 0, 0, 0, 2 8 2 4	=	$r'_2$	$x_1$	1 0, 0, 0, 0, 0, 3 8 2 4
$x_2$	0, 1 0, 0, 0, 0, 3 7 5 3	$x_2$	0, 1 0, 0, 0, 0, 5 8 7 2			$x_2$	0, 1 0, 0, 0, 0, 5 8 7 3
$x_3$	0, 0, 1 0, 0, 0, 2 5 5 2	$x_3$	0, 0, 1 0, 0, 0, 2 2 2 5			$x_3$	0, 0, 1 0, 0, 0, 2 5 5 5
$x_4$	0, 0, 0, 1 0, 0, 7 5 9 3	$x_4$	0, 0, 0, 1 0, 0, 4 7 8 4			$x_4$	0, 0, 0, 1 0, 0, 7 7 9 4
$x_5$	0, 0, 0, 0, 1 6 7 2 3	$x_5$	0, 0, 0, 0, 1 1 1 5 4			$x_5$	0, 0, 0, 0, 1 6 7 5 4

2.1.5b) Для  $r'=(r'_1 \cap r'_2)$

$$\mu_{r'}(x_i, x_j) = (\mu_{r'_1}(x_i, x_j) \& \mu_{r'_2}(x_i, x_j)) = \min_{i,j} \{\mu_{r'_1}(x_i, x_j), \mu_{r'_2}(x_i, x_j)\}.$$

$r'_1$	$x_1$ $x_2$ $x_3$ $x_4$ $x_5$	$r'_2$	$x_1$ $x_2$ $x_3$ $x_4$ $x_5$	$r'_1 \cap r'_2$	$x_1$ $x_2$ $x_3$ $x_4$ $x_5$		
1	1 0, 0, 0, 0, 0, 3 2 1 4	2	1 0, 0, 0, 0, 0, 2 8 2 4	=	$r'_1 \cap r'_2$	$x_1$	1 0, 0, 0, 0, 0, 2 2 1 4
$x_2$	0, 1 0, 0, 0, 0, 3 7 5 3	$x_2$	0, 1 0, 0, 0, 0, 5 8 7 2			$x_2$	0, 1 0, 0, 0, 0, 3 7 5 2
$x_3$	0, 0, 1 0, 0, 0, 2 5 5 2	$x_3$	0, 0, 1 0, 0, 0, 2 2 2 5			$x_3$	0, 0, 1 0, 0, 0, 2 2 2 2
$x_4$	0, 0, 0, 1 0, 0, 7 5 9 3	$x_4$	0, 0, 0, 1 0, 0, 4 7 8 4			$x_4$	0, 0, 0, 1 0, 0, 4 5 8 3



x <sub>1</sub>	1 0, 0, 0, 0, 0,	x <sub>1</sub>	0 0, 0, 0, 0, 0,	x <sub>1</sub>	0 0, 0, 0, 0, 0,
	3 2 1 4 ∩		8 2 8 6 =		3 2 1 4
x <sub>2</sub>	0, 1 0, 0, 0, 0,	x <sub>2</sub>	0, 0 0, 0, 0, 0,	x <sub>2</sub>	0, 0 0, 0, 0, 0,
	3 7 5 3		5 2 3 8		3 2 3 3
x <sub>3</sub>	0, 0, 1 0, 0, 0,	x <sub>3</sub>	0, 0, 0 0, 0, 0,	x <sub>3</sub>	0, 0, 0 0, 0, 0,
	2 5 5 2		8 8 8 5		2 5 5 2
x <sub>4</sub>	0, 0, 0, 1 0,	x <sub>4</sub>	0, 0, 0, 0 0,	x <sub>4</sub>	0, 0, 0, 0 0,
	7 5 9 3		6 3 2 6		6 3 2 3
x <sub>5</sub>	0, 0, 0, 0, 1	x <sub>5</sub>	0, 0, 0, 0, 0	x <sub>5</sub>	0, 0, 0, 0, 0
	6 7 2 3		9 9 5 6		6 7 2 3

Для  $r' = r'_2 \setminus r'_1 = r'_2 \cap \neg r'_1$

$$\mu_{r'}(x_i, x_j) = (\mu_{r'_2}(x_i, x_j) \& \mu_{\neg r'_1}(x_i, x_j)) = \min_{i,j} \{ \mu_{r'_2}(x_i, x_j), (1 - \mu_{r'_1}(x_i, x_j)) \}.$$

r' <sub>2</sub>	x <sub>1</sub> x <sub>2</sub> x <sub>3</sub> x <sub>4</sub> x <sub>5</sub>	¬r' <sub>1</sub>	x <sub>1</sub> x <sub>2</sub> x <sub>3</sub> x <sub>4</sub> x <sub>5</sub>	r' <sub>2</sub> \setminus r' <sub>1</sub>	x <sub>1</sub> x <sub>2</sub> x <sub>3</sub> x <sub>4</sub> x <sub>5</sub>
x <sub>1</sub>	1 0, 0, 0, 0, 0,	x <sub>1</sub>	0 0, 0, 0, 0, 0,	x <sub>1</sub>	0 0, 0, 0, 0, 0,
	2 8 2 4 ∩		7 8 9 6 =		2 8 2 4
x <sub>2</sub>	0, 1 0, 0, 0, 0,	x <sub>2</sub>	0, 0 0, 0, 0, 0,	x <sub>2</sub>	0, 0 0, 0, 0, 0,
	5 8 7 2		7 3 5 7		5 3 5 2
x <sub>3</sub>	0, 0, 1 0, 0, 0,	x <sub>3</sub>	0, 0, 0 0, 0, 0,	x <sub>3</sub>	0, 0, 0 0, 0, 0,
	2 2 2 5		8 5 5 8		2 2 2 5
x <sub>4</sub>	0, 0, 0, 1 0,	x <sub>4</sub>	0, 0, 0, 0 0,	x <sub>4</sub>	0, 0, 0, 0 0,
	4 7 8 4		3 5 1 7		3 5 1 4
x <sub>5</sub>	0, 0, 0, 0, 1	x <sub>5</sub>	0, 0, 0, 0, 0	x <sub>5</sub>	0, 0, 0, 0, 0
	1 1 5 4		4 3 8 7		1 1 5 4

2.1.5e) Для  $r' = (r'_1 \Delta r'_2) = (r'_1 \cap \neg r'_2) \cup (r'_2 \cap \neg r'_1)$

$$\mu_{r'}(x_i, x_j) = (\mu_{r'_1}(x_i, x_j) \& \mu_{\neg r'_2}(x_i, x_j)) \vee (\mu_{r'_2}(x_i, x_j) \& \mu_{\neg r'_1}(x_i, x_j)) = \max \{ \min_{i,j} \{ \mu_{r'_1}(x_i, x_j), (1 - \mu_{r'_2}(x_i, x_j)) \}, \min_{i,j} \{ \mu_{r'_2}(x_i, x_j), (1 - \mu_{r'_1}(x_i, x_j)) \} \}.$$

r' <sub>1</sub> Δ r' <sub>2</sub>	x <sub>1</sub> x <sub>2</sub> x <sub>3</sub> x <sub>4</sub> x <sub>5</sub>	r' <sub>1</sub> Δ r' <sub>2</sub>	x <sub>1</sub> x <sub>2</sub> x <sub>3</sub> x <sub>4</sub> x <sub>5</sub>
1		2	

x <sub>1</sub>	1 0, 0, 0, 0, 0,	x <sub>1</sub>	1 0, 0, 0, 0, 0,	x <sub>1</sub>	1 0, 0, 0, 0, 0,
	3 2 1 4 Δ		2 8 2 4 =		3 8 2 4
x <sub>2</sub>	0, 1 0, 0, 0, 0,	x <sub>2</sub>	0, 1 0, 0, 0, 0,	x <sub>2</sub>	0, 1 0, 0, 0, 0,
	3 7 5 3		5 8 7 2		5 8 7 3
x <sub>3</sub>	0, 0, 1 0, 0, 0,	x <sub>3</sub>	0, 0, 1 0, 0, 0,	x <sub>3</sub>	0, 0, 1 0, 0, 0,
	2 5 5 2		2 2 2 5		2 5 5 5
x <sub>4</sub>	0, 0, 0, 1 0,	x <sub>4</sub>	0, 0, 0, 1 0,	x <sub>4</sub>	0, 0, 0, 1 0,
	7 5 9 3		4 7 8 4		7 7 9 4
x <sub>5</sub>	0, 0, 0, 0, 1	x <sub>5</sub>	0, 0, 0, 0, 1	x <sub>5</sub>	0, 0, 0, 0, 1
	6 7 2 3		1 1 5 4		6 7 5 4

2.1.5f) Для  $r=(r_1 \circ r_2)$

$$\mu_r(x_i, x_k) = \vee(\mu_{r_1}(x_i, x_j) \& \mu_{r_2}(x_j, x_k)) = \max\{\min_j\{\mu_{r_1}(x_i, x_j), \mu_{r_2}(x_j, x_k)\}\}.$$

r'	x <sub>1</sub>	x <sub>2</sub>	x <sub>3</sub>	x <sub>4</sub>	x <sub>5</sub>	r' <sub>2</sub>	x <sub>1</sub>	x <sub>2</sub>	x <sub>3</sub>	x <sub>4</sub>	x <sub>5</sub>	r' <sub>1</sub> ∘ r' <sub>2</sub>	x <sub>1</sub>	x <sub>2</sub>	x <sub>3</sub>	x <sub>4</sub>	x <sub>5</sub>	
1																		
x <sub>1</sub>	1 0, 0, 0, 0, 0,					x <sub>1</sub>	1 0, 0, 0, 0, 0,					x <sub>1</sub>	1 0, 0, 0, 0, 0,					
	3 2 1 4 ∘						2 8 2 4 =						3 8 4 4					
x <sub>2</sub>	0, 1 0, 0, 0, 0,					x <sub>2</sub>	0, 1 0, 0, 0, 0,					x <sub>2</sub>	0, 1 0, 0, 0, 0,					
	3 7 5 3						5 8 7 2						5 8 7 5					
x <sub>3</sub>	0, 0, 1 0, 0, 0,					x <sub>3</sub>	0, 0, 1 0, 0, 0,					x <sub>3</sub>	0, 0, 1 0, 0, 0,					
	2 5 5 2						2 2 2 5						5 5 5 5					
x <sub>4</sub>	0, 0, 0, 1 0,					x <sub>4</sub>	0, 0, 0, 1 0,					x <sub>4</sub>	0, 0, 0, 1 0,					
	7 5 9 3						4 7 8 4						7 7 9 5					
x <sub>5</sub>	0, 0, 0, 0, 1					x <sub>5</sub>	0, 0, 0, 0, 1					x <sub>5</sub>	0, 0, 0, 0, 1					
	6 7 2 3						1 1 5 4						6 7 7 7					

## Литература

- [1] Адаменко А. Н., Кучуков А. М. Логическое программирование и Visual Prolog. – СПб.: БХВ-Петербург, 2003г.,
- [2] Борисов А. Н., Алексеев А. В. и др. Обработка нечёткой информации в системах принятия решений. – М.: Радио и связь, 1989г.,
- [3] Вагин В. Н. Дедукция и обобщение в системах принятия решений. – М.: Наука, 1988г.,
- [4] Временные логики и их применение для анализа программ. – <http://mathcyb.cs.msu.su/courses/tenselog.html>,
- [5] Грей П. Логика, алгебра и базы данных: Пер. с англ. /Килова Х. И./ - М.: Изд. Машиностроение, 1989г.,
- [6] Гуц А. К. Математическая логика и теория алгоритмов: Учебное пособие. - Омск: Изд-во Наследие. Диалог-Сибирь, 2003г.- <http://univ2.omsk.su/~guts/logic.htm>,
- [7] Дубинин В. Н., Зинин С. А. Проектирование вычислительных систем и сетей на основе сетевых формализмов. – <http://alice.stup.ac.ru/~dvn/compex/index.htm>,
- [8] Информационные технологии::Открытые системы::Советы и мнения::- <http://www.osp.ru/os/2003/09/067.htm>,
- [9] Кириллов В. И., Старченко А. А. Логика. – М.: Высшая школа, 1987г.,
- [10] Колмогоров А.Н. Драгилин. Введение в математическую логику, 1982г.,
- [11] Колмогоров А.Н., Драгилин. Математическая логика. Дополнительные главы. – 1984г.
- [12] Кузнецов О.П. , Адельсон-Вельский Г.М. Дискретная Математика для инженера.- М.: Энергоатомиздат, 1988г.,

- [13] Лихтарников Л.М., Сукачева В.М. Математическая логика. Курс лекций. – СПб: Лань 1998г.,
- [14] Мейер Д. Теория реляционных баз данных: Пер. с англ. /М.К. Валиева/. – М.: Мир, 1987г.,
- [15] Непейвода Н.Н. Прикладная логика: Учеб. Пособие. – 2-е изд. Испр. и доп. – Новосибирск: изд. Новосиб. Ун-та, 2000г.,
- [16] Обработка нечёткой информации в системах принятия решений А. Н. Борисов, А. В. Алексеев, Г. В. Меркурьев и др. – М.: Радио и связь, 1986г.
- [17] Першиков В. И., Савинков В. М. Толковый словарь по информатике. – М.: Финансы и статистика, 1991г.,
- [18] Попов Э.В. Экспертные системы. Решение неформализованных задач в диалоге с ЭВМ. – М.: Наука, 1987г.,
- [19] Пономарев В.Ф. Математическая логика. Часть 1. Логика высказываний и логика предикатов. Учебное пособие. – Калининград: изд. КГТУ, 2001г.,
- [20] Пономарев В.Ф. Математическая логика. Часть 2. Логика реляционная. Логика нечёткая. Учебное пособие. – Калининград: изд. КГТУ, 2001г.
- [21] Столл Р. Р. Множества. Логика. Аксиоматические теории. - М.: Просвещение, 1968г.,
- [22] Фейс Р. Модальная логика. (серия математическая логика и основания математики). Перевод с англ. /Г. Е. Минца/ - М.: Наука, 1974г.

Предметный указатель

- 
- аксиома, 30
    - алгебры высказываний, 21
    - исчисления предикатов, 69
  - алгебра
    - высказываний, 11
    - нечёткая, 4, 143
    - предикатов, 55
    - реляционная, 113
  - алгоритм
    - метода резолюции, 41
    - приведения к нормальной форме, 27
  - приведения к ПНФ, 62
    - приведения к ССФ, 173
  - атрибут, 111, 162
  - верификация программы, 171
  - включение нечёткого множества, 143
  - вложенные подзапросы, 126
  - высказывание, 9, 28
  - дизъюнкция, 10, 11, 19, домен, 111
  - дополнение, 143
    - нечёткого множества, 146
    - нечёткого отношения, 147
  - закключение, 11, 18, 34
  - закон
    - алгебры высказываний, 21
    - алгебры предикатов, 58
    - непротиворечия, 20
    - нечёткой алгебры, 4, 153
    - тождества, 20
  - запись, 111
  - импликация, 10, 58, 79, 90, 169
  - интенциональная часть дедуктивной системы, 79
  - интерпретатор, 81
  - исчисление
    - высказываний, 11, 29
    - предикатов, 3, 67
  - квантор
    - всеобщности, 52
    - существования, 51
  - КНФ, 26, 41, 62, 180
  - континуальное множество, 136
  - конъюнкция, 10, 28, 143, 144

- кортеж, 8, 119, 122, 128, 130
- коэффициент  
уверенности, 162
- логика  
алгоритмическая, 136,  
временная, 2, 170  
высказываний, 3, 7, 9,  
классическая логика, 2,  
7  
математическая, 6  
модальная, 5, 8, 136,  
168,  
неклассическая, 7, 136  
предикатов, 3, 7, 50  
реляционная, 111  
темпоральная, 136
- логические операции, 11,  
70,
- матрица, 62, 63
- мера доверия, 162  
недоверия, 162
- метод  
дедуктивного вывода, 3,  
34,  
резолуции, 3, 41
- modus ponens*, 35, 68, 158,  
169
- modus tollens*, 35, 68
- модель реляционная, 111
- носитель множества, 138
- операции  
бинарные, 4, 117  
логические, 12, 55  
унарные, 4, 114
- отношение  
достижимости, 169, 170,  
171  
нечёткого нестрогого  
порядка, 155  
нечёткого строгого  
порядка, 155  
нечёткой  
эквивалентности, 155
- переменная  
-кортеж, 122  
лингвистическая, 156  
предметные, 50, 81, 101  
пропозициональные, 9  
связанная, 54  
свободная, 54
- постусловие, 170
- посылка, 11, 18
- правила  
подстановки, 21, 70,  
188, предикат, 50, 52,  
53, 80, 93
- предметные постоянные,  
50, 79, 80, 81, 156
- предусловие, 170

- преобразования
  - эквивалентные, 23, 61
  - префикс, 62, 187, 189
  - принцип
  - резолюции, 41
  - равенство нечётких
    - множеств, 144
  - равносильные формулы, 21, 60, 153
  - разность нечётких
    - множеств, 147
  - резольвента, 41, 45
  - силлогизм, 6
  - симметрическая разность
    - нечётких множеств, 148
  - сколемовская стандартная форма (ССФ), 3, 65
  - сколемовские функции, 65, 70, 75
  - совершенные
    - нормальные формы
      - дизъюнктивные, 28
      - конъюнктивные, 28
  - степень
    - принадлежности, 137
    - антирефлексивности, 154
    - антисимметричности, 154
    - рефлексивности, 154
    - симметричности, 154
    - транзитивности, 154
- суждение
  - общее, 52
  - частное, 51
- схема
  - дедуктивного вывода, 34
- отношения, 112
- реляционной базы, 112
- таблица истинности, 15
- терм, 55, 59, 122
- терм-множеством, 156
- тестирование, 170
- унификация предикатов, 69
- формула, 11, 26, 143,
  - выводимая, 68
  - тождественно истинная, 30
  - тождественно ложная, 31
  - элементарная, 55, 122
- функция
  - принадлежности, 138, 140,
  - высказывательная, 50
- эквивалентность, 10, 169
- эквисоединение, 120
- экспертная система, 158

экстенциональная часть  
    дедуктивной системы,  
    79,  
элементарная  
    дизъюнкция, 27  
элементарная  
    конъюнкция, 27  
язык SQL, 124

