МОСКОВСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ
имени М. В. ЛОМОНОСОВА
Механико–математический факультет

ИНТЕЛЛЕКТУАЛЬНЫЕ СИСТЕМЫ

Сборник задач по курсу «Дискретный анализ и интеллектуальные системы»

Д. В. Алексеев

Москва — 2021

Оглавление

На протяжении ряда лет студентам, специализирующимся по кафедре математической теории интеллектуальных систем (МаТИС) механико-математического факультета МГУ имени М.В.Ломоносова читается курс лекций "Теория интеллектуальных систем". При этом курсе работает спецсеминар, рассчитанный на студентов третьего курса кафедры МаТИС. Материал, накопленный за годы работы этого семинара, нашел свое отражение в этой книге.

Предисловие

На протяжении ряда лет студентам, специализирующимся по кафедре математической теории интеллектуальных систем (МаТИС) механико-математического факультета МГУ имени М.В.Ломоносова читается курс лекций "Теория интеллектуальных систем". При этом курсе работает спецсеминар, рассчитанный на студентов третьего курса кафедры МаТИС. Материал, накопленный за годы работы этого семинара, нашел свое отражение в этой книге.

Очертим в достаточно общем виде распространенный вариант интеллектуальной системы, изображенный на рисунке 1.

Имеется объект **О**, помещенный в среду **С**, с которой у него имеется двусторонняя связь. Он может воспринимать информацию, поступающую из среды, и влиять на нее, что изображено соответствующим направлением стрелок.

Входная информация из **С** поступает в **О** на блок распознавания **Р**, оттуда она направляется в блок оперативной памяти **П**, где подвергается анализу. При этом анализе используется блок **ДЗ** базы данных и знаний, играющий роль долгосрочной памяти, а сам процесс анализа регулируется управляющим блоком **У**, который учитывает группу параметров, описывающих как внутренние характеристики объекта, так и состояние среды.

Базы данных и знаний вместе с блоком управления образуют "мозговой центр"системы. От достаточности заложенной в них информации и эффективности внутренних операторов зависят ее имитационные возможности.

Функционирование объекта в среде осуществляется во времени пошаго-

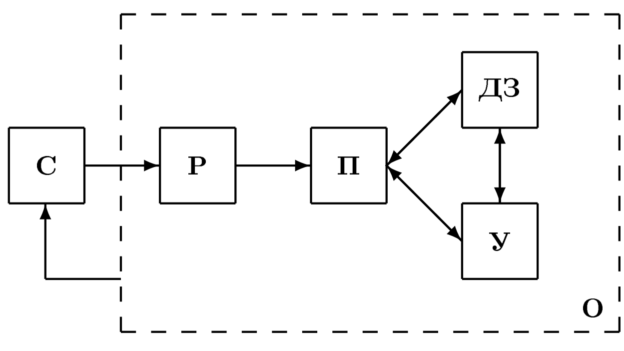


Рис. 1: Интеллектуальная система.

во и оценивается серией внутренних и в общем случае внешних функционалов.

Последовательность значений этих функционалов считается характеризацией взаимодействия объекта и среды. По ней осуществляется оценка "разумности"поведения объекта, включая, в частности, заключение о том, сумел ли объект решить заданную задачу.

В соответствии со схемой интеллектуальной системы, изображенной на рисунке 1, и следуя курсу "Теория интеллектуальных систем"в книге будут рассмотрены 3 основные составляющие интеллектуальной системы. Первая глава, посвященная распознаванию образов, связана с блоком распознавания **Р**. Вторая глава посвящена теории хранения и поиска информации и связана в блоком **ДЗ** — баз данных и знаний. И наконец, последняя глава курса, связанная с блоком управления **У**, посвящена математической логике.

Нам приятно поблагодарить весь коллектив кафедры МаТИС, чье постоянное внимание, доброжелательная критика и советы способствовали появлению этой книги. Авторы выражают особую благодарность В.Б.Кудрявцеву, А.С.Подколзину, В.А.Носову, А.С.Строгалову и А.А.Часовских, которые принимали активное участие в разработке курса "Теория интеллектуальных систем и П.А.Алисейчику, И.Л.Мазуренко, П.А.Пантелееву и А.Б.Холоденко за помощь в составлении упражнений.

Глава 1

Распознавание образов

1.1. Геометрический подход к распознаванию

Преобразования плоскости.

Рассмотрим следующие группы преобразований плоскости

Будем обозначать Γ_0 — множество состоящее из одного тождественного преобразования.

Будем обозначать Γ_m — множество движений, т.е. множество преобразований плоскости, сохраняющих расстояние между точками.

Будем обозначать Γ_s — множество преобразований подобия, т.е. множество преобразований плоскости, при которых расстояния между точками умножаются на фиксированный коэффициент k.

Аффинным преобразованием будем называть отображение, которое отображает точку (x, y) в (x', y'), где $(x', y')^T = A \cdot (x, y)^T + b$, где A — матрица $det(A) \neq 0$. Геометрически аффинные преобразования представляют собой композиции параллельного переноса, поворота, симметрии и растяжения по направлениям. при аффинном преобразовании сохраняется форма предметов, хотя размеры предметов и пропорции могут изменяться. Аффинные преобразования используются при распознавании образов. Множество всех аффинных преобразований будем обозначать Γ_a.

Замечание. Выполняется включение $\Gamma_0 \subseteq \Gamma_m \subseteq \Gamma_s \subseteq \Gamma_a$.

Множества A и A' будем называть *эквивалентными* относительно Γ_0 и обозначать $A \overset{0}{\simeq} A'$, если существует преобразование $\mathcal{T} \in \Gamma_0$, такое, что $\mathcal{T}(A) = A'$.

Аналогично определяется эквивалентность относительно Γ_m, Γ_s и Γ_a. Обозначаются они $A \overset{m}{\simeq} A'$, $A \overset{s}{\simeq} A'$ и $A \overset{a}{\simeq} A'$, соответственно.

Мы рассмотрим далее разные варианты преобразований изображения на сетчатке и для каждого такого случая будем строить кодировку изображения, инвариантную к рассматриваемым преобразованиям. Распознавание будет основываться на этих инвариантах.

Кодирование изображений.

Пусть A — конечное множество точек плоскости. В дальнейшем будем называть его *изображением.* Биекцию $M : A \leftrightarrow \{1, ..., n\}$ (где $n = |A|$) будем называть функцией нумерации изображения A. Несложно заметить, что произвольному множеству можно поставить в соответствие $n!$ различных функций нумерации. Будем обозначать $a_i = M_A^{-1}(i)$ — точку, которая соответствует номеру i; $X(a_i)$ и $Y(a_i)$ — координаты этой точки. Хотелось бы отметить, что эти обозначения имеет смысл только если задана функция нумерации.

Пусть $T : \mathbb{R}^{2n} \to \mathbb{R}^t$ — кодирующая функция. Будем называть кодом изображения A пару $< M_A, T_A >$, где M_A — функция нумерации, а $T_A = T(X(a_1), Y(a_1), \ldots, X(a_n), Y(a_n))$ — значение кодирующей функции на координатах точек изображения. Величину t будем называть сложностью кода.

Будем называть изображения A и B эквивалентными относительно кодирующей функции T, если $|A| = |B|$ и существуют функции нумерации M_A и M_B, такие, что $T_A = T_B$. Эквивалентность обозначается как $A \overset{T}{\sim} B$.

Пусть A — множество из n различных точек плоскости, т.е. $|A| = n$. Это множество в дальнейшем будем называть *изображением*.

Взаимно однозначную функцию $M_A : A \to \{1, 2, \ldots, n\}$ будем называть *функцией нумерации* изображения A. Поскольку M_A взаимно однозначная функция, то определена функция M_A^{-1}, которая номеру из $\{1, 2, \ldots, n\}$ сопоставляет точку из A.

Если a точка плоскости, то через $X(a)$, $Y(a)$ обозначим соответственно абсциссу и ординату точки a.

Вектор

$$K_{M_A} = (X(M_A^{-1}(1)), Y(M_A^{-1}(1)), \ldots, X(M_A^{-1}(n)), Y(M_A^{-1}(n)))$$

будем называть *вектором координат изображения A при нумерации M_A*.

Пусть $T : \mathbb{R}^{2n} \to \mathbb{R}^t$ — некоторая t-мерная вектор функция, называемая *кодирующей*. Пару $\langle M_A, T_A \rangle$, где $T_A = T(K_{M_A})$, будем называть *кодом изображения A при нумерации M_A для кодирующей функции T*. Число t, равное длине вектора T_A и размерности кодирующей функции T, назовем *сложностью кода $\langle M_A, T_A \rangle$*.

Два изображения A и B назовем *эквивалентными относительно кодирующей функции T*, если $|A| = |B|$ и существуют такие функции нумерации M_A и M_B, что $T(K_{M_A}) = T(K_{M_B})$.

Пусть Γ — некоторое множество геометрических преобразований плоскости. В дальнейшем мы будем рассматривать следующие множества:

- Γ_1 — множество, состоящее из одного тождественного преобразования;
- Γ_2 — множество преобразований, получающихся с помощью любых комбинаций сдвига, поворота и преобразований симметрии;
- Γ_3 — множество преобразований подобия, т.е. преобразований, получающихся с помощью любых комбинаций сдвига, поворота и преобразований симметрии и изменения в размерах (с сохранением подобия);
- $\Gamma_4 = \Gamma_a$ — множество невырожденных аффинных преобразований плоскости.

Скажем, что два изображения *эквивалентны относительно множества преобразований* Γ, если одно может быть получено из другого с помощью преобразований из Γ.

Скажем, что кодирующая функция T *правильная для множества геометрических преобразований* Γ, если два изображения эквивалентны относительно кодирующей функции T тогда и только тогда, когда они эквивалентны относительно множества преобразований Γ.

Таким образом, с помощью кода с правильной для множества геометрических преобразований Γ кодирующей функцией может решаться задача распознавания эквивалентности изображений относительно этого множества преобразований.

Примеры кодирования.

Случай 1. Пусть дано изображение A. Перенумеруем его точки некоторым образом так, чтобы номера были попарно различны, т.е. зададим функцию нумерации M_A. Кодом K^1 изображения A (обозначение: K^1_A) назовем пару множеств $< M_A, T_A >$. Здесь T_A множество координат $(x,y)_n$ точек с указанием их номера, то есть, например, $(5,3)_n$ означает, что у точки с номером n координатами являются пара $(5,3)$. То есть для кода K^1_A в качестве кодирующей функции выступает функция $T : \mathbb{R}^{2n} \to \mathbb{R}^{2n}$ такая, что $T_A = T(K_{M_A}) = K_{M_A}$.

Перенумеровав по иному множество точек изображения A, получим другую пару $< M_A, T_A >$. Будем, однако, рассматривать кодировку с точностью до перенумерации и называть изображения, отличающиеся только нумерацией точек, K^1-эквивалентными . Таким образом, если имеются изображение A с кодом $< M_A, T_A >$ и B с кодом $< M_B, T_B >$, то назовем A и B K^1-эквивалентными , если существует такая нумерация точек изображения A, при которой его код есть $< \tilde{M}_A, \tilde{T}_A >$, и такая нумерация для B, при которой его код есть $< \tilde{M}_B, \tilde{T}_B >$, и при этом $\tilde{T}_A = \tilde{T}_B$.

Легко видеть, что код K^1_A является правильным для множества преобразований Γ_1.

Случай 2. Пусть рассматриваются движения плоскости (преобразования из Γ_2). В этом случае меняются координаты точек, но сохраняются расстояния между ними. Рассмотрим код $K^2_A =< M_A, T_A >$, где M_A есть функция нумерации множества A. Множество T_A составляют все числа $r(q,m)$ (q и m — номера точек из $\{1, 2, \dots, |A|\}$), являющиеся расстояниями между точками изображения с номерами q и m. То есть, если $|A| = n$, то кодирующая функция T такая, что $T : \mathbb{R}^{2n} \to \mathbb{R}^t$, где $t = n(n-1)/2$ — число различных пар (q,m), $q < m$, и если нумеровать компоненты вектор-функции T парами (q,m), то (q,m)-я компонента равна

$$T_{q,m}(K_{M_A}) = \sqrt{(X(M_A^{-1}(q) - X(M_A^{-1}(m))^2 + (Y(M_A^{-1}(q) - Y(M_A^{-1}(m))^2}.$$

Аналогично вводится понятие K^2-эквивалентности — изображения имеющие одинаковые с точностью до перенумерации точек коды. Можно показать, что два изображения K^2-эквивалентны тогда и только тогда, когда одно может быть получено из другого комбинацией сдвига, поворота и преобразования симметрии, т.е. код K^2_A — правильный для множества преобразований Γ_2[1].

Отметим, что для изображения A, состоящего из n точек, сложность K^1_A равна $2n$, а сложность K^2_A равна $n(n-1)/2$.

[1]Это можно рассматривать как частный случай утверждения, доказанного в [7]

Вместе с тем код K_A^2 явно избыточен. Это видно, например, из того, что если зафиксировать на плоскости положение трех точек изображения, то для определения положения остальных точек все элементы множества T_A, очевидно, не потребуются.

Случай 3. Пусть рассматриваются преобразования подобия (Γ_3). При таких преобразованиях расстояния не сохраняются, но сохраняются их отношения, что лежит в основе кода K_A^3. Так же, как и в предыдущих случаях, перенумеруем множество точек изображения A и обозначим функцию нумерации через M_A. Далее зададим множество T_A численных значений отношений вида $\rho_{ml,pq} = \frac{r(m,l)}{r(p,q)}$, где $r(m,l)$ и $r(p,q)$ — расстояния между точками с номерами соответственно m и l, p и q, т.е.

$$r(p,q) = \sqrt{(X(M_A^{-1}(p) - X(M_A^{-1}(q))^2 + (Y(M_A^{-1}(p) - Y(M_A^{-1}(q))^2}.$$

Здесь m, l, p, q — номера из $\{1, 2, \ldots, |A|\}$, $m < l$, $p < q$. Для каждого числа из T_A полагаем известной соответствующую ему четверку номеров m, n, p, q. Код K_A^3 есть пара $< M_A, T_A >$. Аналогично тому, как это рассматривалось для предыдущих двух случаев, можно определить одинаковые с точностью до перенумерации точек коды. Изображения с такими кодами будем называть K^3-эквивалентными. В [10] показано, что два изображения K^3-эквивалентны в том и только в том случае, если на плоскости одно может быть получено из другого сдвигом, поворотом, изменением в размерах (с сохранением подобия), преобразованием симметрии или их комбинацией (то есть подобными преобразованиями), т.е. код K_A^3 — правильный для множества преобразований Γ_3.

Процедуру распознавания изображений, полученных преобразованиями подобия, можно провести аналогично процедуре, описанной для случая 2.

Множество T_A имеет $(n(n-1)/2)(n(n-1)/2-1)$ элементов. Вместе с тем код K_A^3, очевидно, избыточен.

Случай 4 (основной). Будем рассматривать аффинные преобразования (невырожденные) изображения (Γ_4).

Коды K_A^1, K_A^2, K_A^3 изображения при сжатии и растяжении, очевидно, меняются. Так, например, код K_A^3 меняется потому, что меняются соотносительные размеры частей изображения (не сохраняется подобие).

Назовем двумерным изображением конечное множество точек на плоскости. Перенумеруем некоторым образом точки изображения A, т.е. зададим функцию нумерации M_A. Пусть S_{mnu} и S_{kps} — площади треугольников с вершинами в тройках точек с номерами m, n, u и k, p, s и пусть $\rho_{mnu,kps} = \frac{S_{mnu}}{S_{kps}}$. Полагаем, что порядок номеров в тройках не важен, сами тройки различны и при $S_{kps} = 0$ значение $\rho_{mnu,kps}$ не определено. Множество индексированных чисел $\rho_{mnu,kps}$ для всех таких пар троек обозначим через T_A. Код K_A^4 изображения A — пара $< M_A, T_A >$. Изображения, все точки которых расположены на одной прямой, не рассматриваем, поскольку код для них не определен. Как и ранее, изображения A и B назовем эквивалентными в смысле случая 4 (далее — просто эквивалентными), если существуют такие нумерации M_A и M_B, что коды T_A и T_B равны. Ясно, что эквивалентность изображений содержательно означает одинаковость их кодов с точностью до перенумерации точек. Если $|A| = N$, то мощность этого кода составляет $C_N^3(C_N^3 - 1)$.

Плоским изображением называется изображение, точки которого не лежат на двух параллельных прямых.

Теорема 1 (В.Н.Козлов, [8]). *Два плоских изображения эквивалентны относительно T_a тогда и только тогда, когда они a-эквивалентны.*

Восстановление трехмерных изображений.

Трехмерным изображением (телом) будем называть конечное множество точек трехмерного евклидова пространства. Как и в плоском случае будем называть аффинно эквивалентными два тела, если существует аффинное преобразование трехмерного пространства, переводящее одно в другое.

Объемным будем называть изображение, если оно не расположено ни в одной плоскости, ни в двух параллельных плоскостях.

Пусть задано тело T и прямая l, называемая направлением проекции. Через каждую точку изображения проводим прямую[2], параллельную l. Рассматриваются точки пересечения этих прямых с некоторой плоскостью Π и обозначаются $P_{l,\Pi}(t)$.

Направления будем называть разными, если соответствующие прямые не параллельны.

Пусть заданы проекции $t'_i = P_{l',\Pi'}$ и $t''_i = P_{l'',\Pi''}$, где $i = 1, \ldots, N$ и l', l'' — разные направления проекции. Тогда возможно восстановить (с точностью до аффинной эквивалентности) исходное тело.

Алгоритм восстановления объемного изображения по его плоским проекциям.

Выберем четыре точки изображения, не лежащие в одной плоскости (по условию такие существуют).

Будем обозначать эти точки a, b, c и d, их проекции на плоскость Π' обозначим a', b', c' и d'; на плоскость Π'' — a'', b'', c'' и d'' соответственно.

- Восстановим положение точек a, b, c и d следующим образом: $\tilde{a}(X(a'), Y(a'), 0)$, $\tilde{b}(X(b'), Y(b'), 0)$, $\tilde{c}(X(c'), Y(c'), 0)$, $\tilde{d}(X(d'), Y(d'), 1)$. Т.е. координаты X и Y просто повторяют координаты проекций на Π' соответствующих точек.

- Расмотрим аффинное преобразования$\mathcal{A}'$, переводящее точки a', b' и c' в точки с координатами $(0, 0)$, $(0, 1)$ и $(1, 0)$, соответственно. Аналогично рассмотрим аффинное преобразования$\mathcal{A}''$, переводящее точки a'', b'' и c'' в точки с такими же координатами: $(0, 0)$, $(0, 1)$ и $(1, 0)$.

- Найдем $s_d = |\mathcal{A}'(d')\mathcal{A}''(d'')|$.

- Проведем цикл по оставшимся точкам изображения (обозначим e переменную точку, по которой идет цикл).

  - Найдем $s_e = |\mathcal{A}'(e')\mathcal{A}''(e'')|$.
  - Восстановим точку $\tilde{e}(X(e'), Y(e'), \frac{s_e}{s_d})$.

[2]Предполагается, что прямая l выбрана так, что на каждой такой прямой расположена одна точка тела

В [2] показано, что указанный алгоритм восстанавливает трехмерное изображение с точностью до аффинной эквивалентности.

Рассмотрим примеры решения задач, связанных с аффинной эквивалентностью.

Пример 1.1. В трехмерном пространстве расположены точки a, b, c, d и e. Известны их проекции на две плоскости. Проекции на Π': $a'(-5,1)$, $b'(-4,2)$, $c'(-6,2)$, $d'(0,4)$ и $e'(4,6)$ (в некоторой системе координат) и проекции на Π'': $a''(-1,5)$, $b''(-2,3)$, $c''(-3,4)$, $d''(7,0)$ и $e''(11,-4)$, соответственно. Известно, что плоскости ABC и CDE пересекаются по прямой L a) Определить уравнения прямых L' и L'' , которые являются проекциями прямой L на плоскости Π' и Π'', соответственно. b) Восстановить положение точек A, B, C, D и E с точностью до аффинной эквивалентности.

Решение пункта а) Рассмотрим аффиннное преобразование, переводящее точки a', b' и c' в точки a'', b'' и c'', соответственно. Из курса линейной алгебры известно, что такое преобразование существует и единственно. Можно найти его матрицу, решив систему линейных уравнений, но в данной задаче можно поступить еще проще. Воспользуемся тем, что аффинное преобразование сохраняет линейную комбинацию векторов и представим вектор $\overline{c'd'} = (6,2)$ в виде линейной комбинации векторов $\bar{v}_1 = \overline{c'a'} = (1,-1)$ и $\bar{v}_2 = \overline{c'b'} = (2,0)$. Легко найти коэффициенты этой линейной комбинации: $\overline{c'd'} = -2 \cdot \bar{v}_1 + 4\bar{v}_2$. Отсюда следует, что $\overline{c''d'''} = -2 \cdot \bar{v'}_1 + 4\bar{v'}_2 = -2 \cdot (2,1) + 4 \cdot (1,-1) = (0,-6)$, где d''' — образ точки d' при вышеуказанном аффинном преобразовании. Следовательно, $d''' = (-3,-2)$. Аналогичным образом найдем координаты точки $e''' = (-4,-7)$. Найдем точку x'' – пересечение прямых $d''e''$ и $d'''e'''$. Ее координаты $x'' = (-1,8)$. Следовательно, проекцией L'' будет прямая, проходящая через точки $c''(-3,4)$ и $x''(-1,8)$, ее уравнение $y = 2x+10$. Разложим теперь вектор $\overline{c''x''} = (2,4)$ как линейную комбинацию векторов $\bar{v''}_1 = \overline{c''a''} = (2,1)$ и $\bar{v''}_2 = \overline{c''b''} = (1,-1)$, получим . $\overline{c''x''} = 2 \cdot \bar{v''}_1 - 2 \cdot \bar{v''}_2$. Отсюда легко получаем координаты точки x' на плоскости Π': $x' = (-8,0)$. Найдем прямую $c'x'$: ее уравнение $y = x+8$.

Ответ: Уравнение прямой L': $y = x + 8$, уравнение прямой L'': $y = 2 \cdot x + 10$.

Решение пункта б) Для упрощения построений рассмотрим аффинные преобразования, переводящие точки $a'(-5,1)$, $b'(-4,2)$, $c'(-6,2)$, и $a''(-1,5)$, $b''(-2,3)$, $c''(-3,4)$ в точки с координатами $a'''(1,0)$, $b'''(0,1)$ и $c'''(0,0)$, соответственно. При таком преобразовании, точки $d'(0,4)$ и $e'(4,6)$ перейдут в $d_1'''(-2,4)$ и $e_1'''(-4,7)$, а точки $d''(7,0)$ и $e''(11,-4)$ — в $d_2'''(2,6)$ и $e_2'''(2,10)$, соответственно. Точки x' и x'' переходят в точку $x'''(2,-2)$. Зададим координаты четырех точек удобным для нас образом: $a(1,0,0)$, $b(0,1,0)$, $c(0,0,0)$, $d(-2,4,1)$, тогда x будет иметь координаты $x'''(2,-2,0)$. Тогда координаты точки e определяются однозначно из условия, что эта точка лежит на прямой xd, причем $\overline{xe} = 3/2 \cdot \overline{xd}$, откуда получаем $e(-4,7,3/2)$.

Ответ: $a(1,0,0)$, $b(0,1,0)$, $c(0,0,0)$, $d(-2,4,1)$ и $e(-4,7,3/2)$.

Пример 1.2. В трехмерном пространстве расположены точки a, b, c, d и e. Известны их проекции на две плоскости. Проекции на Π': $a'(-4,3)$, $b'(-1,2)$, $c'(-3,2)$, $d'(7,0)$ и $e'(-4,5)$ и проекции на Π'': $a''(3,-4)$, $b''(6,-4)$, $c''(5,-5)$, $d''(7,3)$ и $e''(-6,5)$. Восстановите положение точек a, b, c, d и e с точностью до аффинной эквивалентности.

Решение Рассмотрим аффиные преобразования, переводящие точки

$a'(-4,3)$, $b'(-1,2)$, $c'(-3,2)$ и $a''(3,-4)$, $b''(6,-4)$, $c''(5,-5)$ в точки с координатами $\tilde{a}(1,0)$, $\tilde{b}(0,1)$ и $\tilde{c}(0,0)$, соответственно. При таком преобразовании, точки $d'(7,0)$ и $e'(-4,5)$ перейдут в $\tilde{d}_1(-2,4)$ и $\tilde{e}_1(-3,1)$, а точки $d''(7,3)$ и $e''(-6,5)$ — в $\tilde{d}_2(2,6)$ и $\tilde{e}_2(7,3)$, соответсвенно. Заметим, что вектора $\overline{\tilde{d}_1\tilde{e}_1}$ и $\overline{\tilde{d}_2\tilde{e}_2}$ получились равными. Это означает, что прямая de параллельна плоскости abc. Зададим координаты четырех точек удобным для нас образом: $a(1,0,0)$, $b(0,1,0)$, $c(0,0,0)$, $d(-2,4,1)$, тогда из того, что вектор $\overline{de}$ параллелен плоскости abc вытекает , что $e=(3,1,1)$.

Ответ: $a(1,0,0)$, $b(0,1,0)$, $c(0,0,0)$, $d(-2,4,1)$ и $e(3,1,1)$.

Пример 1.3. Докажите, что если аффинное преобразование φ переводит некоторую окружность в себя, то φ — либо поворот, либо симметрия.

Решение Рассмотрим произвольный прямоугольник, вписанный в данную окружность. Его образом будет параллелограмм, вписанный в окружность, т.е. тоже прямоугольник. Следовательно, преобразование φ сохраняет прямые углы. Действительно, для произвольного прямого угла можно построить вписанный прямоугольник, стороны которого параллельны сторонам этого угла.

Впишем произвольный квадрат в окружность. Отображение φ сохраняет прямой угол между диагоналями, следовательно образ тоже будет квадратом. из этого следует, что φ сохраняет расстояние между точками, следовательно, φ — изометрия (движение).

Заметим, что центр окружности совпадает с центром квадрата, т.е. является неподвижной точкой движения φ. Из курса школьной геометрии известно, что любое движение есть параллельный перенос, поворот или скользящая симметрия. Заметим, что неподвижной точкой обладают только поворот и осевая симметрия, что и требовалось доказать.

Пример 1.4. Дан параллелограмм $ABCD$, площадь которого равна S_0. Точки K, L, M и N являются серединами сторон AB, BC, CD и DA, соответственно. Найдите площадь четырехугольника, образованного прямыми AL, BM, CN и DK (см. рис. 1.1).

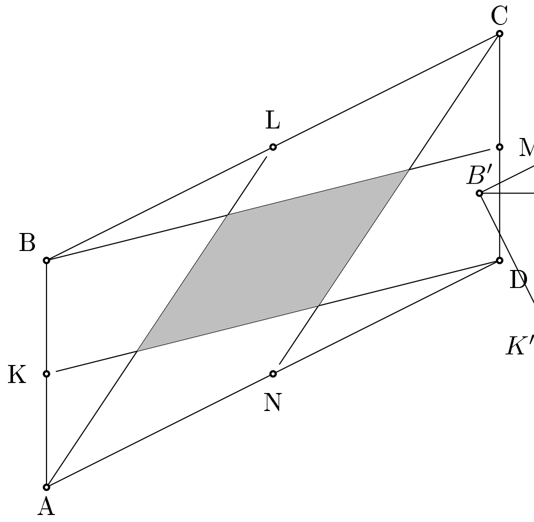


Рис. 1.1: К примеру 1.4.

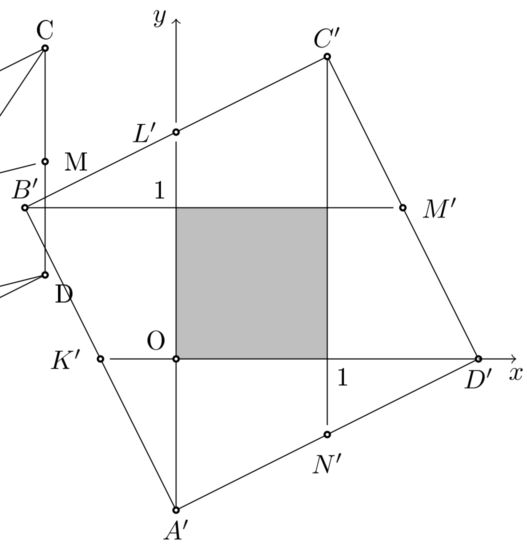


Рис. 1.2: Решение примера 1.4.

Решение. Рассмотрим аффинное преобразование, переводящее указанный четырехугольник в единичный квадрат, так, чтобы одна из вершин перешла в начало координат, а точки D и L попали на оси координат (см. рис. 1.2). Тогда точки A, B, C и D перейдут в точки с координатами $A'(0,-1)$, $B'(-1,-1)$, $C'(1,2)$ и $D'(2,0)$, соответственно. Эти точки образуют квадрат со стороной $\sqrt{5}$. Поскольку отношение площадей сохраняется при аффинном преобразовании, то искомая площадь равна $\frac{1}{5}S_0$.

Пример 1.5. Каждая диагональ выпуклого пятиугольника параллельна одной из его сторон. Докажите, что аффинным преобразованием этот пятиугольник можно перевести в правильный пятиугольник.

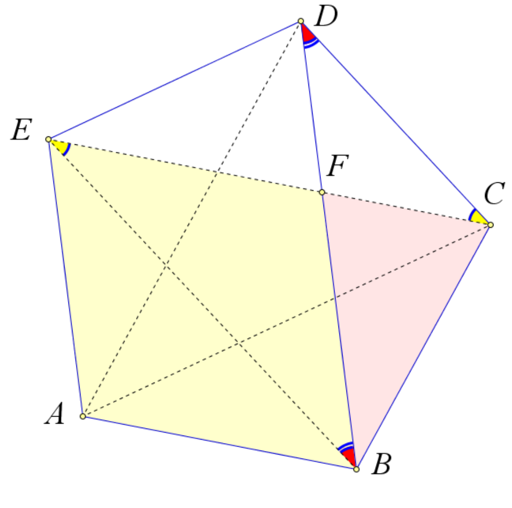


Рис. 1.3

Решение Рассмотрим пятиугольник $ABCDE$. Очевидно, из условия параллельности диагонали EC и стороны AB вытекает равенство площадей треугольников ABC и ABE — у них основание AB общее и высоты равны. Аналогично доказывается, что $S(\triangle ABC) = S(\triangle BCD) = S(\triangle CDE) = S(\triangle DEA) = S(\triangle EAB)$. Будем в дальнейшем считать, что все эти площади равны 1, что не ограничивает общности рассуждений, поскольку нас интересуют только отношения площадей.

Обозначим $AB = a$, $CE = \lambda a$ и запишем площадь трапеции $ABCE$ двумя способами.:

- С одной стороны $S(ABCE) = S(\triangle ABC) + S(\triangle ACE) = 1 + \lambda$, так как $S(\triangle ACE) : S(\triangle ABC) = EC : AB = \lambda$.
- С другой стороны, из подобия $\triangle EBF \cong \triangle CDF$ (где точка F — пересечение диагоналей BD и CE) следует, что $BF : FD = EF : FC = a : (\lambda a - a) = 1 : (\lambda - 1)$. Поэтому $S(\triangle BCF) : S(\triangle BCD) = BF : BD = 1 : \lambda$, следовательно, $S(\triangle BCF) = \frac{1}{\lambda}$, откуда получаем $S(ABCE) = S(ABFE) + S(\triangle BCF) = 2 + \frac{1}{\lambda}$.

Приравнивая полученные выражения для $S(ABCE)$, получим уравнение $1 + \lambda = 2 + \frac{1}{\lambda}$, откуда $\lambda = \frac{1+\sqrt{5}}{2}$ (золотое сечение) [3].

Таким образом, $S(\triangle ABD) = S(\triangle BCE) = S(\triangle DCA) = S(\triangle EBD) = S(\triangle ACE) = \frac{1+\sqrt{5}}{2}$ (подумайте, почему все эти площади равны?).

Заметим, что вышеприведенные рассуждения можно применить и к правильному пятиугольнику. Получается, что коды, т.е. отношения площадей соответствующих треугольников у $ABCDE$ и у правильного пятиугольника совпадают. Согласно теореме В.Н. Козлова эти изображения будут a-эквивалентны, что и требовалось доказать.

[3] Исторически золотое сечение возникло именно как отношение, в котором диагонали правильного пятиугольника делят друг друга.

Упражнения.

Задача 1.1.1. Докажите, что определение аффинного преобразования (повороты, сдвиги, растяжения) из лекции эквивалентно определению в курсе аналитической геометрии(на I курсе).

Задача 1.1.2. Пусть A_1, B_1, C_1, D_1 — образы A, B, C, D при АП. Докажите, что если $\overrightarrow{AB} = \overrightarrow{CD}$ то $\overrightarrow{A_1B_1} = \overrightarrow{C_1D_1}$.

Задача 1.1.3. Пусть A', B', C' — образы точек A, B, C при некотором АП. Докажите, что если C делит отрезок AB в отношении $AC : CB = p : q$, то C' делит отрезок $A'B'$ в том же отношении.

Задача 1.1.4. Являются ли a–эквивалентными а) два ромба; б) два параллелограмма; в) две трапеции?

Задача 1.1.5. а) Докажите, что существует единственное АП, которое переводит данную точку O в данную точку O', а данный базис векторов e_1, e_2 — в данный базис e_1', e_2'. b) Даны два треугольника $\triangle ABC$ и $\triangle A'B'C'$. Докажите, что существует единственное АП, переводящее точку A в A', B — в B' и C — в C'. c) Даны два параллелограмма. Сколькими АП один можно перевести в другой?

Задача 1.1.6. Докажите, что любое аффинное преобразование плоскости можно представить в виде композиции двух растяжений во взаимно перпендикулярных направлениях и одного движения.

Задача 1.1.7. Существует ли линейный по сложности правильный код для множества геометрических преобразований Γ_2?

Задача 1.1.8. Будет ли правильным для множества преобразований Γ_3 код, полученный из кода, описанного в упражнении 1.1.12, делением каждого элемента на $r_{1,2}$?

Задача 1.1.9. Пусть K_A^3, K_B^3 — описанные ранее коды изображений A и B. Оценить сверху сложность (в зависимости от $n = |A| = |B|$) алгоритма , выясняющего эквивалентность этих изображений.

Задача 1.1.10. Придумайте правильный код для множества преобразований Γ_4, который имеет сложность по порядку меньшую, чем сложность кода K_A^4.

Задача 1.1.11. Рассмотрим следующую функцию кодирования: на прямой a_ia_j введем систему координат, в которой точка a_i — начало координат, а отрезок $[a_ia_j]$ задает масштаб (т.е. точка a_j имеет координату 1). Обозначим $\varkappa_{ij,kl}$ координату точки пересечения с прямой a_ka_l в этой системе координат. Если прямые параллельны, считаем, что соответствующий $\varkappa_{ij,kl}$ не определен. Докажите, что код $\{\varkappa_{ij,kl}\}_{i,j,k,l=1}^n$ является правильным для семейства Γ_4.

Задача 1.1.12. Пусть M_A — функция нумерации изображения A, $|A| = n$,

$$r_{i,j} = \sqrt{(X(M_A^{-1}(i) - X(M_A^{-1}(j))^2 + (Y(M_A^{-1}(i) - Y(M_A^{-1}(j))^2},$$

$$T_A = \{r_{1,1}, r_{1,3}, r_{2,3}, r_{1,4}, r_{2,4}, r_{3,4}, \ldots, r_{1,n}, r_{2,n}, r_{3,n}\}.$$

Будет ли код $< M_A, T_A >$ правильным для множества преобразований Γ_2?

Задача 1.1.13. Пусть аффинное преобразование $\mathcal{A}(x) = Ax + b$ обладает свойствами: $\mathcal{A}^2 \equiv \mathcal{A}$ и $\mathrm{rk}(A) = 2$. Докажите, что это преобразование — проекция на некоторую плоскость.

Задача 1.1.14. Дан куб $ABCDA_1B_1C_1D_1$, известны координаты проекций его вершин на некоторую плоскость: $A'(-1,-2)$, $C'(3,4)$, $B_1'(7,0)$, $D_1'(1,6)$. Найдите координаты проекций остальных вершин куба.

Задача 1.1.15. Квадрат $ABCD$ расположен в плоскости α. В различных (относительно α) полупространствах выбраны точки S и S_1, так, что $SABCDS_1$ — правильный октаэдр. Известны координаты проекций на некоторую плоскость β следующих вершин: $A'(1,2)$, $B'(-1,3)$, $C'(0,4)$ и $S'(0,0)$. Найдите координаты проекций вершин D' и S_1'.

Задача 1.1.16. Дана правильная шестиугольная призма $ABCDEFA_1B_1C_1D_1E_1F_1$. Известны координаты проекций на некоторую плоскость вершин $A'(-1,-2)$, $B'(1,2)$, $D'(1,4)$ и $E_1'(-2,-2)$. Найдите координаты проекций остальных вершин.

Задача 1.1.17. В правильном тетраэдре $ABCD$ проведена прямая l, соединяющая центры граней ABC и ABD. Известны проекции вершин на некоторую плоскость: $A'1,3)$; $B'(-3,0)$; $C'(2,3)$; $D'(-1,0)$. Найдите проекцию прямой l (напишите уравнение прямой).

Задача 1.1.18. Известны проекции вершин правильного тетраэдра $ABCD$ на некоторую плоскость: $A'(-3,1)$; $B'(0,-3)$; $C'(-3,4)$; $D'(0,-1)$. Найдите проекцию высоты AH, опущенной на грань BCD (напишите уравнение прямой).

Задача 1.1.19. В правильной 4-угольной пирамиде $SABCD$ на ребре SC выбрана точка K, которая делит ребро в отношении $SK : KC = 2 : 1$. Известны проекции вершин $A'(0,1)$, $C'(2,3)$ и $S'(-1,0)$ на некоторую плоскость. Найдите проекцию на эту же плоскость прямой OK, где O — центр основания $ABCD$.

Задача 1.1.20. Известны проекции следующих вершин правильной 6-угольной призмы $ABCDEFA_1B_1C_1D_1E_1F_1$ на некоторую плоскость: $A'(-2,1)$, $C'(1,2)$, $B_1'(0,4)$ и $D_1'(-3,6)$. Найдите положение проекций остальных вершин.

Задача 1.1.21. В трехмерном пространстве расположены точки a, b, c, d и e. Известны их проекции на две плоскости. Проекции на Π': $a'(-5,3)$, $b'(-4,4)$, $c'(-6,4)$, $d'(-3,3)$ и $e'(-3,4)$ (в некоторой системе координат) и проекции на Π'': $a''(-2,5)$, $b''(-2,4)$, $c''(-3,4)$, $d''(-6,3)$ и $e''(-4,3)$, соответственно. Известно, что плоскости abc и cde пересекаются по прямой L Определить уравнения прямых L' и L'' , которые являются проекциями прямой L на плоскости Π' и Π'', соответственно.

Задача 1.1.22. В пространстве расположены точки a, b, c, d и e. Известны их проекции $a'(3,4)$, $b'(5,2)$, $c'(3,3)$, $d'(2,11)$ и $e'(5,-1)$ на плоскость Π' и проекции $a''(-1,2)$, $b''(0,1)$, $c''(-3,0)$, $d''(5,11)$ и $e''(5,2)$ на плоскость Π''. Пусть L — прямая пересечение плоскостей (abc) и (cde).

a) Найдите проекции на Π' и Π'' точки X, в которой прямая L пересекает прямую de;
b) Выпишите уравнения прямых L' и L'' — проекций L;
c) Восстановите положение точек a, b, c, d и e с точностью до аффинной эквивалентности.

Задача 1.1.23. В пространстве расположены точки a, b, c, d и e. Известны их проекции на Π': $a'(0,1)$, $b'(4,1)$, $c'(1,0)$, $d'(1,4)$ и $e'(17/2,5/2)$ и проекции на Π'': $a''(-1,1)$, $b''(-4,3)$, $c''(-3,2)$, $d''(3,-2)$ и $e''(-6,11/2)$.

a) Найти координаты точек x' и x''.
b) Найти уравнения проекций прямой L.
c) Восстановить трехмерное изображение (с точностью до аффинной эквивалентности).

Задача 1.1.24. В пространстве расположены точки a, b, c, d и e. Известны их проекции на Π': $a'(0,1)$, $b'(4,1)$, $c'(1,0)$, $d'(-2,3)$ и $e'(7,6)$ и проекции на Π'': $a''(-1,1)$, $b''(-4,3)$, $c''(-3,2)$, $d''(3,-2)$ и $e''(0,5/2)$. Найти проекции прямой L и восстановить трехмерное изображение (с точностью до аффинной эквивалентности).

Задача 1.1.25. В трехмерном пространстве расположены точки a, b, c, d и e. Известны их проекции на две плоскости. Проекции на Π': $a'(-5,5)$, $b'(-4,8)$, $c'(-6,8)$, $d'(3,5)$ и $e'(6,2)$ (в некоторой системе координат) и проекции на Π'': $a''(-2,5)$, $b''(-2,4)$, $c''(-3,4)$, $d''(-2,7)$ и $e''(-2,9)$, соответственно. Известно, что плоскости abc и cde пересекаются по прямой L a) Определить уравнения прямых L' и L'' , которые являются проекциями прямой L на плоскости Π' и Π'', соответственно. b) Восстановить положение точек a, b, c, d и e с точностью до аффинной эквивалентности.

Задача 1.1.26. В трехмерном пространстве расположены точки a, b, c, d и e. Известны их проекции на две плоскости. Проекции на Π': $a'(0,1)$, $b'(4,1)$, $c'(1,0)$, $d'(-3,4)$ и $e'(9,4)$ (в некоторой системе координат) и проекции на Π'': $a''(-1,1)$, $b''(-4,3)$, $c''(-3,2)$, $d''(3,-2)$ и $e''(-3,4)$, соответственно. Известно, что плоскости abc и cde пересекаются по прямой L a) Определить уравнения прямых L' и L'' , которые являются проекциями прямой L на плоскости Π' и Π'', соответственно. b) Восстановить положение точек a, b, c, d и e с точностью до аффинной эквивалентности.

Задача 1.1.27. В трехмерном пространстве расположены точки a, b, c, d и e. Известны их проекции на две плоскости: проекции на Π': $a'(2,1)$, $b'(3,-1)$, $c'(1,0)$, $d'(5,4)$ и $e'(8,-11)$ и проекции на Π'': $a''(-1,1)$, $b''(-4,3)$, $c''(-3,2)$, $d''(3,-2)$ и $e''(-15,13)$ Известно, что плоскости abc и cde пересекаются по прямой L Определить уравнения прямых L' и L'', которые являются проекциями прямой L на плоскости Π' и Π'', соответственно.

Задача 1.1.28. В трехмерном пространстве расположены точки a, b, c, d и e. Известны их проекции на две плоскости: проекции на Π': $a'(0,0)$, $b'(-1,-2)$, $c'(1,-1)$, $d'(-3,3)$ и $e'(-3,6)$ и проекции на Π'': $a''(-1,2)$, $b''(-2,4)$, $c''(-2,2)$, $d''(0,-2)$ и $e''(0,-8)$ Известно, что плоскости abc и cde пересекаются по прямой L Определить уравнения прямых L' и L'', которые являются проекциями прямой L на плоскости Π' и Π'', соответственно.

Задача 1.1.29. В трехмерном пространстве расположены точки a, b, c, d и e. Известны их проекции на две плоскости: проекции на Π': $a'(-3,3)$, $b'(-1,3)$, $c'(-3,2)$, $d'(5,4)$ и $e'(-1,6)$ и проекции на Π'': $a''(-1,-1)$, $b''(1,-1)$, $c''(1,-2)$, $d''(-3,5)$ и $e''(-13,7)$ Известно, что плоскости (abc) и (cde) пересекаются по прямой L. a) Определить уравнения прямых L' и L'', которые являются проекциями прямой L на плоскости Π' и Π'', соответственно. b) Восстановить трехмерное изображение с точностью до аффинной эквивалентности

Задача 1.1.30. Дан выпуклый N-угольник, где $N < 100$, и множество $A = \{A_1, \ldots, A_{100}\}$ некоторых точек этого многоугольника. Известно, что все N вершин N-угольника принадлежат множеству A, и что никакие 3 точки не лежат на одной прямой, а никакие 4 точки не лежат на двух параллельных прямых. Разрешается задавать вопросы типа: чему равна площадь $\triangle A_i B_j C_k$? Докажите, что 300 вопросов достаточно, чтобы выяснить, какие точки являются вершинами многоугольника, и чтобы найти площадь многоугольника.

Задача 1.1.31. На плоскости дан многоугольник $A_1 A_2 \ldots A_n$ и точка O внутри его. Докажите, что равенства:

$$\begin{aligned}
\overrightarrow{OA_1} + \overrightarrow{OA_3} &= 2\cos(2\pi/n)\overrightarrow{OA_2}\\
\overrightarrow{OA_2} + \overrightarrow{OA_4} &= 2\cos(2\pi/n)\overrightarrow{OA_3}\\
&\ldots\\
\overrightarrow{OA_{n-1}} + \overrightarrow{OA_1} &= 2\cos(2\pi/n)\overrightarrow{OA_n}
\end{aligned}$$

необходимы и достаточны для того, чтобы существовало аффинное преобразование, переводящее данный многоугольник в правильный, а точку O — в его центр.

Задача 1.1.32. Через каждую вершину треугольника проведены две прямые, делящие противоположную сторону треугольника на три равные части. Докажите, что диагонали, соединяющие противоположные вершины шестиугольника, образованного этими прямыми, пересекаются в одной точке.

Задача 1.1.33. В параллелограмме $ABCD$ точка K — середина стороны AB, L — середина BC, M — середина CD и N — середина DA. Найдите площадь четырехугольника, образованного прямыми AL, BM, CN и DK, если $S(ABCD) = 25$.

Задача 1.1.34. В треугольнике $\triangle ABC$ на стороне AB выбрана точка M, на BC — точка L и на AC — точка N так, что $AM : MB = 3 : 2$, $BL : LC = 1 : 2$ и $AN = NC$. Прямые AL и MC пересекаются в точке O. Определите, какую часть площади $\triangle ABC$ составляет площадь $\triangle BON$.

Задача 1.1.35. На плоскости даны три вектора a, b, c, причем $\alpha a + \beta b + \gamma c = 0$. Докажите, что эти векторы аффинным преобразованием можно перевести в векторы равной длины тогда и только тогда, когда из отрезков с длинами $|\alpha|$, $|\beta|$, $|\gamma|$ можно составить треугольник.

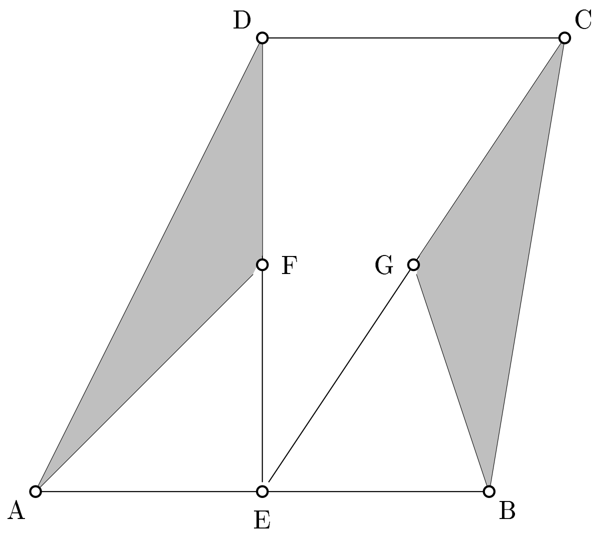


Рис. 1.4: К задаче 1.1.38.

Задача 1.1.36. Внутри треугольника $\triangle ABC$ лежит точка M. Докажите, что площади треугольников $\triangle ABM$ и $\triangle CBM$ равны тогда и только тогда, когда точка M находится на медиане угла $\angle B$ треугольника $\triangle ABC$.

Задача 1.1.37. Докажите, что в произвольной трапеции точка пересечения диагоналей, точка пересечения продолжений боковых сторон и середины оснований расположены на одной прямой.

Задача 1.1.38. Пусть $ABCD$ — трапеция с основаниями $AB \| CD$, точка E — середина основания AB, точки F и G — середины отрезков DE и CE, соответственно (см. рис. 1.4). Докажите, что треугольники $\triangle AFD$ и $\triangle BCG$ равновелики.

Задача 1.1.39. В выпуклом четырехугольнике $ABCD$ продолжения сторон AB и CD пересекаются в точке Z, точки X и Y являются серединами диагоналей AC и BD. Найдите отношение площадей $S(\triangle XYZ) : S(ABCD)$.

Задача 1.1.40. Через точку O, взятую на диагонали AC параллелограмма $ABCS$, проведены прямые, параллельные его сторонам. Данный параллелограмм делится на четыре параллелограмма. Два из них пересекаются диагональю AC (см. рис. 1.5).

a) Докажите, что два других параллелограмма равновелики.
b) Сформулируйте и докажите обратное утверждение.

Задача 1.1.41. Дан треугольник ABC. Пусть O — точка пересечения его медиан, а M, N и P — точки, лежащие на сторонах AB, BC и CA, соответственно, делящие эти стороны в одинаковых отношениях (т. е. $AM : MB = BN : NC = CP : PA = p : q$). Докажите, что:

a) O — точка пересечения медиан треугольника MNP;

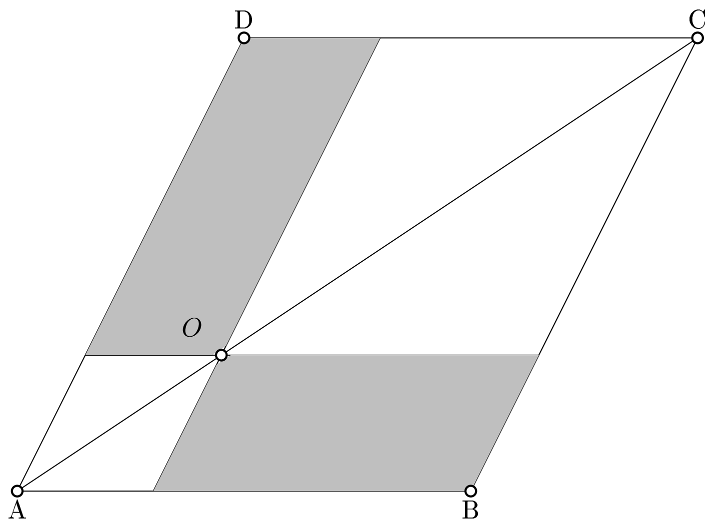


Рис. 1.5: К задаче 1.1.40.

b) O — точка пересечения медиан треугольника, образованного прямыми AN, BP и CM.

Задача 1.1.42. В трапеции $ABCD$ с основаниями AD и BC через точку B проведена прямая, параллельная стороне CD и пересекающая диагональ AC в точке P, а через точку C — прямая, параллельная стороне AB и пересекающая диагональ BD в точке Q. Докажите, что прямая PQ параллельна основаниям трапеции.

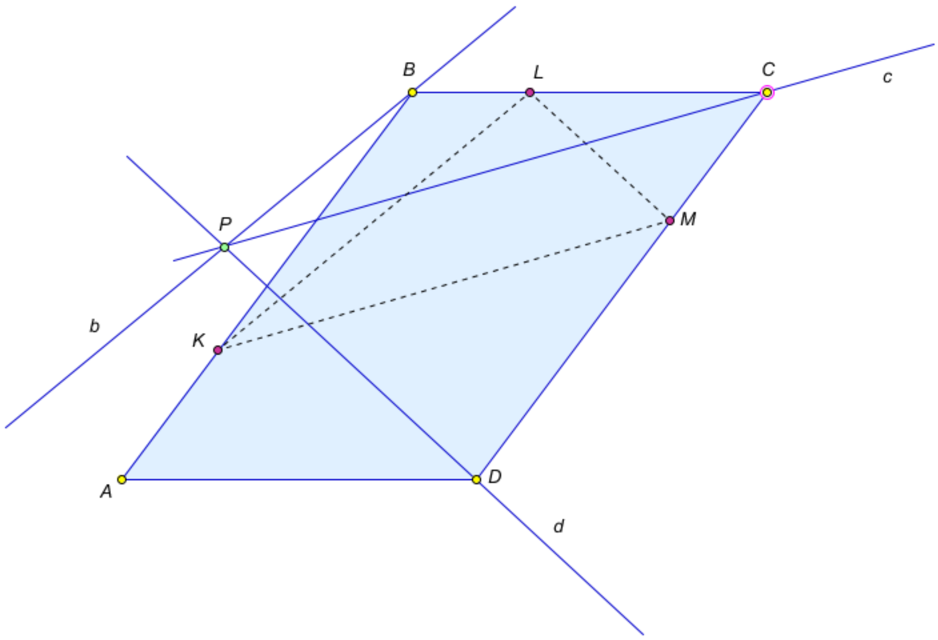
Задача 1.1.43. В эллипс вписан пятиугольник, в котором каждая диагональ параллельна противолежащей стороне. Найти отношение площади эллипса к площади пятиугольника.

Задача 1.1.44. * (Прасолов) На сторонах AB, BC и CD параллелограмма $ABCD$ взяты точки K, L и M соответственно, делящие эти стороны в одинаковых отношениях $AK : KB = BL : LC = CM : MD = \lambda$. Пусть b, c, d — прямые, проходящие через B, C, D параллельно прямым KL, KM, ML соответственно. Докажите, что прямые b, c, d проходят через одну точку.

Задача 1.1.45. * Докажите, что любой выпуклый четырехугольник, кроме трапеции, аффинным преобразованием можно перевести в четырехугольник, у которого противоположные углы прямые.

Задача 1.1.46. * Дан выпуклый шестиугольник $ABCDEF$, в котором $AB \| DE$, $BC \| EF$ и $CD \| FA$. Доказать, что найдется аффинное преобразование $\mathcal{L} : ABCDEF \to A'B'C'D'E'F'$, такое, что $A'D' = B'E' = C'F'$.

Задача 1.1.47. * Ковбой Джимми поспорил с друзьями, что сумеет одним выстрелом пробить все четыре лопасти вертилятора. (Вертилятор устроен следующим образом: на оси, вращающейся со скоростью 50 об/сек, расположены на равных расстояниях друг от друга 4 полудиска, повернутые друг относительно друга под какими-то углами). Джимми может стрелять



К задаче 1.1.44

в любой момент и добиваться произвольной скорости пуль. Доказать, что Джимми выиграет пари.

Задача 1.1.48. * а) Пусть L — непрерывное взаимно однозначное отображение плоскости в себя, обладающее свойством: если три точки лежат на одной прямой, то их образы тоже лежат на одной прямой. Докажите, что L — аффинное преобразование. б) Пусть L — непрерывное взаимно однозначное отображение плоскости в себя, переводящее любую окружность в некоторую окружность. Докажите, что L — аффинное преобразование.

Задача 1.1.49. (Руденко) Пусть проекции A' и A'' тела A в двух различных направлениях являются эквивалентными, но не a-эквивалентными. Докажите, что точки восстановленного с помощью алгоритма тела лежат на двух прямых.

Ответы и указания.

1.1.2 Будем записывать координаты в виде столбца $X = \binom{x}{y}$. Тогда $X_1 = M \cdot X + \mu$, где M — невырожденная матрица, μ — вектор-столбец. Тогда $\overrightarrow{A_1B_1} = B_1 - A_1 = (M \cdot B + \mu) - (M \cdot A + \mu) = M \cdot (B - A) = M \cdot \overrightarrow{AB}$. Таким образом, при аффинном преобразовании вектора умножаются на матрицу M, т.е. равные вектора переходят в равные.

1.1.3 Если C делит отрезок AB в отношении $AC : CB = p : q$, то $\overrightarrow{AB} = \frac{p}{q} \cdot \overrightarrow{CB}$. Следовательно, $\overrightarrow{A_1B_1} = \frac{p}{q} \cdot \overrightarrow{C_1B_1}$, т.е. C' делит отрезок $A'B'$ в том же отношении.

1.1.4 а) Да. б) Да. в) Только в случае, если у них одинаковые отношения оснований.

1.1.5 а) Будем считать, что координаты записаны в виде столбцов. Рассмотрим систему координат с началом в точке O и репером e_1, e_2. Тогда выберем $M = (e_1', e_2')$, $b = O'$. Тогда $\mathcal{A}(O) = O_1$, $\mathcal{A}(A) = A_1$, $\mathcal{B}(O) = B_1$. Единственность следует из того, что должны выполняться равенства $\begin{cases} M \cdot \binom{1}{0} & = e_1' \\ M \cdot \binom{0}{1} & = e_2' \\ M \cdot \binom{0}{0} + b & = O' \end{cases}$ б) воспользуемся пунктом а) и выберем в качестве начала координат точки A и A_1, а в качестве базисов $e_1 = \overrightarrow{AB}$, $e_2 = \overrightarrow{AC}$,$e_1' = \overrightarrow{A'B'}$, $e_2' = \overrightarrow{A'C'}$. в) 8. Выделенную вершину можно перевести в любую из 4, кроме того, возможны два варианта ориентации репера.

1.1.6 Пусть преобразование задано как $x \to y = Ax + b$. Найдем для матрицы A сингулярное разложение: $A = U\Sigma V^T$, где U и V — ортогональные матрицы, а Σ — диагональная. Перейдем в систему координат, образованную столбцами матрицы V: $x = Vx'$. В новой системе координат отображение запишется как $y' = V^T y = V^T U \Sigma x' + V^T b$. Тогда его можно представить как композицию двух растяжений $x' \to x_1' = \Sigma x'$ и движения $x_1' \to y' = V^T U x_1' + V^T b$ (это движение, т.к. матрица $V^T U$ тоже ортогональна).

1.1.7 Да. Достаточно рассматривать расстояния до трех точек, образующих треугольник (т.е не лежащих на одной прямой).

1.1.8 Да, будет. Достаточно заметить что если известны все $\lambda_{ij} = \frac{r_{ij}}{r_{12}}$, то $\rho_{ij,kl} = \frac{r_{ij}}{r_{kl}} = \frac{\lambda_{ij}}{\lambda_{kl}}$.

1.1.9 Заметим, что если зафиксировать две точки изображения a' и a'', то по коду изображение восстанавливается почти однозначно — с точностью до симметрии относительно прямой $a'a''$ и сложность алгоритма восстановления составляет $O(n)$. Зафиксируем пару точек изображения A, a_{i_0} и a_{j_0}, например, выберем такие, у которых $\rho_{i_0 j_0, k_0 l_0} = \max \rho_{ij,kl}$ (это означает, что точки a_{i_0} и a_{j_0} отстоят друг от друга на наибольшее расстояние). Ей должна соответствовать пара $b_{i_0'}$, $b_{j_0'}$, для которой значение кода такое же. Заметим, что таких пар не может быть больше $C_n^2 = O(n^2)$. Таким образом, сложность алгоритма составляет не более $O(n^3)$.

1.1.10 Выбираем треугольник наибольшей площади $\triangle a_i a_j a_k$ (если таких несколько, то выбираем любой). Рассматриваем отношения $\tau_{lmn} = \frac{S(\triangle a_l a_m a_n)}{S(\triangle a_i a_j a_k)}$, таких будет $C_n^3 = O(n^3)$.

1.1.11 Указание: испольуйте то, что $|\varkappa_{ij,kl} : (1 - \varkappa_{ij,kl})| = \frac{S(\triangle a_i a_k a_l)}{S(\triangle a_j a_k a_l)}$

1.1.12 Да. По $r_{1,1}, r_{1,3}, r_{2,3}$ восстанавливается треугольник (или три точки на одной прямой). Потом добавляются точки (по одной).

1.1.13 Пусть $\mathcal{A}(x) = Ax + b$. В качестве направления проектирования выберем вектор v_0, такой, что $Av_0 = 0$, существование такого следует из вырожденности преобразования. В качестве плоскости берем $\Pi = Im(\mathcal{A})$. Выберем произвольную точку x и пусть $x_0 \in \Pi$ — проекция точки x на плоскость Π по направлению вектора v_0, т.е. $x = x_0 + \lambda v_0$. Тогда $\mathcal{A}(x_0) = x_0$ и $\mathcal{A}(x) = \mathcal{A}(x_0 + \lambda v_0) = \mathcal{A}(x_0) + \lambda A v_0 = x_0$.

1.1.14 Обозначим $\overrightarrow{a} = \overrightarrow{A'B'}$, $\overrightarrow{b} = \overrightarrow{A'D'}$ и $\overrightarrow{c} = \overrightarrow{A'A_1'}$. Тогда $\overrightarrow{a} + \overrightarrow{b} = \overrightarrow{A'C'} = (4,6)$, $\overrightarrow{a} + \overrightarrow{c} = \overrightarrow{A'B_1'} = (8,2)$ и $\overrightarrow{b} + \overrightarrow{c} = \overrightarrow{A'D_1'} = (2,8)$. Сложив эти вектора и разделив на 2, получим $\overrightarrow{a} + \overrightarrow{b} + \overrightarrow{c} = (7,8)$. Отсюда легко найти вектора $\overrightarrow{a} = (5,0)$, $\overrightarrow{b} = (-1,6)$, $\overrightarrow{c} = (3,2)$. Отложив эти вектора от точки A' восстанавливаем проекции вершин $C_1'(6,6)$, $B'(4,-2)$, $D'(-2,4)$ и $A_1'(2,0)$.

1.1.15 $D'(2,3)$, $S_1'(1,6)$. Решение аналогично задаче 1.1.14.

1.1.16 $A_1'(-2,-4)$; $B_1'(0,0)$; $C'(2,5)$; $C_1'(1,3)$; $D_1'(0,2)$; $E'(-1,0)$; $F'(-2,-3)$; $F_1'(-3,-5)$. Решение аналогично задаче 1.1.14.

1.1.17 Проекции центров будут $O_1'(0,2)$ и $O_2'(-1,1)$; уравнение прямой $y = x + 2$.

1.1.18 Проекция центра грани (BCD) — точки H имеет координаты $H'(-1,0)$, прямая, проходящая через нее и $A'(-3,1)$ имеет уравнени $y = \frac{-x-1}{2}$.

1.1.19 Точка $(1,2)$.

1.1.20 Обозначим $\overline{A'B'} = \bar{u}$, $\overline{A'F'} = \bar{v}$ и $\overline{A'A_1'} = \bar{w}$. Составим систему:

$$\begin{cases} \overline{A'C'} = 2\bar{u} + \bar{v} & = (3,1); \\ \overline{A'B_1'} = \bar{u} + \bar{w} & = (2,3); \\ \overline{A'D_1'} = 2\bar{u} + 2\bar{v} + \bar{w} & = (-1,5). \end{cases}$$

Решив ее, получим $\bar{u} = (3,0)$, $\bar{v} = (-3,1)$ и $\bar{w} = (-1,3)$. Отсюда легко найти проекции всех оставшихся вершин: $B'(1,1)$, $D'(-2,3)$, $E'(-5,3)$, $F'(-5,2)$, $A_1'(-3,4)$, $C_1'(0,5)$, $E_1'(-6,6)$ и $F_1'(-6,5)$.

1.1.21 a) уравнение прямой L' $y = 1/3 \cdot x + 6$, уравнение прямой L'' $y = -x + 1$

1.1.22a) $x'(3,7)$ и $x''(5,8)$;
b) L': $x = 3$ и L'': $y = x + 3$;
c) $a(1,0,0)$, $b(0,1,0)$, $c(0,0,0)$, $d(5,-1,1)$ и $e(2,2,-2)$.

1.1.23 $x'(6,3)$, $x''(-3,3)$; уравнение прямой L': $y = 3/5 \cdot x - 3/5$ уравнение прямой L'': $x = -3$ Трехмерные координаты $a(1,0,0)$, $b(0,1,0)$, $c(0,0,0)$, $d(3,1,1)$ и $e(0,5/2,-1/2)$

1.1.24 $x'(4,5)$, $x''(1,1)$ уравнение прямой L' $y = 5/3 \cdot x - 5/3$ уравнение прямой L'' $y = -1/4 \cdot x + 5/4$

Трехмерные координаты $a(1,0,0)$, $b(0,1,0)$, $c(0,0,0)$, $d(3,0,1)$ и $e(3,3,-1/2)$

1.1.25 a) уравнение прямой L' $y = x + 14$, уравнение прямой L'' $y = -x + 1$; b) Трехмерные координаты $a(1,0,0)$, $b(0,1,0)$, $c(0,0,0)$, $d(1,4,1)$ и $e(2,5,3/2)$

1.1.26 a) уравнение прямой L' $y = x - 1$ уравнение прямой L'' $y = 2$ b) Трехмерные координаты $a(1,0,0)$, $b(0,1,0)$, $c(0,0,0)$, $d(4,0,1)$ и $e(1,3,-1/2)$.

1.1.27 a) Уравнение прямой L': $y = \frac{-x+1}{5}$; уравнение прямой L'': $x = -3$. b)Трехмерные координаты $a(1,0,0)$, $b(0,1,0)$, $c(0,0,0)$, $d(4,0,1)$ и $e(-5,6,-2)$.

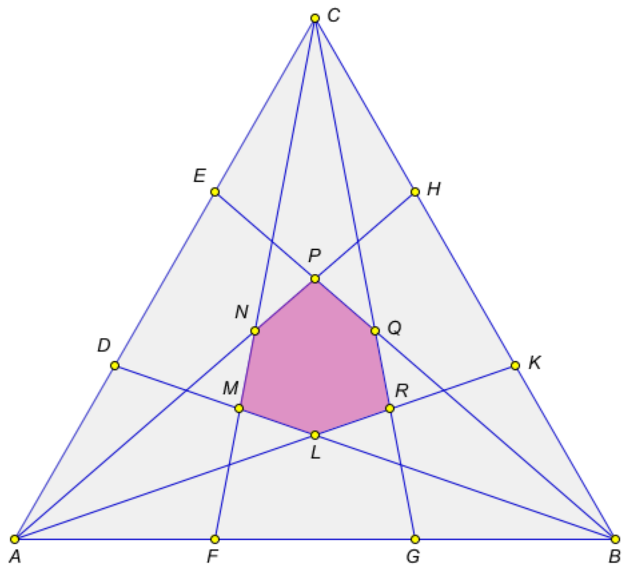
1.1.28 a) Уравнение прямой L': $y = -\frac{1}{4} \cdot x - \frac{3}{4}$, уравнение прямой L'': $y = x + 4$. b)Трехмерные координаты $a(1,0,0)$, $b(0,1,0)$, $c(0,0,0)$, $d(4,0,1)$ и $e(6,-1,2)$.

1.1.29 a) Уравнение прямой L': $y = -\frac{1}{3} \cdot x + 1$; уравнение прямой L'' : $y = -\frac{1}{5} \cdot x - \frac{9}{5}$. b)Трехмерные координаты $a(1,0,0)$, $b(0,1,0)$, $c(0,0,0)$, $d(-2,4,1)$ и $e(3,1,1)$.

1.1.30 Восстановите многоугольник с точностью до аффинной эквивалентности.

1.1.31 Аффинным преобразованием переведите трегольник OA_1A_2 в $O'A'_1A'_2$, где $A'_1A'_2\dots A'_n$ — правильный n−угольник с центром O' .

1.1.32 Аффинным преобразованием переведите треугольник в правильный. Тогда прямые, соединяющие противоположные вершины шестиугольника будут проходить через центр треугольника.



1.1.33 Аффинным преобразованием переводим $ABCD$ в квадрат $A'B'C'D'$ со стороной 2, выберем систему координат с началом в точке A' и осями, проходящими через D (ось x) и B (ось y). Тогда уравнения прямых AL: $y = 2x$; BM: $y = -\frac{1}{2}x + 2$; CN: $y = 2x - 2$ и DK: $y = -\frac{1}{2}x + 1$. Эти прямые образуют квадрат со стороной $\frac{\sqrt{20}}{5}$. Поскольку аффинные преобразования сохраняют отношение площадей, то $S(ABCD) : S(PQRT) = S(A'B'C'D') : : S(P'Q'R'T')$, следовательно, $S(PQRT) = \frac{4}{5} \cdot 25 : 4 = 5$.

1.1.34 Переведем аффинным преобразованием треугольник ABC в треугольник с вершинами $A'(10, 0)$, $B'(0, 0)$, $C'(0, 6)$. Тогда образы остальных точек будут иметь координаты: $L'(0, 2)$, $M'(4, 0)$, $N'(5, 3)$ и $O'(\frac{40}{13}, \frac{18}{13})$. Найдем площади $S(\triangle A'B'C') = 30$, $(\triangle B'O'N') = \frac{15}{13}$, следовательно, т.к. аффинные преобразования сохраняют отношения площадей, $S(\triangle BON) = \frac{1}{26}S(\triangle ABC)$.

1.1.35 Треугольник составленный из векторов αa, βb и γc аффинным преобразованием переведите в треугольник со сторонами $|\alpha|$, $|\beta|$, $|\gamma|$.

1.1.36 Аффинным преобразованием переведите треугольник в равнобедренный.

1.1.37 Аффинным преобразованием переводим трапецию в равнобедренную. Тогда образы всех указанных точек расположены на оси симметрии трапеции.

1.1.38 Переведите трапецию аффинным преобразованием в равнобокую, тогда треугольники перейдут в равные.

1.1.39 Рассмотрим аффинное преобразование, переводящее вектора $\overline{AB}$ и $\overline{DC}$ в вектора единичного репера, а точку Z — в начало координат. Пусть образы точек A и C имеют координаты $A'(a, 0)$ и $C'(0, c)$, соответственно. Тогда координаты оставшихся вершин будут $B'(a + 1, 0)$, $D'(0, c - 1)$ (см. рис. 1.7) и площадь $S(A'B'C'D') = S(\triangle Z'B'C') - S(\triangle Z'A'D') = \frac{1}{2}(a+1)c - - \frac{1}{2}a(c-1) = \frac{1}{2}(c + a)$.

Найдем координаты середин диагоналей четырехугольника $A'B'C'D'$: $X'\left(\frac{a}{2}, \frac{c}{2}\right)$ и $Y'\left(\frac{a+1}{2}, \frac{c-1}{2}\right)$. Тогда площадь треугольника $S(\triangle X'Y'Z') = = \frac{1}{2}\det\begin{pmatrix} (a+1)/2 & (c-1)/2 \\ a/2 & c/2 \end{pmatrix} = \frac{1}{8}\big((a+1)c - a(c-1)\big) = \frac{1}{4}S(A'B'C'D')$. Следовательно, $S(\triangle XYZ) : S(ABCD) = S(\triangle X'Y'Z') : S(A'B'C'D') = 1 : : 4$.

1.1.40 а) Аффинным преобразованием переведем параллелограмм в квадрат. Тогда указанные параллелограммы переходят в равные прямоугольники.

1.1.41 а), b) Аффинным преобразованием переведите треугольник в правильный.

1.1.42 Аффинным преобразование переведем трапецию в равнобедренную. Например, обозначим M и N середины оснований AD и BC, переведем точку M в начало координат, а вектора $\overrightarrow{MN}$ и $\overrightarrow{MD}$ — в единичный

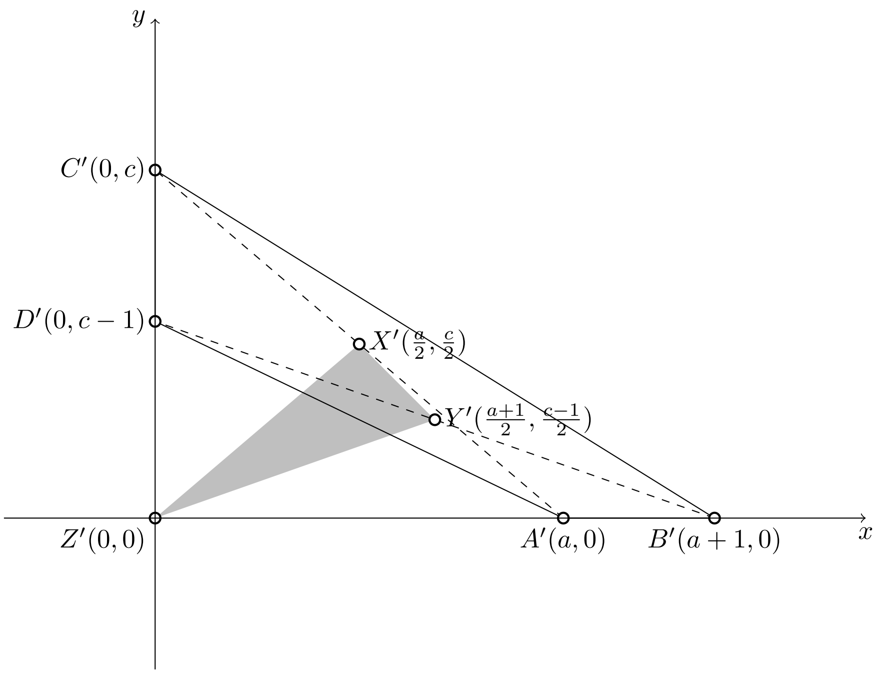
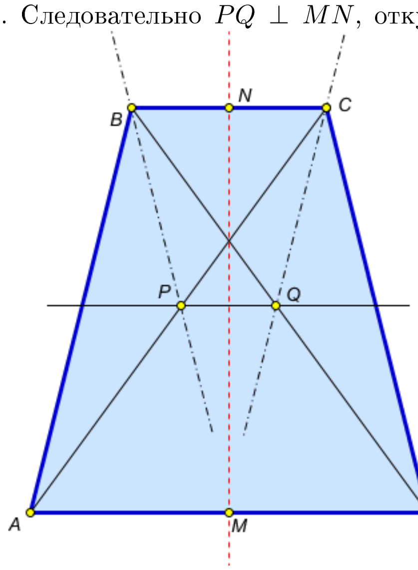


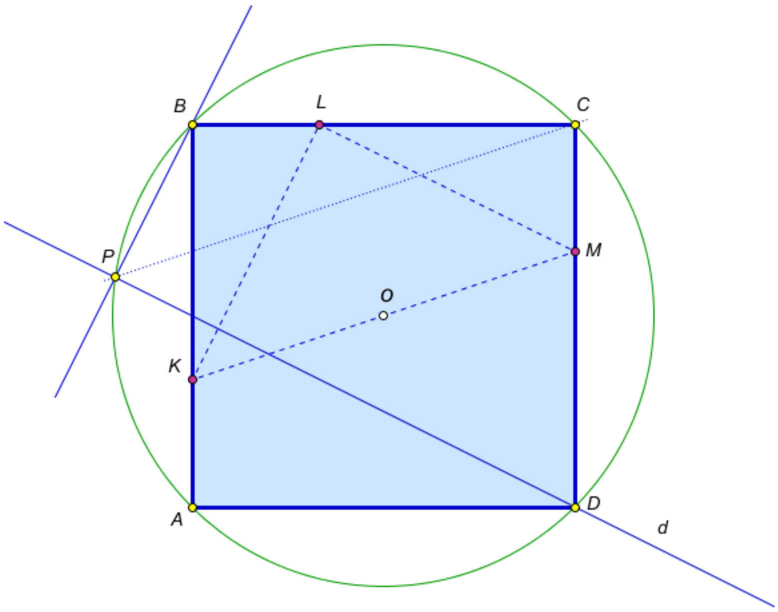
Рис. 1.7: К задаче 1.1.39.

репер. Тогда фигура симметрична относительно прямой MN, откуда и вытекает, что точки P и Q тоже симметричны относительно этой прямой. Следовательно $PQ \perp MN$, откуда и следует искомое утверждение.



1.1.43 Рассмотрим аффинное преобразование, которое переводит эллипс в окружность. Воспользовавшись тем, что параллельные прямые высекают на окружности равные дуги, получим, что вершины пятиугольника разбивают окружность на 5 равных дуг. Следовательно, пятиугольник правильный.

Осталось заметить, что аффинное преобразование сохраняет отношение площадей, значит площадь круга равна πr^2, а 5–угольника (разбиваем на 5 треугольников с вершиной в центре окружности) равна $5 \cdot \frac{1}{2} r^2 \sin \frac{2\pi}{5}$. Получаем отношение $\frac{2\pi}{5} : \sin \frac{2\pi}{5}$.



К задаче 1.1.44

1.1.44 Любой параллелограмм аффинным преобразованием можно перевести в квадрат. Поскольку при этом сохраняются отношения длин параллельных отрезков, достаточно доказать утверждение задачи в случае, когда $ABCD$ — квадрат. Обозначим через P точку пересечения прямых b и d. Нам достаточно доказать, что $PC \| MK$. Отрезок KL переходит в LM при повороте на $90°$ вокруг центра квадрата $ABCD$, поэтому прямые b и d, которые соответственно параллельны этим отрезкам, перпендикулярны; значит, P лежит на окружности, описанной вокруг $ABCD$. Тогда $\angle CPD = \angle CBD = 45°$, следовательно, угол между прямыми CP и b равен $45°$, но угол между прямыми MK и KL тоже равен $45°$, и $b \| KL$, следовательно, $CP \| MK$.

1.1.45 Параллелограмм, очевидно, можно перевести в прямоугольник. Рассмотрим четырехугольник $ABCD$, у которого нет параллельных сторон. Для определенности будем считать, что пересекаются лучи AB и DC, BC и AD. Пусть $AB = a$, $BC = b$, $CD = pa + qb$, $DA = ua + vb$. Тогда $p < 0$,

$q > 0$, $u < 0$, $v < 0$. Рассмотрим аффинное преобразование, которое переводит векторы a и b в ортогональные векторы a' и b', длины которых равны λ и μ . Нам нужно, чтобы обращалось в нуль скалярное произведение $(pa+qb, ua+vb) = pu\lambda^2 + qv\mu^2$. Поскольку $pu > 0$ и $qv < 0$, этого всегда можно добиться выбором чисел λ и μ.

1.1.46 Пусть A_1, B_1, ..., F_1 — середины сторон AB, BC, ..., FA. Равенство диагоналей AD и BE эквивалентно тому, что прямая A_1D_1 перпендикулярна прямым AB и DE. Пусть O — точка пересечения прямых A_1D_1 и B_1E_1. Нужно построить аффинное преобразование, которое переводит углы A_1 и B_1 четырехугольника A_1BB_1O в прямые углы. Для этого можно воспользоваться результатом задачи 1.1.45. То, что точки пересечения продолжений сторон четырехугольника A_1BB_1O расположены именно так, как нужно, следует из выпуклости шестиугольника.

1.1.47 Рассмотрим систему координат (x,t), где x — координата точки на траектории пули, а t — время. Траектория пули на этой плоскости есть прямая, и лопасть вентилятора попадает на нее, если отрезок $x = a_i$, $t \in [t_i + n/50, t_i + (n+0,5)/50]$ пересекается с ней. Итак, достаточно доказать, что каковы бы ни были t_1, t_2, t_3, t_4 (они соответствуют углам поворота дисков друг относительно друга), существует прямая, пересекающая какие-то 4 из рассматриваемых отрезков. Рассмотрим аффинное преобразование $(x,t) \mapsto (x, t+ax+b)$. Если после такого преобразования нашлась прямая, пересекающая 4 отрезка, то она существовала и до него. Поэтому с помощью преобразования такого вида (а именно, при $a = (t_1-t_3)/(a_3-a_1)$, $b = (a_3t_1 - a_1t_3)/(a_3 - a_1)$) мы можем свести задачу к случаю, когда $t_1 = t_3 = 0$. Пусть теперь s — сумма длин частей наших отрезков, попавших в полосу $0 \leqslant t \leqslant 0,5/50$, и лежащих на прямых $x = a_2$ и $x = a_4$. Если $s \geqslant 0,25$, то очевидно, что найдется прямая, параллельная оси x , пересекающая 4 отрезка. Случай $s < 0,25$ с помощью преобразования $(x,t) \to (x, t + 0,5\cdot(x-a_3)/(a_2-a_3))$ сводится к случаю $s > 0,25$.

1.1.48 а) Прежде всего заметим, что преобразование L взаимно однозначно отображает любую прямую на некоторую прямую. Действительно, пусть A_1 и B_1 — образы двух различных точек A и B. Тогда образ любой точки прямой AB лежит на прямой A_1B_1. Остается доказать, что если C_1 — точка прямой A_1B_1, то ее прообраз C лежит на прямой AB. Предположим, что точка C не лежит на прямой AB. Тогда прямые AC и BC различны, а их образы лежат на прямой A_1B_1. Пусть X — произвольная точка плоскости. Проведем через X прямую, пересекающую прямые AC и BC в различных точках A' и B'. Образы точек A' и B' лежат на прямой A_1B_1, поэтому образ точки X тоже лежит на прямой A_1B_1. Это противоречит тому, что образом отображения L служит вся плоскость. Итак, пусть L — взаимно однозначное отображение плоскости в себя, переводящее любую прямую в некоторую прямую. Будем последовательно доказывать свойства этого отображения, используя каждый раз то, что было доказано на предыдущих шагах:

a) Отображение L переводит параллельные прямые в параллельные прямые.

b) Корректно определено действие L на векторах, т.е. если $\overline{AB} = \overline{CD}$, то $\overline{A_1B_1} = \overline{C_1D_1}$, где A_1, B_1, C_1, D_1 — образы точек A, B, C, D.

c) $L(0) = 0$.
d) $L(a + b) = L(a) + L(b)$.
e) $L(ka) = kL(a)$ при рациональном k.
f) $L(\lambda a) = \lambda L(a)$ при действительных λ.

b) Воспользуемся тем, что через три различные точки можно провести окружность, тогда и только тогда, когда они не лежат на одной прямой. Тогда, по предыдущему пункту это преобразование — аффинное.

1.1.49 Из Т. Козлова об аффинной эквивалентности вытекает, что такое возможно только в случае, если изображение — не плоское, т.е. точки расположены на двух параллельных прямых. Пусть точки $a'_1, \ldots, a'_l$ изображения A' лежат на первой прямой, а точки $a'_{l+1}, \ldots, a'_n$ — на второй, параллельной ей. Из того, что сооответстующие точки второго изображения $a''_1, \ldots, a''_l$ также лежат на одной прямой следует, что либо точки $a_1, \ldots, a_l$ восстановленного тела A_0 лежат на одной прямой, либо оба направления проектирования параллельны плоскости, в которой они лежат. Пусть выполнено второе, тогда выберем подизображение $P(a'_1, \ldots, a'_l, a'_{l+1})$. Оно a-эквивалентно подизображению $Q(a''_1, \ldots, a''_l, a''_{l+1})$, а значит точки $a_1, \ldots, a_l, a_{l+1}$ тела A_0 лежат в одной плоскости, не параллельной направлениям проектирования. Но тогда точки $a_1, \ldots, a_l$ лежат в пересечении двух различных плоскостей, то есть на одной прямой. Аналогично проводится рассуждение для точек $a_{l+1}, \ldots, a_n$.

1.2. Вероятностный подход к распознаванию

Рассматривается следующая задача распознавания образов. Дано множество M объектов, относительно которых производится распознавание. Без существенного ограничения общности будем считать, что оно представляется в виде объединения двух классов $M = K_0 \cup K_1$. Классы K_0 и K_1 неизвестны, но известны вероятности появления объектов этих классов P_0 и P_1, соответственно. Также известно распределение некоторого признака $x \in A$, характеризующего объекты множества X.

Байесовское решающее правило.

Пусть известно значение признака x_0 и требуется отнести объект к одному из двух классов. При этом хотелось бы минимизировать вероятность ошибки, т.е. неправильной классификации объекта.

Будем называть функцией правдоподобия $L_i(x) = P_i \cdot p_i(x)$, где $i = 0, 1$, а $p_i(x)$ — в случае непрерывного распределения — плотность распределения в точке x, а в случае дискретного распределения — вероятность точки x. Тогда при $L_0(x) \geqslant L_1(x)$ относим объект к множеству K_0, а при $L_0 < L_1$ — к K_1.

Возможна более сложная задача. Допустим, что существует штраф S_0 за ошибку первого рода (отнести объект $k \in K_0$ к классу K_1), и штраф S_1 за ошибку второго рода (отнести $k \in K_1$ к K_0). Требуется выбрать решающее правило, минимизирующее средний штраф.

Теорема 2. *Средний штраф минимален при использовании следующего правила $\mathcal{P}$: при $S_0 \cdot L_0(x) \geqslant S_1 \cdot L_1(x)$ относим объект к множеству K_0, иначе — к K_1.*

Доказательство. Доказательство будет проведено для случая непрерывного распределения, для дискретного доказательство проводится аналогично.

Рассмотрим решающее правило $\mathcal{P}^*$, которое отличается от правила $\mathcal{P}$. Обозначим A_0 — множество значений атрибутов, на которых правило $\mathcal{P}$ классифицирует объекты как K_0, а A_1 — как K_1. Аналогично, A_0^* — множество значений атрибутов, на которых правило $\mathcal{P}^*$ классифицирует объекты как K_0, а A_1^* — как K_1. Пусть $\overline{S(\mathcal{P})}$ и $\overline{S(\mathcal{P}^*)}$ — средние штрафы в случае применения соответствующих правил.

Тогда $\overline{S(\mathcal{P})} = S_0P_0 \int\limits_{A_1} p_0(x)dx + S_1P_1 \int\limits_{A_0} p_1(x)dx = \int\limits_{A_1\cap A_1^*} S_0P_0p_0(x)dx + {}$
$+ \int\limits_{A_1\cap A_0^*} S_0P_0p_0(x)dx + \int\limits_{A_0\cap A_0^*} S_1P_1p_1(x)dx + \int\limits_{A_0\cap A_1^*} S_1P_1p_1(x)dx.$

Заметим, что при $x \in A_1$ выполнено неравенство $S_0P_0p_0(x) \leqslant S_1P_1p_1(x)$ (это следует из определения правила $\mathcal{P}$, следовательно $\int\limits_{A_1\cap A_0^*} S_0P_0p_0(x)dx \leqslant$
$\leqslant \int\limits_{A_1\cap A_0^*} S_1P_1p_1(x)dx$. Аналогично $\int\limits_{A_0\cap A_1^*} S_1P_1p_1(x)dx \leqslant \int\limits_{A_0\cap A_1^*} S_0P_0p_0(x)dx.$

Следовательно, $\overline{S(\mathcal{P})} \leqslant \int\limits_{A_1\cap A_1^*} S_0P_0p_0(x)dx + \int\limits_{A_1\cap A_0^*} S_1P_1p_1(x)dx + {}$
$+ \int\limits_{A_0\cap A_0^*} S_1P_1p_1(x)dx + \int\limits_{A_0\cap A_1^*} S_0P_0p_0(x)dx = \int\limits_{A_1^*} S_0P_0p_0(x)dx + \int\limits_{A_0^*} S_1P_1p_1(x)dx =$
$= \overline{S(\mathcal{P}^*)}$, что и требовалось доказать.

Упражнения.

Задача 1.2.1. Объекты классов K_1 и K_2 появляются с частотами $P_1 = \frac{1}{4}$ и $P_2 = \frac{3}{4}$, соответственно. Их атрибуты распределены с плотностями:
$p_1(x) = \begin{cases} \frac{2}{9} \cdot (3 - x), & \text{при } x \in [0,3] \\ 0, & \text{иначе} \end{cases}$ и $p_2(x) = \begin{cases} \frac{1}{8} \cdot (x - 1), & \text{при } x \in [1,5] \\ 0, & \text{иначе} \end{cases}$.
Штрафы за ошибки I и II рода равны $S_1 = 27$ и $S_2 = 16$, соответственно. Найдите решающее правило и средний штраф для него.

Задача 1.2.2. Объекты классов K_1 и K_2 появляются с частотами $P_1 = \frac{2}{3}$ и $P_2 = \frac{1}{3}$, соответственно. Их атрибуты распределены с плотностями:
$p_1(x) = \begin{cases} \frac{1}{8} \cdot (4 - x), & \text{при } x \in [0,4] \\ 0, & \text{иначе} \end{cases}$ и $p_2(x) = \begin{cases} \frac{1}{8} \cdot (x - 2), & \text{при } x \in [2,6] \\ 0, & \text{иначе} \end{cases}$.
Штрафы за ошибки I и II рода равны $S_1 = 3$ и $S_2 = 2$, соответственно. Найдите решающее правило и средний штраф для него.

Задача 1.2.3. Объекты классов K_1 и K_2 появляются с частотами $P_1 = \frac{2}{3}$ и $P_2 = \frac{1}{3}$, соответственно. Их атрибуты распределены с плотностями:
$p_1(x) = \begin{cases} 1 - |x - 5|, & \text{при } x \in [4,6] \\ 0, & \text{иначе} \end{cases}$ и $p_2(x) = \begin{cases} \frac{2}{9} \cdot (5 - x), & \text{при } x \in [2,5] \\ 0, & \text{иначе} \end{cases}$.
Штрафы за ошибки I и II рода равны $S_1 = 1$ и $S_2 = 9$, соответственно. Найдите решающее правило и средний штраф для него.

Задача 1.2.4. Объекты классов K_1 и K_2 появляются с частотами $P_1 = \frac{2}{3}$ и $P_2 = \frac{1}{3}$, соответственно. Их атрибуты распределены с плотностями: $p_1(x) = = \begin{cases} \frac{1}{2} - \frac{1}{4} \cdot |x|, & \text{при } x \in [-2, 2] \\ 0, & \text{иначе} \end{cases}$ и $p_2(x) = \begin{cases} 1 - |x-2|, & \text{при } x \in [1, 3] \\ 0, & \text{иначе} \end{cases}$. Штрафы за ошибки I и II рода равны $S_1 = 2$ и $S_2 = 1$, соответственно. Найдите решающее правило и средний штраф для него.

Задача 1.2.5. Объекты классов K_0 и K_1 появляются с частотами $P_0 = \frac{1}{5}$ и $P_1 = \frac{4}{5}$, сответственно. Их атрибуты распределены с плотностями: $p_0(x) = = \begin{cases} \frac{1}{x^2}, & \text{при } x \geqslant 1 \\ 0, & \text{иначе} \end{cases}$ и $p_1(x) = \begin{cases} \frac{3}{(6-x)^4}, & \text{при } x \leqslant 5 \\ 0, & \text{иначе} \end{cases}$. Штрафы за ошибки I и II рода равны $S_0 = 3$ и $S_1 = 16$, соответственно. Найдите решающее правило и средний штраф для него.

Задача 1.2.6. Объекты классов K_1 и K_2 появляются с частотами $P_1 = \frac{3}{5}$ и $P_2 = \frac{2}{5}$, сответственно. Их атрибуты распределены с плотностями: $p_1(x) = \begin{cases} \frac{2}{(x+1)^2}, & \text{при } x \geqslant 1 \\ 0, & \text{иначе} \end{cases}$ и $p_2(x) = \begin{cases} \frac{1}{(4-x)^2}, & \text{при } x \leqslant 3 \\ 0, & \text{иначе} \end{cases}$. Штрафы за ошибки I и II рода равны $S_1 = 1$ и $S_2 = 3$, соответственно. Найдите решающее правило и средний штраф для него.

Задача 1.2.7. Объекты классов K_1 и K_2 появляются с частотами $P_1 = \frac{3}{5}$ и $P_2 = \frac{2}{5}$, соответственно. Их атрибуты распределены с плотностями: $p_1(x) = = \begin{cases} \frac{2}{x^3}, & \text{при } x \geqslant 1 \\ 0, & \text{иначе} \end{cases}$ и $p_2(x) = \begin{cases} \frac{2}{(5-x)^3}, & \text{при } x \leqslant 4 \\ 0, & \text{иначе} \end{cases}$. Штрафы за ошибки I и II рода равны $S_1 = 9$ и $S_2 = 4$, соответственно. Найдите решающее правило и средний штраф для него.

Задача 1.2.8. Объекты классов K_1 и K_2 появляются с частотами $P_1 = \frac{1}{5}$ и $P_2 = \frac{4}{5}$, соответственно. Их атрибуты распределены с плотностями: $p_1(x) = \begin{cases} \frac{8}{(x+1)^3}, & \text{при } x \geqslant 1 \\ 0, & \text{иначе} \end{cases}$ и $p_2(x) = \begin{cases} \frac{5}{(5-x)^6}, & \text{при } x \leqslant 4 \\ 0, & \text{иначе} \end{cases}$. Штрафы за ошибки I и II рода равны $S_1 = 5$ и $S_2 = 2$, соответственно. Найдите решающее правило и средний штраф для него.

Задача 1.2.9. Объекты классов K_1 и K_2 появляются с частотами $P_1 = \frac{5}{6}$ и $P_2 = \frac{1}{6}$, соответственно. Их атрибуты распределены с плотностями: $p_1(x) = \begin{cases} \frac{2}{(x+2)^2}, & \text{при } x \geqslant 0 \\ 0, & \text{иначе} \end{cases}$ и $p_2(x) = \begin{cases} \frac{1}{(6-x)^2}, & \text{при } x \leqslant 5 \\ 0, & \text{иначе} \end{cases}$. Штрафы за ошибки I и II рода равны $S_1 = 5$ и $S_2 = 18$, соответственно. Найдите решающее правило и средний штраф для него.

Задача 1.2.10. Объекты классов K_1 и K_2 появляются с частотами $P_1 = \frac{1}{4}$ и $P_2 = \frac{3}{4}$, соответственно. Их атрибуты распределены с плотностями: $p_1(x) = \begin{cases} \frac{1}{(x+1)^2}, & \text{при } x \geqslant 0 \\ 0, & \text{иначе} \end{cases}$ и $p_2(x) = \begin{cases} \frac{1}{(5-x)^2}, & \text{при } x \leqslant 4 \\ 0, & \text{иначе} \end{cases}$. Штрафы за ошибки I и II рода равны $S_1 = 12$ и $S_2 = 1$, соответственно. Найдите решающее правило и средний штраф для него.

Задача 1.2.11. Вероятности появления объектов из K_1 и K_2 равны $P_1 = \frac{1}{5}$, $P_2 = \frac{4}{5}$, соответственно; их атрибуты распределены с плотностями: $p_1(x) = \begin{cases} \frac{2}{9} \cdot (3 - x), & \text{при } x \in [0, 3] \\ 0, & \text{иначе} \end{cases}$ и $p_2(x) = \begin{cases} \frac{1}{8} \cdot (x - 1), & \text{при } x \in [1, 5] \\ 0, & \text{иначе} \end{cases}$. Штрафы за ошибки I и II рода равны $S_1 = 3$ и $S_2 = 4$, соответственно. Найти решающее правило и средний штраф для него.

Задача 1.2.12. Вероятности появления объектов из K_1 и K_2 равны $P_1 = \frac{1}{4}$, $P_2 = \frac{3}{4}$, соответственно; Атрибуты распределены с плотностями: $p_1(x) = \begin{cases} \frac{1}{4} - \frac{|x-1|}{16}, & |x - 1| \leqslant 4; \\ 0, & \text{иначе} \end{cases}$ и $p_2(x) = \begin{cases} \frac{1}{6} - \frac{|x-5|}{36}, & |x - 5| \leqslant 6; \\ 0, & \text{иначе} \end{cases}$ Штрафы за ошибки 1 и 2 рода: $S_1 = 8$ и $S_2 = 6$, соответственно Найти решающее правило и средний штраф для него.

Задача 1.2.13. Вероятности появления объектов 1 и 2 классов равны $P_1 = P_2 = \frac{1}{2}$, их атрибуты распределены с плотностями: $p_1(x) = \begin{cases} \frac{1}{5} - \frac{|x-2|}{25}, & |x - 2| \leqslant 5; \\ 0, & \text{иначе} \end{cases}$ и $p_2(x) = \begin{cases} \frac{1}{4} - \frac{|x-5|}{16}, & |x - 5| \leqslant 4; \\ 0, & \text{иначе} \end{cases}$ Штрафы за ошибки 1 и 2 рода: $S_1 = \frac{5}{2}$ и $S_2 = 8$, соответственно. Найти решающее правило и средний штраф для него.

Задача 1.2.14. Вероятности появления объектов 1 и 2 классов равны $P_1 = P_2 = 1/2$. Их атрибуты распределены с плотностями: $p_2(x) = \begin{cases} \frac{1}{5} - \frac{|x-3|}{25}, & |x - 3| \leqslant 5; \\ 0, & \text{иначе} \end{cases}$ и $p_1(x) = \begin{cases} \frac{x-4}{18}, & 4 \leqslant x \leqslant 10; \\ 0, & \text{иначе} \end{cases}$ Штрафы за ошибки 1 и 2 рода: $S_1 = 20$ и $S_2 = 24$, соответственно. Найти решающее правило и средний штраф для него.

Задача 1.2.15. Вероятности появления объектов 1 и 2 классов равны $P_1 = 3/5$ и $P_2 = 2/5$. Их атрибуты распределены с плотностями: $p_1(x) = \begin{cases} \frac{2}{5} - 2 \cdot \frac{x-1}{25}, & 1 \leqslant x \leqslant 6; \\ 0, & \text{иначе} \end{cases}$ и $p_2(x) = \begin{cases} \frac{1}{10} - \frac{|x-6|}{100}, & |x - 6| \leqslant 10; \\ 0, & \text{иначе} \end{cases}$ Штрафы за ошибки 1 и 2 рода: $S_1 = 35/6$ и $S_2 = 30$, соответственно. Найти решающее правило и средний штраф для него.

Задача 1.2.16. Вероятности появления объектов 1 и 2 классов равны $P_1 = \frac{8}{9}$ и $P_2 = \frac{1}{9}$. Их атрибуты распределены с плотностями: $p_1(x) = \begin{cases} \frac{1}{2} - \frac{x-1}{8}, & x \in [1; 5]; \\ 0, & \text{иначе} \end{cases}$ и $p_2(x) = \begin{cases} \frac{1}{5} - \cdot \frac{6-x}{50}, & x \in [-4; 6]; \\ 0, & \text{иначе} \end{cases}$ Штрафы за ошибки 1 и 2 рода: $S_1 = 4$ и $S_2 = 25$, соответственно. Найти решающее правило и средний штраф для него.

Задача 1.2.17. Объекты класса K_1 встречаются с частотой $P_1 = 1/4$ и имеют распределение признака x с плотностью $p_1(x) = \begin{cases} -\frac{1}{2} \cos x, & x \in [\pi/2, 3\pi/2] \\ 0, & x \notin [\pi/2, 3\pi/2] \end{cases}$. Объекты класса K_2 встречаются с частотой $P_2 = 3/4$ и имеют распределение признака x с плотностью

$p_2(x) = \begin{cases} \frac{1}{2}\sin x, & x \in [0,\pi] \\ 0, & x \notin [0,\pi] \end{cases}$. Штрафы ошибки 1 рода (принять K_1 за K_2) $F_1 = 3$ и второго рода $F_2 = 1$. Определите оптимальное решающее правило и найдите средний штраф для него.

Задача 1.2.18. Объекты класса K_1 встречаются с частотой $P_1 = \frac{1}{3}$ и имеют распределение признака x с плотностью $p_1(x) = \begin{cases} \frac{1}{2}\sin x, & x \in [0,\pi] \\ 0, & x \notin [0,\pi] \end{cases}$. Объекты класса K_2 встречаются с частотой $P_2 = \frac{2}{3}$ и имеют распределение признака x с плотностью $p_2(x) = \begin{cases} -\frac{1}{2}\cos x, & x \in [\pi/2, 3\pi/2] \\ 0, & x \notin [\pi/2, 3\pi/2] \end{cases}$. Штрафы ошибки 1 рода (принять K_1 за K_2) $F_1 = 2$ и второго рода $F_2 = 1$. Определите оптимальное решающее правило и найдите средний штраф для него.

Задача 1.2.19. Объекты класса K_1 встречаются с частотой $P_1 = \frac{1}{3}$ и имеют распределение признака x с плотностью $p_1(x) = \begin{cases} \frac{\pi}{12}\sin\frac{\pi x}{6}, & x \in [0,6] \\ 0, & \text{иначе} \end{cases}$. Объекты класса K_2 встречаются с частотой $P_2 = \frac{2}{3}$ и имеют распределение признака x с плотностью $p_2(x) = \begin{cases} \frac{\pi}{12}\cos\frac{\pi(x-5)}{6}, & x \in [2,8] \\ 0, & \text{иначе} \end{cases}$. Штрафы ошибки 1 рода (принять K_1 за K_2) $F_1 = 6$ и второго рода $F_2 = 3$. Определите оптимальное решающее правило и найдите средний штраф для него.

Задача 1.2.20. Элементы классов K_1 и K_2 имеют вероятности появления $P_1 = \frac{3}{4}$, $P_2 = \frac{1}{4}$ и распределены случайно с плотностями: $p_1 = \frac{1}{\sqrt{8\pi}}\mathrm{e}^{-x^2/8}$ и $p_2 = \frac{1}{\sqrt{8\pi}}\mathrm{e}^{-(x-4)^2/8}$. Штраф ошибки 1–го рода (принять K_1 за K_2) равен 2, штраф ошибки 2–го рода равен 6.

a) Найти решающее правило, минимизирующее средний штраф и классифицировать по этому правилу точки 1, $\frac{5}{2}$ и 3.

b) Выразить средний штраф через Q-функцию $Q(x) = \int\limits_x^\infty e^{-t^2/2}dt$.

Задача 1.2.21. Объекты классов K_1 и K_2 распределены экспоненциально с плотностями $p_1(x) = \begin{cases} 3e^{-3x}, & x \geqslant 0; \\ 0, & x < 0 \end{cases}$ и $p_2(x) = \begin{cases} 5e^{-5(4-x)}, & x \leqslant 4; \\ 0, & x > 4 \end{cases}$, и имеют вероятности появления $P_1 = \frac{1}{4}$ и $P_2 = \frac{3}{4}$, соответственно. Штраф ошибки 1–го рода (принять K_1 за K_2) равен 5, а 2–го рода — 1. а) Сформулировать решающее правило и отнести к соответствующему классу точки 0,1,2,3,4. б) Вычислить математическое ожидание штрафа для полученного решающего правила.

Задача 1.2.22. Объекты классов K_1 и K_2 распределены с плотностями $p_1(x) = \begin{cases} 2x^{-3}, & x \geqslant 1; \\ 0, & x < 1 \end{cases}$ и $p_2(x) = \begin{cases} (5-x)^{-2}, & x \leqslant 4; \\ 0, & x > 4 \end{cases}$ (распределение Парето), и имеют вероятности появления $0,6$ и $0,4$, соответственно. Штраф ошибки 1–го рода (принять K_1 за K_2) равен 9, а 2–го рода — 4. а) Сформулировать решающее правило и отнести к соответствующему классу точки

1,2,3,4,5. б) Вычислить математическое ожидание штрафа для полученного решающего правила.

Задача 1.2.23. Доказать, что если у двух $n-$мерных нормальных распределений априорные вероятности равны $1/2$ и совпадают матрицы ковариаций (и отличаются вектора средних), то байесовская граница есть некоторая гиперплоскость в $\mathbb{R}^n$.

Задача 1.2.24. Известно, что фальшивые монеты встречаются с частотой P. Фальшивая монета отличается от настоящей тем, что у нее вероятность выпадания орла отличается от $1/2$ и равна $q_0 > \frac{1}{2}$. Надо определить, настоящая ли монета, сделав N подбрасываний. Найдите критерий и определите вероятность ошибки для него.

Задача 1.2.25. При клинических испытании нового лекарства улучшение наступило у n больных (из общей выборки величины N). Возможно, это было связано с «эффектом плацебо», который действует с вероятностью q_0. Возможно, лекарство является действенным и помогает с вероятностью $q_1 > q_0$. Известны штрафы: за ошибку I рода S_1 — стоимость производства бесполезного лекарства[4]; за ошибку второго рода S_2 — недополученная прибыль в случае отклонения действующего лекарства. Считая оба случая (действующего и не действующего лекарства) априори равновероятными (т.е. $P_1 = P_2$), определить алгоритм минимизации штрафа.

Задача 1.2.26. Для передачи по аналоговому каналу с шумом используется фазовая модуляция сигнала. Символ 0 кодируется двумя последовательными сигналами с фазами $x_1 = 1$ и $x_2 = -1$; 1 — сигналами с фазами $x_1 = -1$ и $x_2 = 1$. При передаче к фазе каждого из сигналов добавляется Гауссов шум (ξ_1, ξ_2), где $\xi_{1,2}$ независимы и нормально распределены $N(0, \sigma^2)$. Т.е. приемник фиксирует искаженные сигналы: $y_1 = x_1 + \xi_1$ и $y_2 = x_2 + \xi_2$. Построить решающее правило, минимизирующее вероятность ошибки и найти эту вероятность. Считаем, что символы 1 и 0 равновероятны ($P_0 = P_1 = \frac{1}{2}$).

Ответы и указания.

1.2.1 Относим к K_1 при $x \in [0; 2]$ и к K_2 при $x \in [2; 5]$. Средний штраф $\bar{S} = \frac{3}{4} + \frac{3}{4} = \frac{3}{2}$.

1.2.2 Граница: $T = \frac{7}{2}$, Средний штраф $\bar{S} = \frac{3}{32} + \frac{1}{32} = \frac{1}{8}$.

1.2.3 Классифицируем как K_1 при $x \in [\frac{9}{2}, 6]$ и как K_2 при $x \in [2, \frac{9}{2}]$, Средний штраф $\bar{S} = \frac{1}{12} + \frac{1}{12} = \frac{1}{6}$.

1.2.4 Относим к K_1 при $x \in [-2, \frac{3}{2}]$, к K_2 при $x \in [\frac{3}{2}, 3]$. Средний штраф $\bar{S} = \frac{1}{24} + \frac{1}{24} = \frac{1}{12}$.

1.2.5 Относим к K_1 при $x \in [1, 2]$, к K_2 при $x \in (2, 5]$. Средний штраф $\bar{S} = \frac{61}{625} + \frac{9}{50} = \frac{347}{1250}$.

[4]В задаче представлена идеальная ситуация, когда бесполезные лекарства не пользуются спросом.

1.2.6 При $x \in [1; \frac{3}{2}] \cup (3; +\infty)$ классифицируем как K_1, иначе — K_2. Средний штраф $\bar{S} = \frac{2}{25} + \frac{9}{50} = \frac{13}{50} = 0.26$.

1.2.7 При $x \in [1; 3]$ классифицируем как K_1, иначе — K_2. Средний штраф $\bar{S} = \frac{3}{10} + \frac{21}{80} = \frac{9}{16}$.

1.2.8 Граница : $T = 3$, Средний штраф $\bar{S} = \frac{31}{640} + \frac{9}{100} = \frac{443}{3200}$.

1.2.9 Граница : $T = 3$, Средний штраф $\bar{S} = \frac{1}{2} + \frac{10}{21} = \frac{41}{42}$.

1.2.10 Граница : $T = 3$, Средний штраф $\bar{S} = \frac{9}{40} + \frac{3}{20} = \frac{3}{8}$.

1.2.11 Решающее правило: относим к K_1 при $x \in [0; 1.5]$, к K_2 при $x \in$ $\in [1.5; 5]$. Средний штраф $\bar{S} = \frac{1}{5}$.

1.2.12 Решающее правило: относим к K_1 при $x \in [-3; 2]$, к K_2 при $x \in$ $\in [2; 11]$. Средний штраф $\bar{S} = \frac{9}{8}$.

1.2.13 Решающее правило: относим к K_1 при $x \in [-3; 2]$, к K_2 при $x \in [2; 9]$. Средний штраф $\bar{S} = 1$.

1.2.14 Граница : $c = 5.5$, Средний штраф 2.

1.2.15 Если $x \in [-4; 3] \cup [6; 16]$, то относим к K_1, иначе — к K_2, Средний штраф $\frac{21}{25}$.

1.2.16 Граница : $T = 4$, Средний штраф $\frac{7}{4}$.

1.2.17 Граница: $T = \frac{3\pi}{4}$, средний штраф $\bar{S} = \frac{6-3\sqrt{2}}{8}$.

1.2.18 При $x \leqslant \frac{3\pi}{4}$ относим к K_1, иначе — к K_2. Средний штраф $\bar{S} = \frac{2-\sqrt{2}}{3}$.

1.2.19 Граница: $T = 4$, средний штраф равен 1.

1.2.20 a) Если $x < 2$, то относим к K_1, иначе — к K_2. b) $\bar{S} = 3Q(1)$.

1.2.21 Если $x \in [0, \frac{5}{2}] \cup (4, +\infty)$, то относим к первому классу, иначе — ко второму. Средний штраф $\bar{S} = 2e^{-15/2} - \frac{3}{4}e^{-20} - \frac{5}{4}e^{-12}$.

1.2.22 Если $x \in [1, 3] \cup (4, +\infty)$, то относим к первому классу, иначе — ко второму. Средний штраф $\bar{S} = \frac{9}{16}$.

1.2.23 Пусть K — матрица ковариаций, μ_1 и μ_2 — вектора средних. Тогда байесовская граница задается условием

$$\frac{1}{2\pi\sqrt{\det K}} \exp((x-\mu_1)K^{-1}(x-\mu_1)^T) = \frac{1}{2\pi\sqrt{\det K}} \exp((x-\mu_2)K^{-1}(x-\mu_2)^T),$$

т.е. $(x - \mu_1)K^{-1}(x - \mu_1)^T = (x - \mu_2)K^{-1}(x - \mu_2)^T$. Преобразовав правую часть как

$$\begin{aligned}(x - \mu_2)K^{-1}(x - \mu_2)^T &= (x - \mu_1 + \Delta\mu)K^{-1}(x - \mu_1 + \Delta\mu)^T = \\ &= (x - \mu_1)K^{-1}(x - \mu_1)^T + 2\Delta\mu \cdot K^{-1}(x - \mu_1)^T + \Delta\mu K^{-1}\Delta\mu^T,\end{aligned}$$

где $\Delta\mu = \mu_1 - \mu_2$. Тогда байесовская граница может быть записана как

$$\Delta\mu K^{-1}(x - \mu_1)^T + 2\Delta\mu \cdot K^{-1}\Delta\mu^T = 0,$$

а это — линейное уравнение, следовательно, задает некоторую гиперплоскость.

1.2.24 Количество выпавших орлов распределено по Бернулли, т.е. вероятность выпадения n орлов равна $C_N^n(1/2)^N$ в случае настоящей монеты и $C_N^n \cdot (q_0)^n \cdot (1-q_0)^{N-n}$ для фальшивой. По правилу Байеса монета идентифицируется как фальшивая, если $P \cdot C_N^n q_0^n (1-q_0)^{N-n} \geqslant (1-P) \cdot C_N^n (1/2)^N$. Прологарифмировав, получим $n \geqslant n_0$, где $n_0 = \left\lceil -\frac{\ln((1-P)/P)+N\ln 2(1-q_0)}{\ln((1-q_0)/q_0)} \right\rceil$. Вероятность неправильной классификации равна $P_{err} = P \sum\limits_{n=0}^{n_0-1} C_N^n q_0^n (1 - q_0)^{N-n} + (1-P) \sum\limits_{n=n_0}^{N} C_N^n (1/2)^N$.

1.2.25 Аналогично задаче 1.2.24

1.2.26 Будем относить к K_0 в случае, когда $p_0 \geqslant p_1$, т.е.

$$\frac{1}{2\pi\sigma^2}\exp\left(\frac{-(y_1-1)^2-(y_2+1)^2)}{2\sigma^2}\right) \geqslant \frac{1}{2\pi\sigma^2}\exp\left(\frac{-(y_1+1)^2-(y_2-1)^2)}{2\sigma^2}\right).$$

После упрощений получим $y_1 - y_2 \geqslant 0$. Таким образом, выбираем символ «0» при $y_1 \geqslant y_2$ и символ «1» — иначе. Вероятность ошибки для этого решающего правила равна

$$P_E = I_0 + I_1 = \frac{1}{2\pi\sigma^2}\int\limits_{y_1<y_2}\exp\left(\frac{-(y_1-1)^2-(y_2+1)^2)}{2\sigma^2}\right)dy_1dy_2+$$
$$+\frac{1}{2\pi\sigma^2}\int\limits_{y_1\geqslant y_2}\exp\left(\frac{-(y_1+1)^2-(y_2-1)^2)}{2\sigma^2}\right)dy_1dy_2.$$

Для подсчета интеграла $I_0 = \frac{1}{2\pi\sigma^2}\int\limits_{y_1<y_2}\exp\left(\frac{-(y_1-1)^2-(y_2+1)^2)}{2\sigma^2}\right)dy_1dy_2$ рассмотрим двумерное нормальное распределение со средними $(1,-1)$ и матрицей ковариаций $\sigma^2 \cdot I$. Интеграл I_0 равен вероятности попадания в полуплоскость $y_1 < y_2$. После сдвига на вектор $(-1,1)$ и поворота на $\pi/4$ получится нормальное распределение с нулевым средним и той же матрицей $\sigma^2 I$, при этом полуплоскость будет $y_1' < -\sqrt{2}$. Вероятность этого события легко найти, она равна $Q(\frac{\sqrt{2}}{\sigma})$. Второй интеграл I_1 равен той же величине, в итоге получаем $P_E = Q(\frac{\sqrt{2}}{\sigma})$.

1.3. Тестовый подход к распознаванию

Основные обозначения и определения.

Тестовый подход связан с понятие теста, предложенного С.В.Яблонским и И.А.Чегис [17, 18].

Пусть имеется множество объектов $M = \{x_1, \ldots, x_m\}$, и каждый из объектов описывается в заданной системе признаков наборами значений этих признаков, или, иными словами, задается вектором признакового пространства. Тогда множество M может быть описано матрицей

$$T_{m,n} = \begin{pmatrix} x_1(1) & x_1(2) & \ldots & x_1(n) \\ x_2(1) & x_2(2) & \ldots & x_2(n) \\ . & . & \ldots & . \\ x_m(1) & x_m(2) & \ldots & x_m(n) \end{pmatrix},$$

где через $x_i(j)$ обозначено значение j-го признака объекта x_i, $i = 1, 2, \ldots, m$, $j = 1, 2, \ldots, n$. Пусть M разбито на два класса K_1 и K_2, а тем самым и матрица $T_{m,n}$ состоит из двух подматриц T_{K_1} и T_{K_2}. Классы K_1 и K_2 нам неизвестны, но нам дана некоторая обучающая выборка. Пусть T_1 и T_2 — подматрицы матриц T_{K_1} и T_{K_2}, соответствующие элементам обучающей выборки. Требуется для любого набора признаков, который есть описание объекта из M определить, к какому классу он относится.

Будем считать, что значения признаков $1, 2, \ldots, n$ принадлежат множеству $\{0, 1\}$, и эти признаки выбраны так, что в матрице $T_{m,n}$ нет двух одинаковых строк, отвечающих объектам из разных классов.

Если T — некоторая матрица, а τ — некоторый набор признаков (набор столбцов), то через $T(\tau)$ обозначим подматрицу матрицы T, полученную удалением всех столбцов кроме столбцов из τ.

Набор столбцов τ назовем *тестом*, если не существует строки, содержащейся в $T_1(\tau)$ и $T_2(\tau)$ одновременно. Тем самым, тест — это множество признаков, которое позволяет отличить на обучающей выборке классы K_1 и K_2.

Будем говорить, что τ — *тупиковый тест*, если τ — тест, а любой его поднабор не является тестом. Иными словами, тупиковый тест — минимальное по вложению множество признаков, которое все еще может отличать классы K_1 и K_2 на обучающей выборке.

Универсальный алгоритм построения тестов.

В [17] приводится описание следующего алгоритма, позволяющего находить все тупиковые тесты:

Пусть заданы таблицы признаков:

$$T_1 = \begin{pmatrix} a_{11} & a_{12} & \ldots & a_{1n} \\ a_{21} & a_{22} & \ldots & a_{2n} \\ . & . & \ldots & . \\ a_{m_1 1} & a_{m_1 2} & \ldots & a_{m_1 n} \end{pmatrix},$$

$$T_2 = \begin{pmatrix} b_{11} & b_{12} & \ldots & b_{1n} \\ b_{21} & b_{22} & \ldots & b_{2n} \\ . & . & \ldots & . \\ b_{m_2 1} & b_{m_2 2} & \ldots & b_{m_2 n} \end{pmatrix}$$

Нижеприведенный алгоритм позволяет находить множество всех тупиковых тестов для указанных таблиц.

1. **ШАГ1 — XOR.** Рассмотрим множество

   $$T^{\oplus} = T_1 \bigoplus T_2 = \{(a_{s_1 1} \oplus b_{s_2 1}, ..., a_{s_1 n} \oplus b_{s_2 n}) \,|\, s_1 = 1, ..., m_1, s_2 = 1, ..., m_2\}$$

   — множество попарных XOR-сумм строк первой и второй таблицы (этот набор рассматривается как множество булевых векторов, т.е. повторения отбрасываются).

2. **ШАГ2 — MIN.** Найдем минимальные элементы этого множества

   $$\mathcal{M} = \min(T^{\oplus}) = \left\{ v \in T^{\oplus} \,|\, \nexists\, v' \in T^{\oplus}, v' \leqslant v \right\}.$$

   Т.е. на практике это означает, что для любых сравнимых наборов $v_1 \leqslant$ $\leqslant v_2$ надо вычеркнуть v_2 из списка.

3. **ШАГ3 — КНФ.** Каждому вектору $v = (v_1, \ldots, v_n)$ из $\mathcal{M}$ ставится в соответствие дизъюнкция $D_v = (x_{i_1} \vee \ldots \vee x_{i_k})$, где $i_1, \ldots, i_k$ — номера единичных компонент вектора v. Рассматривается их конъюнкция

   $$K(x_1, \ldots, x_n) = \bigwedge_{v \in \mathcal{M}} D_v.$$

4. **ШАГ4 — ДНФ.** В конъюнктивной форме $K(x_1, \ldots, x_n)$ раскрываются скобки и производятся упрощения в соответствии с правилом $A \vee AB \to A$. Получается дизъюнктивная нормальная форма:

   $$\mathcal{D} = \bigvee_{t \in \mathcal{T}} k_t,$$

   где каждая конъюнкция $k_t = x_{j_1} \ldots x_{j_l}$ соответствует тупиковому тесту $t = (j_1, ..., j_l)$ и $\mathcal{T}$ — множество всех тупиковых тестов.

Проиллюстрируем работу этого алгоритма на простом примере:
Пример 1.6. Найдите все тупиковые тесты для таблиц признаков:

$$\left(\frac{T_1}{T_2}\right) = \left(\begin{array}{c|c|c|c|c} 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 \\ \hline 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{array}\right). \qquad (1.1)$$

Решение Применим указанный алгоритм:

1. **ШАГ1 — XOR.**

   $$T^{\oplus} = T_1 \bigoplus T_2 = \left(\begin{array}{ccccc} (1, & 1, & 0, & 1, & 0) \\ \hline (0, & 1, & 1, & 1, & 0) \\ \hline (1, & 0, & 1, & 1, & 0) \\ \hline (1, & 1, & 0, & 1, & 1) \\ \hline (0, & 1, & 1, & 1, & 1) \\ \hline (1, & 0, & 1, & 1, & 1) \end{array}\right).$$

   — множество попарных XOR-сумм строк первой и второй таблицы.

2. **ШАГ2 — MIN.** Найдем минимальные элементы этого множества — вычеркнем последние три строки.

$$\min(T^{\oplus}) = \begin{pmatrix} (1, & 1, & 0, & 1, & 0) \\ (0, & 1, & 1, & 1, & 0) \\ (1, & 0, & 1, & 1, & 0) \end{pmatrix}.$$

3. **ШАГ3 — КНФ.**

$$K(x) = (x_1 \vee x_2 \vee x_4) \wedge (x_2 \vee x_3 \vee x_4) \wedge (x_1 \vee x_3 \vee x_4).$$

4. **ШАГ4 — ДНФ.** Раскрываем скобки и упрощаем: $(x_1 \vee x_2 \vee x_4) \wedge (x_2 \vee \vee x_3 \vee x_4) \wedge (x_1 \vee x_3 \vee x_4) = (x_1x_3 \vee x_2 \vee x_4) \wedge (x_1 \vee x_3 \vee x_4) = x_1x_2 \vee x_1x_3 \vee \vee x_2x_3 \vee x_4$. Итак $\mathcal{T} = \{(1,2); (1,3); (2,3); (4)\}$ — множество тупиковых тестов.

Этот алгоритм допускает и использование в обратном порядке:

Пример 1.7. Привести пример матриц признаков T_0, T_1, для которых тупиковыми будут следующие тесты (и только они): $[1], [2,3], [2,4], [3,4,5]$.

Решение. Запишем тесты в виде ДНФ: $f(x) = x_1 \vee (x_2 \wedge x_3) \vee (x_2 \wedge x_4) \vee \vee (x_3 \wedge x_4 \wedge x_5)$. Преобразуем к КНФ и упростим:

$$f(x) = (x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee x_2 \vee x_4) \wedge (x_1 \vee x_2 \vee x_5) \wedge (x_1 \vee x_3 \vee x_4) \wedge$$
$$\wedge (x_1 \vee x_2 \vee x_3 \vee x_4) \wedge (x_1 \vee x_2 \vee x_3 \vee x_5) \wedge (x_1 \vee x_2 \vee x_4 \vee x_5) \wedge (x_1 \vee x_3 \vee x_4 \vee x_5) =$$
$$= (x_1 \vee x_2 \vee x_3) \wedge (x_1 \vee x_2 \vee x_4) \wedge (x_1 \vee x_2 \vee x_5) \wedge (x_1 \vee x_3 \vee x_4).$$

В качестве матриц можно взять $T_1 = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 1 & 1 & 0 & 1 & 0 \end{pmatrix}$, $T_2 = \\ = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \end{pmatrix}$.

Алгоритм A_1.

Алгоритм A_1 [5], как и многие другие, основывается на понятии информационного веса признака, предложенного Ю.И. Журавлёвым.

Для пары матриц T_1 и T_2 ***информационным весом*** p_i ***признака*** i, $i \in \{1, 2, \ldots, n\}$, будем называть отношение числа тупиковых тестов, в которые вошёл признак i, к общему числу тупиковых тестов.

Чем больше вес p_i, тем важнее признак i с точки зрения различения классов K_1 и K_2.

Вектор $p = (p_1, \ldots, p_n)$ назовем ***вектором информационных весов признаков*** или ***весовым вектором.***

Например, для матриц (1.1) в предыдущем параграфе были получены тупиковые тесты $\mathcal{T} = \{(1,2); (1,3); (2,3); (4)\}$. Следовательно, вектор информационных весов равен $p = (\frac{1}{2}, \frac{1}{2}, \frac{1}{2}, \frac{1}{4}, 0)$.

Сначала в алгоритме A_1 для каждого признака i вычисляется его информационный вес p_i, $i \in \{1, 2, \ldots, n\}$.

Если

$$T_1 = \begin{pmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ . & . & \dots & . \\ a_{m_1 1} & a_{m_1 2} & \dots & a_{m_1 n} \end{pmatrix},$$

$$T_2 = \begin{pmatrix} b_{11} & b_{12} & \dots & b_{1n} \\ b_{21} & b_{22} & \dots & b_{2n} \\ . & . & \dots & . \\ b_{m_2 1} & b_{m_2 2} & \dots & b_{m_2 n} \end{pmatrix}$$

и $x = (x_1, x_2, \dots, x_n)$ — объект, подлежащий классификации, то вычисляются две величины

$$r_1 = \frac{1}{m_1} \sum_{i=1}^{m_1} \sum_{j=1}^{n} p_j (a_{ij} \oplus x_j) \tag{1.2}$$

и

$$r_2 = \frac{1}{m_2} \sum_{i=1}^{m_2} \sum_{j=1}^{n} p_j (b_{ij} \oplus x_j), \tag{1.3}$$

где операция $\oplus$ — есть сумма по модулю 2, а $\sum$ — обычная сумма.

Теперь, если $r_1 > r_2$, то x следует отнести к классу K_2, и если $r_1 \leqslant r_2$, то x следует отнести к классу K_1. Описанное решающее правило разбивает множество объектов M на две части, разделяемые гиперплоскостью $r_1(x) = r_2(x)$.

Проиллюстрируем этот алгоритм на примере матриц из предыдущего параграфа (1.1) и объекта с признаками $x = (1, 1, 1, 0, 1)$. Вычислим[5] величины: $r_1(x) = \frac{1}{3}\left(\frac{1}{2} \cdot 1 + \frac{1}{2} \cdot 1 + \frac{1}{2} \cdot 1 + \frac{1}{4} \cdot 3\right) = \frac{3}{4}$ и $r_2(x) = \frac{1}{2}\left(\frac{1}{2} \cdot 2 + \frac{1}{2} \cdot 2 + \frac{1}{2} \cdot 2\right) = \frac{3}{2}$. Поскольку $r_2 = \frac{3}{2} > \frac{3}{4} = r_1$, то относим объект к классу K_1.

Алгоритм A_2/A_3.

В алгоритме A_2 [11] решающее правило выглядит следующим образом. Для каждой строки $s = (s_1, \dots, s_n)$ матриц T_1 и T_2 находим скалярное произведение

$$s \cdot p = \sum_{i=1}^{n} s_i \cdot p_i,$$

называемое весом строки.

Пусть q — вес объекта $x = (x_1, \dots, x_n)$, подлежащего классификации. Тогда, если $q \geqslant h$, то отнесем x к классу K_1, а иначе — к классу K_2. Значение h подбирается с учетом требования, чтобы наибольшее число строк матриц T_1 и T_2 классифицировались правильно. Разделяющей поверхностью в алгоритме A_2 является гиперплоскость, заданная уравнением

$$\sum_{i=1}^{n} x_i \cdot p_i = h.$$

[5] В каждой скобке по столбцам суммируется количество несовпадений атрибутов объекта для данного столбца умноженное на информационный вес этого столбца.

Проиллюстрируем работу алгоритма A_2 на том же примере 1.1 (информационные веса строк указаны в правом столбце):

$$\left(\frac{T_1}{T_2}\right) = \left(\begin{array}{ccccc|c} 1 & 1 & 0 & 1 & 0 & 5/4 \\ 0 & 1 & 1 & 1 & 0 & 5/4 \\ 1 & 0 & 1 & 1 & 0 & 5/4 \\ \hline 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{array}\right). \tag{1.4}$$

Выберем $h = \frac{1}{2} \cdot (0 + \frac{5}{4}) = \frac{5}{8}$. Будем классифицировать вектор $x = (1, 1, 1, 0, 1)$. Скалярное произведение $(x, p) = \frac{3}{2} > h$, поэтому относим вектор к первому классу.

Пусть $\alpha_1, \alpha_2, \ldots, \alpha_{m_1}$ — веса строк матрицы T_1, $\beta_1, \beta_2, \ldots, \beta_{m_2}$ — веса строк матрицы T_2, $z = \min\limits_i a_i - \max\limits_j b_j$. Заметим, что алгоритм A_2 правильно классифицирует все векторы из обучающей выборки только при условии $z > 0$.

В случае $z \leqslant 0$ может быть использована модификация алгоритма A_2 — алгоритм A_3 [14, 15].

Для некоторого признака i заменим его на его отрицание. При этом в таблицах T_1 и T_2 i-ый столбец инвертируется. Такая замена не влияет на информационные веса признаков $p = (p_1, \ldots, p_n)$ и сохраняет все тупиковые тесты. Но при этом изменяются величины $a_1, a_2, \ldots, a_{m_1}$ и $b_1, b_2, \ldots, b_{m_2}$, и величина z может увеличиться. Если это произошло мы можем принять это инвертирование i-го признака, и перейти к рассмотрению следующего признака. При $z > 0$ тестовый алгоритм A_3 правильно классифицирует всю обучающую выборку, и в качестве пороговой величины h можно взять

$$h = (\min_i a_i + \max_j b_j)/2.$$

Приведем пример использования этого алгоритма:

Рассмотрим матрицы

$$\left(\frac{T_1}{T_2}\right) = \left(\begin{array}{cccc} 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 \\ \hline 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{array}\right).$$

Очевидно, тупиковые тесты $\mathcal{T} = \{(1, 2), (3), (4)\}$, а все информационные веса равны $\frac{1}{3}$. Тогда $\min \alpha_i = \frac{1}{3}$ и $\max \beta_j = \frac{2}{3}$, т.е. $z = -\frac{1}{3} < 0$ и алгоритм A_2, следовательно, неприменим.

Инвертируем третий столбец, тогда (справа указаны веса строк)

$$\left(\frac{T_1}{T_2}\right) = \left(\begin{array}{cccc|c} 1 & 1 & 1 & 1 & 4/3 \\ 0 & 0 & 1 & 1 & 2/3 \\ \hline 1 & 0 & 0 & 0 & 1/3 \\ 0 & 1 & 0 & 0 & 1/3 \end{array}\right),$$

следовательно $z = \frac{1}{3}$ и можно взять $h = \frac{1}{2} \cdot \left(\frac{2}{3} + \frac{1}{3}\right) = \frac{1}{2}$. В качестве примера классифицируем вектор $x = (0, 1, 0, 1)$. Сначала инвертируем его третий признак (т.к. мы это делали в матрице) $x_1 = (0, 1, 1, 1)$. Найдем $(x_1, p) = 1 > \frac{1}{2}$, следовательно, относим x к классу K_1.

Алгоритм Кудрявцева (голосования по тестам)

Пусть $\{1, 2, \dots, n\}$ — множество признаков. Договоримся подмножество $\tau \subset \{1, 2, \dots, n\}$ обозначать булевским вектором $t = (t_1, \dots, t_n)$, подразумевая, что признак $i \in \tau$ точно тогда, когда $t_i = 1$.

Пусть как и ранее $T_1 = \{a_j = (a_{j1}, \dots, a_{jn}) \,:\, j = 1, 2, \dots, m_1\}$, $T_2 = \{b_j = (b_{j1}, \dots, b_{jn}) \,:\, j = 1, 2, \dots, m_2\}$ — множества элементов обучающей выборки, принадлежащие классам K_1 и K_2 соответственно.

Обозначим $\mathcal{T} = \mathcal{T}(T_1, T_2)$ — опорное множество тестов для обучающей выборки T_1, T_2. В качестве опорного множества $\mathcal{T}$ могут выступать, например, множество всех тестов, множество тупиковых тестов, или множество тестов фиксированной длины.

Если $a = (a_1, \dots, a_n)$ — элемент обучающей выборки, $t = (t_1, \dots, t_n) \in \mathcal{T}$ — некоторый тест, а $x = (x_1, \dots, x_n)$ — объект, подлежащий классификации, то скажем, что *тест t голосует за элемент a на объекте x*, если

$$\gamma_t^a(x) = \prod_{i=1}^{n} (1 - t_i |x_i - a_i|) = 1,$$

т.е. на признаках из теста t объекты a и x совпадают.

Тогда сумму

$$\sum_{a \in T_j} \gamma_t^a(x)$$

можно интерпретировать как интеграл (по дискретной мере) голосования множества T_j, $j = 1, 2$, а многочлен

$$\Gamma_j(x) = \frac{1}{m_j} \sum_{t \in \mathcal{T}} \sum_{a \in T_j} \gamma_t^a(x)$$

есть интегральный голос всего опорного множества тестов $\mathcal{T}$ за класс K_j, $j = 1, 2$.

Многочлен $R(x) = \Gamma_2(x) - \Gamma_1(x)$ называется *многочленом голосования.*

Алгоритм голосования по тестам, предложенный В.Б.Кудрявцевым, состоит в том, что на объекте x, подлежащем классификации, вычисляется многочлен голосования $R(x)$, и если $R(x) \leqslant 0$, то x относится к классу K_1, и к классу K_2 — в противном случае.

Проиллюстрируем на примере матриц (1.1) и вектора $x = (1, 1, 0, 0, 1)$. В качестве опорного множества тестов выберем все тупиковые тесты: $\mathcal{T} = \{(1, 2); (1, 3); (2, 3); (4)\}$. Подсчитываем количество голосов по каждому тесту

$$\left(\begin{array}{c|c|c|c|c} & (1,2) & (1,3) & (2,3) & (4) \\ \hline T_1 & 1 & 0 & 0 & 0 \\ \hline T_2 & 0 & 0 & 0 & 2 \end{array}\right)$$

Тогда $\Gamma_1(x) = \frac{1}{3}(1 + 0 + 0 + 0) = \frac{1}{3}$, $\Gamma_2(x) = \frac{1}{2}(0 + 0 + 0 + 2) = 1$ и многочлен голосования $R(x) = 1 - \frac{1}{3} = \frac{2}{3} > 0$, поэтому относим объект к K_2.

Теорема Анселя.

Зафиксируем T_1, T_2 и рассмотрим множество всех тестов $\tau = \tau(T_1, T_2)$. Каждому такому τ поставим в соответствие характеристическую функцию $f_\tau(t) = \begin{cases} 1, & t \in \tau \\ 0, & t \notin \tau \end{cases}$. Очевидно, f_τ — монотонная[6] булевая функция. Верно и обратное — для произвольной монотонной функции $f \not\equiv const$ можно подобрать T_1 и T_2, для которых $f = f_{\tau(T_1,T_2)}$.

Нижней единицей монотонной функции f называется набор $\alpha \in E_2^n$, на котором $f(\alpha) = 1$, но $f(\beta) = 0$ для любого $\beta < \alpha$.

Верхним нулем монотонной функции f называется набор $\alpha \in E_2^n$, на котором $f(\alpha) = 0$, но $f(\beta) = 1$ для любого $\beta > \alpha$.

Последовательность $\beta^{(1)}, \beta^{(2)}, \ldots, \beta^{(m)}$ элементов из E^n называется *цепью*, если $\beta^{(i+1)}$ получается из $\beta^{(i)}$ заменой одного нуля (в наборе координат) на единицу, $i = 1, 2, \ldots, m-1$. Тем самым, $\beta^{(1)} < \beta^{(2)} < \ldots < \beta^{(m)}$.

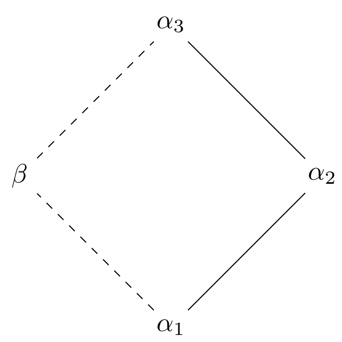


Рис. 1.9: Дополнение до квадрата

Если три элемента $\alpha_1, \alpha_2, \alpha_3$ из E^n образуют цепь, то четвертый элемент β, образующий вместе с ними квадрат (см. рис. **??**), называют дополнением цепи $\alpha_1, \alpha_2, \alpha_3$ до квадрата.

Лемма 1 (Анселя)**.** *Единичный n-мерный куб E^n может быть покрыт множеством из $C_n^{[n/2]}$ попарно непересекающихся цепей, обладающих следующими свойствами:*

а) число цепей длины $n - 2p + 1$ равно $C_n^p - C_n^{p-1}$ $(0 \leqslant p \leqslant \left[\frac{n}{2}\right])$, и минимальный элемент каждой такой цепи есть набор с p единицами и $n - p$ нулями, а максимальный — с p нулями и $n - p$ единицами;

б) если заданы три элемента $\alpha_1, \alpha_2, \alpha_3$, образующие цепь и принадлежащие одной и той же цепи длины $n - 2p + 1$, то дополнение цепи $\alpha_1, \alpha_2, \alpha_3$ до квадрата принадлежит цепи длины $n - 2p - 1$.

[6]Булева функция $f(x_1, \ldots, x_n)$ называется *монотонной*, если для любых наборов $(\alpha_1, \ldots, \alpha_n) \leqslant (\beta_1, \ldots, \beta_n)$ выполнено соотношение $f(\alpha_1, \ldots, \alpha_n) \leqslant f(\beta_1, \ldots, \beta_n)$.

Упражнения.

Задача 1.3.1. Даны матрицы признаков $T_1 = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}$ и $T_2 = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \end{pmatrix}$, для них известен набор тупиковых тестов:$[1, 3, 4], [1, 3, 5], [1, 3, 6], [1, 2, 5, 6], [1, 4, 5, 6]$. Классифицировать вектор $V = (0, 1, 0, 1, 0, 0)$ с помощью алгоритмов a) Журавлева; b) Королева/Переяславского; c) Кудрявцева.

Задача 1.3.2. Даны матрицы признаков $T_1 = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 0 \end{pmatrix}$, $T_2 = \begin{pmatrix} 1 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 \end{pmatrix}$, для них известен набор тупиковых тестов:$[1, 2, 4, 5], [1, 2, 3], [1, 6]$. Найти информационные веса признаков и классифицировать вектор $V = (1, 1, 0, 0, 0, 1)$ с помощью алгоритмов a) Журавлева; b) Королева/Переяславского; c) Кудрявцева.

Задача 1.3.3. Даны матрицы признаков $T_1 = \begin{pmatrix} 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}$, $T_2 = \begin{pmatrix} 1 & 1 & 1 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \end{pmatrix}$, для них известен набор тупиковых тестов:$[4, 5, 6], [1, 4, 5], [2, 4, 6], [1, 4, 6], [2, 4, 5]$. Найти информационные веса признаков и классифицировать вектор $V = (1, 1, 0, 1, 1, 1)$ с помощью алгоритмов a) Журавлева; b) Королева/Переяславского; c) Кудрявцева.

Задача 1.3.4. Даны матрицы признаков $T_1 = \begin{pmatrix} 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 1 & 0 \end{pmatrix}$, $T_2 = \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 0 & 1 & 1 \end{pmatrix}$, для них известен набор тупиковых тестов:$[4], [1], [2, 5, 7], [3, 5, 7], [2, 3, 5]$. Найти информационные веса признаков и классифицировать вектор $V = (1, 0, 1, 0, 1, 0, 0)$ с помощью алгоритмов a) Журавлева; b) Королева/Переяславского; c) Кудрявцева.

Задача 1.3.5. Даны матрицы признаков $T_1 = \begin{pmatrix} 0 & 0 & 1 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}$, $T_2 = \begin{pmatrix} 1 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 \end{pmatrix}$, для них известен набор тупиковых тестов:$[4, 5, 6], [1, 4, 5], [3, 5], [2, 4, 5]$. Классифицировать вектор $V = (0, 0, 0, 1, 0, 1)$ с помощью алгоритмов a) Журавлева; b) Королева/Переяславского; c) Кудрявцева.

Задача 1.3.6. Найти все тупиковые тесты и информационные веса признаков:

$$
\text{a)}\left(\begin{array}{c|c|c|c|c}
0&0&0&0&0\\
0&0&1&1&1\\
0&0&0&1&0\\
0&1&1&1&0\\
0&0&1&0&1\\
0&1&1&0&0\\ \hline
1&1&0&0&1\\
1&1&0&1&1
\end{array}\right);\ \text{b)}\left(\begin{array}{c|c|c|c|c}
0&1&1&1&1\\
0&0&1&0&0\\
1&1&1&0&1\\
0&1&1&1&0\\
0&0&1&0&1\\
1&1&1&0&0\\ \hline
1&0&0&1&0\\
1&0&0&1&1
\end{array}\right);\ \text{c)}\left(\begin{array}{c|c|c|c|c|c}
0&1&0&0&1&1\\
0&1&1&0&1&1\\
1&1&1&1&1&1\\
0&0&1&1&1&1\\
0&0&0&1&1&1\\
1&1&0&1&1&1\\ \hline
1&0&0&0&1&0\\
1&0&1&0&1&0
\end{array}\right);
$$

$$
\text{d)}\left(\begin{array}{c|c|c|c|c|c}
1&0&0&0&0&1\\
1&1&0&0&0&0\\
1&0&0&1&0&0\\
1&0&1&0&0&1\\
1&1&1&0&0&0\\
1&0&1&1&0&0\\ \hline
0&1&0&1&0&1\\
0&1&1&1&0&1
\end{array}\right);\ \text{e)}\left(\begin{array}{c|c|c|c|c|c}
0&0&0&0&1&0\\
1&1&1&0&1&0\\
0&1&0&1&1&0\\
0&1&1&0&1&0\\
1&1&0&1&1&0\\
1&0&0&0&1&0\\ \hline
0&0&1&1&1&1\\
0&0&1&1&0&1\\
1&0&1&1&1&1
\end{array}\right);\ \text{f)}\left(\begin{array}{c|c|c|c|c|c}
0&1&0&0&0&1\\
0&0&0&0&1&1\\
0&0&0&1&0&1\\
1&1&0&0&0&1\\
1&0&0&1&0&1\\
1&0&0&0&1&1\\ \hline
1&1&1&1&1&1\\
0&1&1&1&1&1\\
1&1&1&1&1&0
\end{array}\right).
$$

Задача 1.3.7. a) Найти все тупиковые тесты и информационные веса признаков: $\left(\begin{array}{c|c|c|c|c|c}
0&1&0&1&0&0\\
1&1&0&0&0&0\\
1&1&0&0&1&1\\
0&1&1&1&0&1\\
1&1&1&0&0&1\\
0&1&0&1&1&1\\ \hline
0&1&1&0&1&0\\
1&1&1&1&1&0\\
0&0&1&0&1&0\\
1&0&1&1&1&0
\end{array}\right)$. ... отнести набор (1,1,1,1,1,1) к одному из классов методом b) Журавлева; c) Королева/Переяславского; d) Кудрявцева.

Задача 1.3.8. a) Найти все тупиковые тесты и информационные веса признаков: $\left(\begin{array}{c|c|c|c|c|c}
0&0&1&1&0&0\\
1&0&0&0&0&1\\
1&1&0&0&0&0\\
1&0&1&0&0&0\\
0&0&0&1&0&1\\
0&1&0&1&0&0\\ \hline
0&1&1&0&1&1\\
1&1&1&1&0&1\\
0&1&1&0&0&1\\
1&1&1&1&1&1
\end{array}\right)$. Классифицировать вектор (1, 1, 1, 0, 0, 1) методом b) Журавлева; c) Королева/Переяславского; d) Кудрявцева.

Задача 1.3.9. a) Найти все тупиковые тесты и информационные веса признаков: $\left(\begin{array}{c|c|c|c|c|c} 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 1 & 0 & 1 & 1 & 1 \\ \hline 1 & 1 & 1 & 0 & 0 & 0 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 0 & 1 & 0 \end{array}\right)$. b) Классифицировать вектор (1, 1, 0, 0, 1, 0) методом Журавлева; c) ...Королева/Переяславского; d) ...Кудрявцева.

Задача 1.3.10. a) Найти все тупиковые тесты и информационные веса признаков: $\left(\begin{array}{c|c|c|c|c|c} 0 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 1 & 0 & 1 & 1 \\ \hline 1 & 0 & 1 & 1 & 0 & 1 \\ 1 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 & 1 & 1 \end{array}\right)$. b) Классифицировать вектор (0, 0, 0, 0, 0, 0) методом Журавлева; c) ...Королева/Переяславского; d) ...Кудрявцева.

Задача 1.3.11. a) Найти все тупиковые тесты и информационные веса признаков: $\left(\begin{array}{c|c|c|c|c|c} 1 & 1 & 1 & 1 & 1 & 0 \\ 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 & 1 & 1 \\ \hline 1 & 0 & 0 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 \end{array}\right)$. b) Классифицировать вектор (0, 0, 0, 0, 0, 0) методом Журавлева; c) ...Королева/Переяславского; d) ...Кудрявцева.

Задача 1.3.12. Даны матрицы признаков $T_1 = \begin{pmatrix} 0 & 1 & 1 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 \end{pmatrix}, T_2 =$ $= \begin{pmatrix} 1 & 1 & 0 & 1 & 1 & 0 \end{pmatrix}$, для них известен набор тупиковых тестов: $[1,2], [1,4,5], [3,4], [3,6], [1,5,6], [2,4]$. Найти информационные веса признаков и классифицировать вектор $V = (0,0,0,1,1,0)$ с помощью алгоритмов A_1, A_2/A_3 и A_K (голосования по тестам).

Задача 1.3.13. a) Найти все тупиковые тесты и информационные веса признаков: $\left(\begin{array}{c|c|c|c|c|c} 0 & 0 & 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 \\ \hline 1 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \end{array}\right)$. b) Классифицировать вектор (0, 1, 0, 0, 0, 0) методом Журавлева; c) ...Королева/Переяславского; d) ...Кудрявцева.

Задача 1.3.14. a) Найти все тупиковые тесты и информационные веса признаков: $\left(\begin{array}{c|c|c|c|c|c} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 1 & 0 & 1 & 1 \end{array}\right)$. b) Классифицировать вектор (1, 0, 0, 1, 1, 1) методом Журавлева; c) ...Королева/Переяславского; d) ...Кудрявцева.

Задача 1.3.15. Привести пример матриц признаков T_1, T_2, для которых тупиковыми будут следующие тесты (и только они): $[1, 2, 3], [1, 2, 4], [2, 3, 5], [4, 5]$.

Задача 1.3.16. Привести пример матриц признаковT_1, T_2, для которых тупиковыми будут следующие тесты (и только они): $[1], [2, 3, 4], [2, 3, 5], [2, 6]$.

Задача 1.3.17. Пусть размерность пространства признаков равна 4. Существуют ли такие матрицы T_1, T_2, которые имеют a) ровно три тупиковых теста длины 2 и один — длины 3? b) ...ровно три тупиковых теста длины 3 и один — длины 2?

Задача 1.3.18. Придумать пару матриц, имеющих: a) ровно один тупиковый тест длины 1, два длины 2 и три — длины 3. б) ровно три тупиковых теста длины 2, два длины 1 и один — длины 3.

Задача 1.3.19. Построить покрытие непересекающимися цепями булевского куба (как в Т. Анселя): a) E_2^2; b) E_2^3; c) E_2^4.

Задача 1.3.20. Найти a) $\varphi(2)$ — число монотонных функций от 2 переменных; b) $\varphi(3)$.

Задача 1.3.21. Найти все нижние единицы и верхние нули функции a) $f(x_1, x_2, x_3) = x_1x_2 \vee x_2x_3 \vee x_1x_3$; b) $f(x_1, \ldots, x_5) = (x_2 \vee x_3 \vee x_4) \wedge (x_1 \vee x_5)$.

Задача 1.3.22. Пусть размерность пространства признаков (бинарных) равна 5.

a) Доказать, что количество тупиковых тестов для любой пары таблиц T_2 и T_2 не превосходит 10.
b) Привести пример пары, для которой их ровно 10.

Задача 1.3.23. Пусть размерность пространства признаков (бинарных) равна 6. а) Доказать, что количество тупиковых тестов для любой пары таблиц T_2 и T_2 не превосходит 15. б) Привести пример пары, для которой их ровно 15. в) Найти оценку для числа тупиковых тестов в случае, когда размерность равна n.

Задача 1.3.24. Привести пример двух матриц и вектора, на которых результаты алгоритма A_3 зависят от порядка инверсии столбцов. Т.е. при одном порядке вектор классифицируется как K_1, а при другом — как K_2.

Задача 1.3.25. Найдите какое-либо тупиковое разрешающее множество для класса: a) T_1; b) L; c) S.

Ответы и указания.

1.3.1 Алгоритм A_1: $G_0 = \frac{31}{20}$, $G_1 = \frac{11}{10}$ — относим к K_2; Алгоритм A_K: $G_0 = \frac{5}{4}$, $G_1 = \frac{1}{2}$ — относим к K_1;

1.3.2 Информационные веса признаков: $W = (1, \frac{2}{3}, \frac{1}{3}, \frac{1}{3}, \frac{1}{3}, \frac{1}{3})$. Алгоритм A_1: $G_0 = \frac{25}{12}$, $G_1 = 1$ — относим к K_2; Алгоритм A_K: $G_0 = 0$, $G_1 = \frac{1}{2}$ — относим к K_2;

1.3.3 Информационные веса признаков: $W = (\frac{2}{5}, \frac{2}{5}, 0, 1, \frac{3}{5}, \frac{3}{5})$. Алгоритм A_1: $G_0 = \frac{11}{10}$, $G_1 = \frac{23}{15}$ — относим к K_1; Алгоритм A_K: $G_0 = \frac{3}{2}$, $G_1 = 0$ — относим к K_1;

1.3.4 Информационные веса признаков: $W = (\frac{1}{5}, \frac{2}{5}, \frac{2}{5}, \frac{1}{5}, \frac{3}{5}, 0, \frac{2}{5})$. Алгоритм A_1: $G_0 = \frac{6}{5}$, $G_1 = \frac{2}{3}$ — относим к K_2; Алгоритм A_K: $G_0 = 1$, $G_1 = 3$ — относим к K_2;

1.3.5 Алгоритм A_1: $G_0 = \frac{17}{16}$, $G_1 = \frac{7}{4}$ — относим к K_1; Алгоритм A_K: $G_0 = \frac{3}{2}$, $G_1 = 0$ — относим к K_1;

1.3.6 a) Тупиковые тесты: $(4,1)$, $(1,3)$, $(4,3)$, (5); b) Тупиковые тесты: $(2,1)$, $(1,4)$, $(2,4)$, (3); c) Тупиковые тесты: $(2,1)$, $(1,5)$, $(2,5)$, (3) d) Тупиковые тесты: $(4,1)$, $(1,5)$, $(4,5)$, (3), e) Тупиковые тесты: $(5,3)$, $(3,1)$, $(5,1)$, (2); f) Тупиковые тесты: $(5,3)$, $(3,4)$, $(5,4)$, (1).

1.3.7 a) Тупиковые тесты: $(4,6)$, $(6,2)$, $(4,2)$, $(5,1)$,

1.3.8 a) Тупиковые тесты: $(6,2)$, $(2,3)$, $(6,3)$, $(4,1)$; b) Информационные веса признаков: $(\frac{1}{4}, \frac{1}{2}, \frac{1}{2}, \frac{1}{4}, 0, \frac{1}{2})$; Алгоритм A_1: $r_0 = \frac{5}{4}$, $r_1 = \frac{1}{4}$, относим к K_2; Алгоритм A_2/A_3: инвертируем 1–й и 4–й столбцы, $\min_1 = \frac{3}{2}$; $\max_0 = \frac{3}{4}$, Граница $h = \frac{9}{8}$. Значение на векторе $\frac{1}{4}$, следовательно, относим к K_1. Алгоритм Кудрявцева: $\Gamma_1 = \frac{1}{2}$, $\Gamma_2 = 3$ — относим к K_2.

1.3.9 a) Тупиковые тесты: $(6,4)$, $(4,1)$, $(6,1)$, $(5,2)$. Информационные веса признаков: $(\frac{1}{2}, \frac{1}{4}, 0, \frac{1}{2}, \frac{1}{4}, \frac{1}{2})$; b) Алгоритм A_1: $r_0 = \frac{5}{4}$, $r_1 = \frac{1}{4}$, относим к K_2. Алгоритм A_2/A_3: инвертируем 1–й и 3–й столбцы, $\min_0 = 1$, $\max_1 = \frac{1}{4}$, граница $h = \frac{5}{8}$. Значение на векторе $\frac{1}{4}$, следовательно, относим к K_2. Алгоритм Кудрявцева: $\Gamma_1 = 0.5$, $\Gamma_2 = 3$, относим к K_2

1.3.10 a) Тупиковые тесты: $(4,2)$, $(2,1)$, $(4,1)$, $(5,3,6)$, информационные веса признаков: $(\frac{1}{2}, \frac{1}{2}, \frac{1}{4}, \frac{1}{2}, \frac{1}{4}, \frac{1}{4})$. b) Алгоритм A_1: $r_0 = \frac{17}{12}$, $r_1 = \frac{11}{8}$, относим к K_2. Алгоритм A_2/A_3: инвертируем 1–й и 4–й, $\min_0 = \frac{7}{4}$, $\max_1 = \frac{1}{2}$, граница $h = \frac{9}{8}$. Значение на векторе равно $p \cdot v = \frac{7}{4}$, значит относим к K_1 Алгоритм Кудрявцева: $\Gamma_1 = 1$, $\Gamma_2 = \frac{1}{4}$ — относим к K_1.

1.3.11 a) Тупиковые тесты: $(1,4)$, $(4,6)$, $(1,6)$, $(5,3,2)$, Информационные веса признаков: (0.5, 0.25, 0.25, 0.5, 0.25, 0.5) Алгоритм A_1: $r_0 = 1.416667$, $r_1 = 1.375$, относим к K_2. Алгоритм A_2/A_3: инвертируем первый и последний столбцы. $\min_0 = 1.75$, $\max_1 = 0.5$, граница 1.125. Значение на векторе 1.75, следовательно, относим к K_1. Алгоритм Кудрявцева: $\Gamma_1 = 1$, $\Gamma_2 = 0.25$ относим к K_1.

1.3.12 Информационные веса признаков: $W = (\frac{1}{2}, \frac{1}{3}, \frac{1}{3}, \frac{1}{2}, \frac{1}{3}, \frac{1}{3})$. Алгоритм A_1: $G_0 = \frac{35}{36}$, $G_1 = \frac{5}{6}$ — относим к K_2; Алгоритм A_K: $G_0 = \frac{3}{2}$, $G_1 = 2$ — относим к K_2;

1.3.13 a) Тупиковые тесты: $(5,3)$, $(3,6)$, $(5,6)$, (2). Информационные веса признаков: $w = (0, \frac{1}{4}, \frac{1}{2}, 0, \frac{1}{2}, \frac{1}{2})$; веса признаков, умноженные на 4: $\hat{w} = (0,1,2,0,2,2,)$; b) Алгоритм A_1: $r_0 = \frac{13}{3}$, $r_1 = 2$, относим к K_2 Алгоритм A_3: инвертируем 1–й и 2–й и 5–й столбцы $\min(K_1) = 5$, $\max(K_2) = 0$, граница $\frac{5}{2}$. Значение на векторе равно 3 , следовательно, относим к K_1 Алгоритм Кудрявцева: $\Gamma_1 = 2/3$, $\Gamma_2 = 2$, следовательно, относим к K_2.

1.3.14 Тупиковые тесты: $(4,5)$, $(5,2)$, $(4,2)$, (6), . Информационные веса признаков: $w = (0, \frac{1}{2}, 0, \frac{1}{2}, \frac{1}{2}, \frac{1}{4})$; домноженные веса признаков: $\hat{w} = (0,2,0,2,2,1,)$; Алгоритм A_1: $r_0 = 13/3$, $r_1 = 2$, относим к K_2 Алгоритм A_3: инвертируем 2–й и 4–й столбцы, $\min(K_2) = 7$ $\max(K_1) = 2$, граница $T = \frac{9}{2}$. Значение на векторе равно 3, следовательно, относим к K_1 Алгоритм Кудрявцева: $\Gamma_1 = \frac{2}{3}$, $\Gamma_2 = 2$, следовательно относим к K_2

1.3.15 Например, $T_1 = \begin{pmatrix} 0 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \end{pmatrix}$, $T_2 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \end{pmatrix}$.

1.3.16 Например, $T_1 = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 1 & 0 & 0 & 1 \end{pmatrix}$, $T_2 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$.

1.3.17 a) Да, существуют. Выберем следующие тесты в качестве тупиковых: $[1,2], [1,3], [1,4], [2,3,4]$. Запишем тесты в виде ДНФ: $D = (x_1 \wedge x_2) \vee \vee (x_1 \wedge x_3) \vee (x_1 \wedge x_4) \vee (x_2 \wedge x_3 \wedge x_4)$ и преобразуем к КНФ, после упрощений получим: $D = (x_1 \vee x_2) \wedge (x_1 \vee x_3) \wedge (x_1 \vee x_4) \wedge (x_2 \vee x_3 \vee x_4)$. Тогда в качестве матриц можно взять $T_1 = \begin{pmatrix} 1 & 0 & 1 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 \\ 1 & 1 & 0 & 0 \end{pmatrix}$, $T_2 = \begin{pmatrix} 0 & 0 & 0 & 0 \end{pmatrix}$.

b) Нет, не существуют. Всего существует 4 подмножества из 3 элементов. Если взять любые 3 в качестве тупиковых тестов, например $[1,2,3], [1,2,4], [1,3,4]$, то любое подмножество из 2 элементов покрывается ими (и, следовательно, не может быть тестом).

1.3.18 a) Найдем матрицы ,у которых тупиковые тесты $[1], [2,3], [2,4], [2,5,6], [3,5,6], [4,5,6]$. Запишем ДНФ $D = x_1 \vee (x_2 \wedge x_3) \vee (x_2 \wedge x_4) \vee (x_2 \wedge x_5 \wedge x_6) \vee (x_3 \wedge x_5 \wedge x_6) \vee \vee (x_4 \wedge x_5 \wedge x_6)$, преобразуем в КНФ и упростим: $(x_1 \vee x_2 \vee x_5) \wedge \wedge (x_1 \vee x_2 \vee x_6) \wedge (x_1 \vee x_2 \vee x_3 \vee x_4) \wedge (x_1 \vee x_2 \vee x_3 \vee x_5) \wedge (x_1 \vee x_2 \vee x_3 \vee x_6) \wedge \wedge (x_1 \vee x_2 \vee x_4 \vee x_5) \wedge (x_1 \vee x_2 \vee x_4 \vee x_6) \wedge (x_1 \vee x_2 \vee x_5 \vee x_6) \wedge (x_1 \vee x_3 \vee x_4 \vee x_5) \wedge \wedge (x_1 \vee x_3 \vee x_4 \vee x_6) \wedge (x_1 \vee x_2 \vee x_3 \vee x_4 \vee x_5) \wedge (x_1 \vee x_2 \vee x_3 \vee x_4 \vee x_6) \wedge \wedge (x_1 \vee x_2 \vee x_3 \vee x_5 \vee x_6) \wedge (x_1 \vee x_2 \vee x_4 \vee x_5 \vee x_6) \wedge (x_1 \vee x_3 \vee x_4 \vee x_5 \vee x_6) \wedge \wedge (x_1 \vee x_2 \vee x_3 \vee x_4 \vee x_5 \vee x_6) = (x_1 \vee x_2 \vee x_6) \wedge (x_1 \vee x_3 \vee x_4 \vee x_5) \wedge \wedge (x_1 \vee x_2 \vee x_3 \vee x_4) \wedge (x_1 \vee x_2 \vee x_5) \wedge (x_1 \vee x_3 \vee x_4 \vee x_6)$. Запишем мат-

рицы: $T_1 = \begin{pmatrix} 1 & 1 & 0 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ 1 & 1 & 0 & 0 & 1 & 0 \\ 1 & 0 & 1 & 1 & 0 & 1 \end{pmatrix}$ и $T_2 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$.
b) Аналогично, тесты : $[1,2],[1,3],[1,4],[5],[6],[2,3,4]$. Матрицы: $T_1 = \begin{pmatrix} 1 & 0 & 0 & 1 & 1 & 1 \\ 0 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 & 1 \end{pmatrix}$, $T_2 = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 & 0 \end{pmatrix}$.

1.3.19 c) Две цепи длины 1: [0101], [0001]; три цепи длины 3: $[0100 - 0110 - 0111]$; $[0010 - 1010 - 1011]$, $[0001 - 1001 - 1101]$; и одна цепь длины 5: $[0000 - 1000 - 1100 - 1110 - 1111]$.

1.3.20 a) 6; b) 20.

1.3.21 a) Нижние единицы: $[110, 101, 011]$; верхние нули: $[100, 010, 001]$. b) Нижние единицы: $[11000, 10100, 10010, 01001, 00101, 00011]$; верхние нули: $[01110, 10001]$.

1.3.22 a) По Т. Анселя число цепей для 5-мерного булева куба равно $C_5^2 = 10$. Каждой цепи может соответствовать не более одного тупикового теста. б) Возьмем $T_1 = I$ (единичная матрица размера 5×5) и $T_2 = (1,1,1,1,1)$. Тогда любая пара признаков будет тупиковым тестом, их количество равно $C_5^2 = 10$.

1.3.23 Аналогично задаче 1.3.22.

1.3.24 Рассмотрим $T_1 = \begin{pmatrix} 0 & 1 & 0 & 1 \end{pmatrix}$, $T_2 = \begin{pmatrix} 1 & 0 & 1 & 0 \end{pmatrix}$ и вектор $V = (1,1,0,0)$. Очевидно, что каждый признак является тупиковым тестом и информационные веса каждого признака равны $\frac{1}{4}$, взвешенная сумма каждой строки (в исходных матрицах) равна $\frac{1}{2}$. Если инвертировать первый или третий столбец, то $\min_1 = \frac{3}{4}$, $\max_2 = \frac{1}{4}$, т.е. границу можно взять, например $h = \frac{1}{2}$. При этом, вес вектора, если инвертировать 1-й столбец равен $p \cdot v = \frac{1}{4}$, т.е. относим к K_2, а если инвертировать 3-й столбец, то $p \cdot v = \frac{3}{4}$ — относим к K_1.

1.3.25 a) $E_2^n \setminus \{(0,...,0)\}$; b) $\{(0,...,0),(1,0,...,0),...,(0,...,0,1)\}$ (все вектора веса 0 и 1). c) $\{0\} \times E_2^{n-1}$, т.е. все вектора, у которых первый компонент равен нулю.

1.4. Персептрон. Теорема Новикова.

Персептрон.

В середине 50-х годов была предложена модель нейрона (см Рис. 1.10) — живой клетки которая, различая по заряду цитоплазмы, может находиться в двух состояниях: покоя и возбуждения.

Аксоны одних клеток соединены с дендритами других. Текущее состояние нейрона зависит от состояния аксонов соединенных с ним других нейронов и чувствительности дендритов.

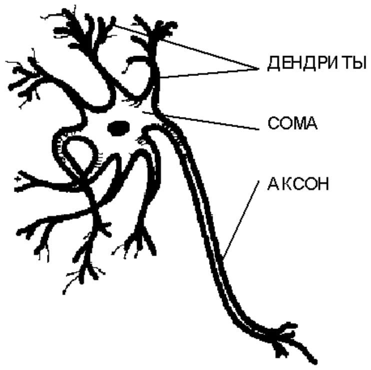


Рис. 1.10: Общая схема строения биологического нейрона.

Основываясь на этой модели нейрона Розенблатт (F.Rosenblatt) в 1957 году предложил модель персептрона (PERCEPTRON), одну из первых искусственных сетей, способных к перцепции (восприятию) и формированию реакции на воспринятый стимул.

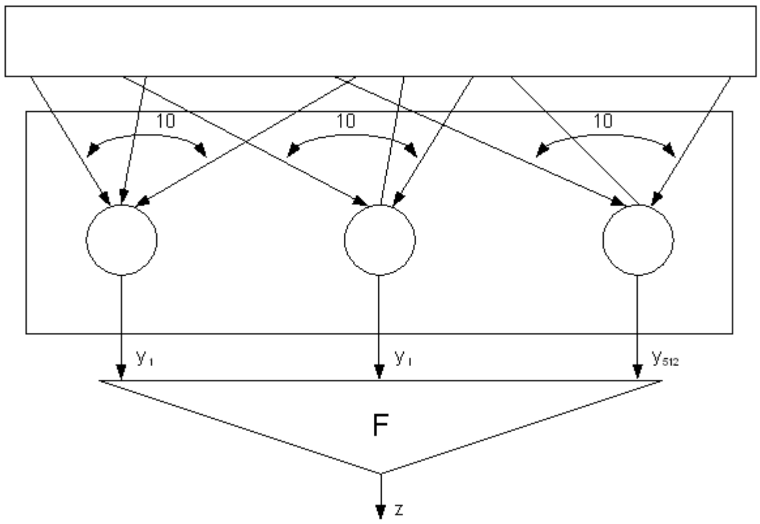


Рис. 1.11: Персептрон Розенблатта

Это физическое устройство, состоящее из трех слоев:

- рецепторный слой (20×20 фотоэлементов);
- передающий слой (512 нейронов, каждый имеет по 10 входов, случайным образом соединенных с элементами рецепторного слоя), причем для каждого $j = 1, 2, \ldots, 512$ имеет место

$$y_j = \begin{cases} 1, \sum_{i=1}^{10} x_{ji} \geqslant 5 \\ 0, \text{ иначе} \end{cases},$$

где x_{ji} — входы j-го элемента рецепторного слоя;

- решающий элемент, принимающий значение
$$z = F(y_1, \dots, y_{512}) = \begin{cases} 1, & \text{если } \bar{a} \cdot \bar{y} \geqslant c \\ 0, & \text{иначе} \end{cases},$$
где $\bar{a}$ — весовой вектор решающего элемента, c — пороговое число.

Считается, что схема соединения нейронов фиксирована и не может изменяться в процессе обучения, а дендриты нейронов могут менять чувствительность. То есть в процессе обучения персептрона можно менять весовой вектор $\bar{a}$ и порог c.

В данной схеме образ подается на рецепторный слой. Поскольку каждый элемент передающего слоя жестко связан с элементами рецепторного слоя, то на входе решающего элемента образ кодируется вектором длины, равной количеству элементов в передающем слое, в данном случае 512. Этот вектор $\bar{y} = (y_1, \dots, y_{512})$ называется вектором признаков.

Считается, что образы, подаваемые на персептрон, принадлежат одному из двух классов. Хотелось бы так настроить весовые коэффициенты $\bar{a} = (a_1, \dots, a_{512})$ решающего элемента, чтобы на образах из первого класса решающий элемент выдавал 0, а на образах из второго класса — 1. Настройка весовых коэффициентов осуществляется с помощью обучающего алгоритма, на вход которого поступает обучающая выборка, т.е. последовательность образов, для которых известно, к какому классу они принадлежат. В зависимости от правильности отнесения очередного образа из обучающей выборки к своему классу обучающий алгоритм может менять весовые коэффициенты.

Возникает вопрос: какие классы образов могут распознаваться персептроном? Ответ на этот вопрос тривиален. Из самого вида решающего правила видно, что распознаваться могут только классы, которые в признаковом пространстве могут быть отделены друг от друга гиперплоскостью. Тогда встает вопрос: если классы образов отделимы гиперплоскостью в признаковом пространстве, то существует ли алгоритм обучения персептрона? Ответ на этот вопрос дает теорема американского ученого Новикова.

Теорема Новикова

Пусть $\mathbb{R}^k$ — признаковое пространство, $M_i \subseteq \mathbb{R}^k$, $M_i \neq \varnothing$, $i = 1, 2$, $M_1 \bigcap M_2 = \varnothing$. Множества M_1 и M_2 *линейно отделимы* (*строго линейно отделимы*) точно тогда, когда существуют $\bar{a} \in \mathbb{R}^k$ и $c \in \mathbb{R}$, что для любых $\bar{x}_1 \in M_1$ и $\bar{x}_2 \in M_2$ выполнено $\bar{a} \cdot \bar{x}_1 \geqslant c, \bar{a} \cdot \bar{x}_2 < c$ $(\bar{a} \cdot \bar{x}_1 > c, \bar{a} \cdot \bar{x}_2 < c)$.

Множества $M_1, M_2 \subseteq \mathbb{R}^k$ строго 0-отделимы точно тогда, когда существуют $\bar{a} \in \mathbb{R}^n$, что для любых $\bar{x}_1 \in M_1$ и $\bar{x}_2 \in M_2$ выполнено $\bar{a} \cdot \bar{x}_1 > 0, \bar{a} \cdot \bar{x}_2 < 0$.

Множество $M \subseteq \mathbb{R}^k$ строго линейно отделимо от нуля точно тогда, когда существует $\bar{a} \in \mathbb{R}^k$, что для любых $\bar{x} \in M$ выполнено $\bar{a} \cdot \bar{x} > 0$.

Утверждение 1. *Если $M_1, M_2 \in \mathbb{R}^k$ и $\widetilde{M_i} = \{(x_1, \dots, x_k, 1) : (x_1, \dots, x_k) \in \in M_i\}, i = 1, 2$, то M_1 и M_2 строго линейно отделимы тогда и только тогда, когда $\widetilde{M_1}$ и $\widetilde{M_2}$ строго 0-отделимы.*

Утверждение 2. *Если $M_1, M_2 \in \mathbb{R}^k$ и $M' = M_1 \bigcup \{-\bar{x} : \bar{x} \in M_2\}$, то M_1 и M_2 строго 0-отделимы тогда и только тогда, когда M' строго линейно отделимо от 0.*

Пусть даны два строго линейно отделимых множества $M_1, M_2 \subset \mathbb{R}^{n-1}$. Пусть $M' = M_1 \bigcup \{-\bar{x} : \bar{x} \in M_2\}$ и $M = \{(x_1, \dots, x_{n-1}, 1) : (x_1, \dots, x_{n-1}) \in \in M'\} \subset \mathbb{R}^n$. Тогда согласно утверждениям 1 и 2 множество M строго линейно отделимо от нуля.

Рассмотрим следующий алгоритм A, который для строго линейно отделимого от нуля множества M позволяет находить нормаль отделяющей гиперплоскости. На вход алгоритма поступает бесконечная последовательность $\bar{y}_1, \bar{y}_2, \dots$ такая, что для любого $i = 1, 2, \dots$ выполнено $\bar{y}_i \in M$. Эта последовательность называется обучающей последовательностью. Алгоритм A состоит в итеративном уточнении нормали $\bar{a}$ отделяющей гиперплоскости, называемой весовым вектором, по следующей схеме:

Шаг 0. $\bar{a}_0 := (0, \dots, 0)$.

Шаг i $(i > 0)$. Если $\bar{y}_i \cdot \bar{a}_{i-1} \leftrightarrow 0$, то $\bar{a}_i = \bar{a}_{i-1} + \bar{y}_i$, иначе $\bar{a}_i = \bar{a}_{i-1}$.

Для конечного $M = \{\bar{y}_1, \dots, \bar{y}_s\}$ из $\mathbb{R}^n$ введем обозначения:

$$D(M) = \max_{\bar{y} \in M} ||\bar{y}||,$$

$$V(M) = \left\{ \sum_{i=1}^{s} \alpha_i \bar{y}_i : \sum_{i=1}^{s} \alpha_i = 1, \alpha_i \geq 0, i = 1, \dots, s \right\} \text{ —}$$

выпуклая оболочка множества M,

$$\rho(M) = \min_{\bar{y} \in V(M)} ||\bar{y}||.$$

Теорема 3 (Новикова). *Пусть $M \subset \mathbb{R}^n$ строго линейно отделимо от 0, $|M| < \infty$ и $\rho(M) > 0$. Пусть $\bar{y}_1, \bar{y}_2, \dots$ — обучающая последовательность такая, что для любого $i = 1, 2, \dots$ выполнено $\bar{y}_i \in M$, и каждый элемент $\bar{y}$ из M встречается в обучающей последовательности бесконечное число раз. Пусть $\bar{a}_1, \bar{a}_2, \dots$ — последовательность весовых векторов, полученных в результате применения алгоритма A к обучающей последовательности $\bar{y}_1, \bar{y}_2, \dots$. Тогда существует натуральное N такое, что для любого $i \geq N$ выполняется $\bar{a}_i = \bar{a}_N$, при этом для любого $\bar{y} \in M$ справедливо $\bar{a}_N \cdot \bar{y} > > 0$, и для числа изменений s в последовательности $\bar{a}_1, \bar{a}_2, \dots$ выполнено $s \leqslant \left[\frac{D^2(M)}{\rho^2(M)}\right]$.*

Упражнения.

Задача 1.4.1. а) Построить выпуклую оболочку множества $M = = \{(-3; -5), (-1; -2), (0; -1), (1; 1), (2; 3), (5; 8)\}$. Показать, что множество M линейно отделимо от 0. б) Найти $\rho(M)$ и $D(M)$. Выписать оценку из теоремы Новикова.

Задача 1.4.2. Построить разделяющую прямую для множеств: $A_1 = = \{(3; 3), (3; -5)\}$ $A_2 = \{(-3; 4), (0; 0)\}$

Задача 1.4.3. Построить разделяющую прямую для множеств: $A_1 = = \{(-4; 2), (1; 5), (2; -1)\}$ $A_2 = \{(-3; -4), (0; -2)\}$

Задача 1.4.4. Построить разделяющую прямую для множеств: $A_1 = \{(-1;4),(4;0)\}$ $A_2 = \{(-5;1),(0;1)\}$

Задача 1.4.5. Построить разделяющую прямую для множеств: $A_1 = \{(-2;5),(-4;4),(2;-4)\}$ $A_2 = \{(2;-1),(2;-1),(0;4)\}$

Задача 1.4.6. Построить разделяющую прямую для множеств: $A_1 = \{(-3;-2),(-2;5),(-3;1)\}$ $A_2 = \{(1;0),(1;3),(-1;0)\}$

Задача 1.4.7. Построить разделяющую плоскость для множеств: $A_1 = \{(-2;5;-2),(-2;-2;3),(-3;-4;2)\}$ и $A_2 = \{(2;4;-2),(-5;-2;-4),(-1;1;-1)\}$

Задача 1.4.8. Построить разделяющую плоскость для множеств: $A_1 = \{(-2;0;-1),(-3;5;1)\}$ и $A_2 = \{(1;4;-2),(2;5;3),(-1;3;-4)\}$

Задача 1.4.9. Построить разделяющую плоскость для множеств: $A_1 = \{(3;-4;-1),(2;-1;0),(2;0;-2)\}$ и $A_2 = \{(3;0;0),(-1;3;4),(2;5;-3)\}$

Задача 1.4.10. Построить разделяющую плоскость для множеств: $A_1 = \{(-5;-5;3),(-5;-5;3),(-5;5;-5)\}$ и $A_2 = \{(-4;5;-2),(-3;5;-3),(4;-2;3)\}$

Задача 1.4.11. Построить разделяющую плоскость для множеств: $A_1 = \{(-5;0;2),(5;3;3),(2;-1;2),(-3;-3;1)\}$ и $A_2 = \{(3;-5;-3),(0;1;-4),(-2;-4;0)\}$

Задача 1.4.12. Доказать, что множества $M_1, M_2 \subset \mathbb{R}^2$ нельзя отделить линейной функцией и построить нелинейную функцию разделяющую их, если

а) $M_1 = \{(0,2),(0,-2),(1,-2),(1,4),(2,3),(-1,3),(-1,-2)\}$ и $M_2 = \{(1,0),(2,0),(2,-1),(3,1),(4,-2),(5,2)\}$;

б) $M_1 = \{1,0),(2,0),(2,1),(3,-1),(4,2),(5,-2)\}$ и $M_2 = \{(0,-2),(0,2),(1,2),(1,-4),(2,-3),(-1,-3),(-1,2)\}$.

Задача 1.4.13. Какие функции алгебры логики от двух переменных представимы персептроном, а какие – нет? Ответ обосновать.

Задача 1.4.14. Привести пример двух линейно отделимых множеств, не разделяемых алгоритмом Новикова.

Задача 1.4.15. Дано множество $M \subset \mathbb{R}^n$, строго линейно отделимое от 0, $|M| = 7$, $D(M) = 10$, $\rho(M) = 1$. Во власти экспериментатора составить обучающую выборку для подачи на вход персептрона. Каждую секунду на вход персептрона может подаваться 1 элемент обучающей выборки. Чему равно минимальное время, через которое экспериментатор гарантированно будет знать, что персептрон получил нормальный вектор отделяющей гиперплоскости для множества M? Как должна выглядеть обучающая выборка, чтобы обеспечить этот результат?

Задача 1.4.16. а) Координаты(x, y) точек множества $M \subset \mathbb{R}^2$ принимают значения только $0, \pm1$. Докажите, что при построении 0-отделяющей прямой произойдет не более 10 изменений весового вектора.

b*) Пусть координаты принимают значения из множества $\{0, \pm1, \pm2, \dots, \pm n\}$. Докажите что число изменений весового вектора не превосходит $16n^4$.

Задача 1.4.17.

Пусть $M_1 = \{(m,n) \in \mathbb{N}^2 \,|\, n,m \leqslant 20, n^2 > 2m^2\}$; $M_2 = \{(m,n) \in \in \mathbb{N}^2 \,|\, n,m \leqslant 20, n^2 < 2m^2\}$. а) Доказать, что M_1 и M_2 являются линейно 0–отделимыми. б*) Оценить число изменений весового вектора.

Задача 1.4.18. Будем считать элементарными следующие операции: i) вычисление скалярного произведения; ii) сравнение его с нулем; iii) изменение весового вектора.
а) Оценить время работы алгоритма Новикова на компьютере, производительности ν эл.оп./сек. на линейно отделимом от нуля множестве $|M| = m$, для которого известны $\rho = \rho(M)$ и $D = D(M)$. б) Рассчитать это время для $\nu = 2 \cdot 10^9$, $m = 500$, $\rho = 1$, $D = 200$.

Ответы и указания.

1.4.1 а) Выпуклая оболочка — треугольник с вершинами $A(-3,-5)$, $B(0,-1)$ и $C(5,8)$. Точка $(0,0)$ лежит вне этого треугольника, поэтому M отделимо от нуля. б) Проведем прямую AC, ее уравнение $13x-8y-1=0$. Найдем расстояние $\rho = \frac{1}{\sqrt{13^2+(-8)^2}} = \frac{1}{\sqrt{233}}$. Очевидно $D(M) = \sqrt{5^2+8^2} = \sqrt{89}$. По теореме Новикова получаем оценку $s \leqslant D^2/\rho^2 = 20737$.

1.4.2 Разделяющая прямая: $6x - 2y - 1 = 0$. Изменения весового вектора: $a_1 = [0,0,0]$; $a_2 = [3,3,1]$; $a_3 = [6,-2,2]$; $a_4 = [6,-2,1]$; $a_5 = [6,-2,0]$; $a_6 = [6,-2,-1]$;

1.4.3 Разделяющая прямая: $x + 5y + 4 = 0$. Изменения весового вектора: $a_1 = [0,0,0]$; $a_2 = [-4,2,1]$; $a_3 = [-2,1,2]$; $a_4 = [1,5,1]$; $a_5 = [3,4,2]$; $a_6 = = [-1,6,3]$; $a_7 = [1,5,4]$;

1.4.4 Разделяющая прямая: $2x + 2y - 3 = 0$. Изменения весового вектора: $a_1 = [0,0,0]$; $a_2 = [-1,4,1]$; $a_3 = [3,4,2]$; $a_4 = [3,3,1]$; $a_5 = [3,2,0]$; $a_6 = = [3,1,-1]$; $a_7 = [2,5,0]$; $a_8 = [2,4,-1]$; $a_9 = [2,3,-2]$; $a_10 = [2,2,-3]$;

1.4.5 Разделяющая прямая: $2 - y - 2x = 0$. Изменения весового вектора: $a_1 = [0,0,0]$; $a_2 = [-2,5,1]$; $a_3 = [0,1,2]$; $a_4 = [-2,2,1]$; $a_5 = [-2,-2,0]$; $a_6 = [-4,3,1]$; $a_7 = [-2,-1,2]$;

1.4.6 Разделяющая прямая: $-2x-3 = 0$. Изменения весового вектора: $a_1 = = [0,0,0]$; $a_2 = [-3,-2,1]$; $a_3 = [-5,3,2]$; $a_4 = [-6,0,1]$; $a_5 = [-5,0,0]$; $a_6 = [-4,0,-1]$; $a_7 = [-3,0,-2]$; $a_8 = [-2,0,-3]$;

1.4.7 Разделяющая плоскость: $4\,y - x + 11\,z + 1 = 0$. Изменения весового вектора: $a_1 = [0,0,0,0]$; $a_2 = [-2,5,-2,1]$; $a_3 = [-4,3,1,2]$; $a_4 = [-6,-1,3,1]$; $a_5 = [-1,1,7,0]$; $a_6 = [-3,6,5,1]$; $a_7 = [-6,2,7,2]$; $a_8 = [-1,4,11,1]$;

1.4.8 Разделяющая плоскость: $4\,z - 2\,y - 10\,x = 0$. Изменения весового вектора: $a_1 = [0,0,0,0]$; $a_2 = [-2,0,-1,1]$; $a_3 = [-3,-4,1,0]$; $a_4 = [-6,1,2,1]$; $a_5 = [-8,-4,-1,0]$; $a_6 = [-7,-7,3,-1]$; $a_7 = [-10,-2,4,0]$;

1.4.9 Разделяющая плоскость: $-2\,x - 5\,y - 3\,z = 0$. Изменения весового вектора: $a_1 = [0,0,0,0]$; $a_2 = [3,-4,-1,1]$; $a_3 = [0,-4,-1,0]$; $a_4 = [-3,-4,-1,-1]$; $a_5 = [-1,-5,-1,0]$; $a_6 = [1,-5,-3,1]$; $a_7 = [-2,-5,-3,0]$;

1.4.10 Разделяющая плоскость: $-7x - 8y - 4z - 1 = 0$. Изменения весового вектора: $a_1 = [0,0,0,0]$; $a_2 = [-5,-5,3,1]$; $a_3 = [-10,0,-2,2]$; $a_4 = [-6,-5,0,1]$; $a_5 = [-2,-10,2,0]$; $a_6 = [-6,-8,-1,-1]$; $a_7 = [-11,-3,-6,0]$; $a_8 = [-7,-8,-4,-1]$;

1.4.11 Разделяющая плоскость: $6y - 3x + 8z + 2 = 0$. Изменения весового вектора: $a_1 = [0,0,0,0]$; $a_2 = [-5,0,2,1]$; $a_3 = [0,3,5,2]$; $a_4 = [-3,0,6,3]$; $a_5 = [-1,4,6,2]$; $a_6 = [-4,1,7,3]$; $a_7 = [-2,5,7,2]$; $a_8 = [-5,2,8,3]$; $a_9 = [-3,6,8,2]$;

1.4.12 а) Построим выпуклые оболочки множеств — они пересекаются, например, отрезок $(1,-2) - (2,3)$ пересекает $(1,0) - (2,-1)$. В то же время существует нелинейная функция $f(x,y) = x - y^2$, разделяющая эти множества. б) Аналогично.

1.4.13 Множества нулей и единиц булевской функции не будут линейно отделимы только в случае, когда каждое содержит по две точки и отрезки соединяющие пары точек пересекаются. Это будут функции $x \oplus y$ и $x \oplus y \oplus 1$.

1.4.14 Очевидно, эти множества не могут быть конечными. Пример бесконечных множеств легко строится, например $M_1 = \{(m,n) \in \mathbb{N}^2 \bigg| n > m\sqrt{2}\}$ и $M_2 = \{(m,n) \in \mathbb{N}^2 \bigg| n < m\sqrt{2}\}$. Разделяющая прямая $y = x\sqrt{2}$ не может быть построена, т.к. при обучении все веса будут иметь целые координаты.

1.4.15 По т. Новикова число изменений весового вектора не превосходит $D^2/\rho^2 = 100$. Будем циклически подавать последовательность $m_1, m_2, ..., m_7, m_1, m_2, ...m_7, m_1, ...$ Если в течение периода веса не изменились, то отделяющая гиперплоскость построена. Следовательно, надо будет подать не более $\frac{D^2}{\rho^2} \cdot |M| = 700$ элементов.

1.4.16 а) Небольшой перебор показывает, что прямые, соединяющие пары таких точек отстоят от начала координат не менее, чем на $1/\sqrt{5}$. А D, очевидно, не превосходит $\sqrt{2}$. b) Очевидно, $D \leqslant n\sqrt{2}$. Рассмотрим произвольный треугольник $\triangle ABO$, где $A, B \in M$, O — начало координат. Координаты этих точек целые, поэтому площадь треугольника $S(\triangle ABO) \geqslant \frac{1}{2}$. С другой стороны, $|AB| \leqslant 2\sqrt{2}n$. Из формулы для площади получим: $\rho = 2 \cdot S/|AB| \geqslant \frac{1}{2\sqrt{2}n}$. Отсюда $\frac{D^2}{\rho^2} \leqslant 2n^2 \cdot 8n^2 = 16n^4$.

1.4.17 а) Точки множества M_1 лежат выше прямой $y = \sqrt{2}x$, а точки M_2 — ниже. б) Решеается аналогично задаче 1.4.16 пункт б).

1.4.18 а) Можно показать (см. реш. 1.4.15), что на вход потребуется подать не более $m\frac{D^2}{\rho^2}$ элементов множества M. Значит столько будет операций типа i) и ii). А по Т. Новикова число изменений весового вектора не превосходит $m\frac{D^2}{\rho^2}$. Итого, общее число операций не превосходит $(2m+1)\frac{D^2}{\rho^2}$, а время $t \leqslant (2m+1)\frac{D^2}{\nu\rho^2}$.

Глава 2

Теория хранения и поиска информации

2.1. Реляционная алгебра

Основные обозначения и определения.

Реляционная алгебра

Если $D_1, D_2, \dots, D_n$ — некоторые множества, то подмножество $r \subseteq D_1 \times D_2 \times \cdots \times D_n$ называется *таблицей или отношением арности* n. Элементы $x = (x_1, x_2, \dots, x_n) \in r$ отношения r принято называть *кортежами*, компоненты отношения именовать символами $A_1, A_2, \dots, A_n$ и называть *атрибутами*, а область D_i значений атрибута A_i $(i = 1, 2, \dots, n)$ называть *доменом* и обозначать $D_i = dom(A_i)$. *Схемой отношения* R называется множество атрибутов $R = \{A_1, A_2, \dots, A_n\}$, и если $D_i = dom(A_i)$, $i = 1, 2, \dots, n$, то любое отношение $r \subseteq dom(R) = D_1 \times D_2 \times \cdots \times D_n$ называется *отношением со схемой* R; в этом случае также используют обозначение $r(R)$ или $r(A_1 A_2 \dots A_n)$.

Значения отдельных атрибутов обозначают $x_i = x(A_i)$. Для выделения подмножества атрибутов $S = \{A_{i_1}, \dots, A_{i_k}\} \subseteq R$, будет использоваться обозначение $x(S) = x(A_{i_1} \dots A_{i_k}) = (x_{i_1}, \dots, x_{i_k})$.

В теории реляционных баз данных рассматривают только конечные отношения (т.е. $|R| < \infty$) и отображают их в виде плоских таблиц со строками — кортежами и столбцами — атрибутами.

Реляционная база данных — это множество отношений.

Определим некоторые операции над отношениями.

Булевы операции

Если r и s отношения с одной и той же схемой R, т.е. подмножества $dom(R)$, то к ним могут быть применены обычные теоретико-множественные операции: объединение $\cup$, пересечение $\cap$, разность $\setminus$ (в реляционной алгебре ее принято обозначать знаком «–»), дополнение $\bar{\ }$.

Активным доменом атрибута A_i называется множество всех его значений, присутствующих в данном отношении, т.е. $adom(A_i, r) = \{d \in$

$\in dom(A_i) \,|\, \exists x \in r : x(A_i) = d\}$,

Активным доменом отношения r со схемой $R = (A_1, ..., A_n)$ называется $adom(R, r) = adom(A_1, r) \times \cdots \times adom(A_n, r)$.

Активным дополнением отношения R называется $\tilde{r} = adom(R, r) \setminus r$.

Пример 2.1. Пусть заданы $r_1 = \left(\begin{array}{cc} A & B \\ \hline a_1 & b_1 \\ a_2 & b_1 \end{array}\right)$ и $r_2 = \left(\begin{array}{cc} A & B \\ \hline a_1 & b_2 \\ a_2 & b_1 \\ a_3 & b_2 \end{array}\right)$. Тогда

$r_1 \cup r_2 = \left(\begin{array}{cc} A & B \\ \hline a_1 & b_1 \\ a_1 & b_2 \\ a_2 & b_1 \\ a_3 & b_2 \end{array}\right)$, $r_1 \cap r_2 = \left(\begin{array}{cc} A & B \\ \hline a_1 & b_2 \end{array}\right)$, $r_1 - r_2 = \left(\begin{array}{cc} A & B \\ \hline a_1 & b_1 \end{array}\right)$ и $\tilde{r}_2 =$

$= \left(\begin{array}{cc} A & B \\ \hline a_1 & b_1 \\ a_2 & b_2 \\ a_3 & b_1 \end{array}\right)$

Операторы выбора

Пусть r — отношение со схемой R, пусть $A \in R$ — атрибут и константа $a \in dom(A)$. Тогда $\sigma_{A=a}(R)$ — обозначение операции выбора в r кортежей, в которых значение A равно a, т.е.

$$\sigma_{A=a}(r) = \{x \in r : x(A) = a\}.$$

Если на $dom(A)$ задано бинарное отношение $\rho \subseteq dom(A) \times dom(A)$ (пишем $a\rho b$, если $(a, b) \in \rho$), то выбор можно осуществлять с помощью этого отношения

$$\sigma_{A\rho a}(r) = \{x \in r : \, x(A)\rho a\}.$$

Пусть на $dom(R)$ задан предикат $\rho(x)$. Тогда оператор $\sigma_\rho(r)$ выбирает все кортежи, на которых этот предикат выполняется, т.е. $\sigma_\rho(r) = \{x \in r \,|\, \rho(x)\}$.

Пример 2.2. Пусть задано отношение $r = \left(\begin{array}{cc} A & B \\ \hline 1 & 1 \\ 1 & 2 \\ 3 & 2 \end{array}\right)$. Тогда $\sigma_{B>1}(r) =$

$= \left(\begin{array}{cc} A & B \\ \hline 1 & 2 \\ 3 & 2 \end{array}\right)$,

$\sigma_{A \geqslant B}(r) = \left(\begin{array}{cc} A & B \\ \hline 1 & 1 \\ 3 & 2 \end{array}\right)$.

Оператор проекции

Оператор *проекции* выбирает подмножество атрибутов в отношении.

Пусть r — отношение со схемой R и $S \subseteq R$. Проекция r на S, записанная как $\pi_S(r)$, есть отношение,

$$\pi_S(r) = \{x(S) \,|\, x \in r\}.$$

Пример 2.3. Пусть задано отношение $r(A,B,C) = \left(\begin{array}{ccc} A & B & C \\ \hline 1 & 1 & 2 \\ 1 & 1 & 3 \\ 1 & 2 & 2 \\ 3 & 2 & 2 \end{array}\right)$. Тогда $\pi_{AC}(r) = \left(\begin{array}{cc} A & C \\ \hline 1 & 2 \\ 1 & 3 \\ 3 & 2 \end{array}\right)$

Оператор соединения

Соединение — это бинарный оператор для комбинирования двух отношений по всем их общим атрибутам. Пусть заданы два отношения $r_1(R_1)$ и $r_2(R_2)$. *Соединением* отношений r_1 и r_2, записываемым как $r_1 \bowtie r_2$, является отношение $r(R) = \{x \in dom(R_1 \cup R_2) \,|\, x(R_1) \in r_1, x(R_2) \in r_2\}$. Таким образом, каждый кортеж в R является комбинацией корежа из R_1 и кортежа из R_2, у которых равны значения общих атрибутов из $(R_1 \cap R_2)$.

Если $R_1 \cap R_2 = \emptyset$, то $r_1 \bowtie r_2$ — это декартово произведение $r_1 \times r_2$.

Пример 2.4. Пусть R_1 и R_2 таковы:

$R_1(A,$	$B)$
a_1	b_1
a_2	b_1

$R_2(B,$	$C)$
b_1	c_1
b_2	c_1
b_2	c_2

Найти $R_1 \bowtie R_2$.

Решение. $R_1 \bowtie R_2$ будет:

$R_1 \bowtie R_2(A,$	$B,$	$C)$
a_1	b_1	c_1
a_2	b_1	c_1

Переименование атрибутов

Если R — схема отношения, $A \in R$, $B \notin R$ — атрибуты, и $dom(A) = dom(B)$, то переименование атрибута A на B состоит в замене в множестве R элемента A на B. Схема результата $R' = (R \setminus A) \cup B$. Более формально

$$\rho_{A \to B}(R) = \{x' \in dom(R') \,|\, \exists x \in r \,:\, x(R \setminus A) = x'(R \setminus A), x(A) = x'(B)\}$$

Ключи отношений

Суперключом отношения r со схемой R называют подмножество атрибутов $S \subset R$, для которого не существует двух кортежей $x, x' \in r$, таких, что $x(S) = x'(S)$.

Ключом (потенциальным) отношения r со схемой R называют суперключ, обладающий свойством минимальности, т.е. S — суперключ и не существует суперключа S', такого, что $S' \subset S$ и $S' \neq S$.

Упражнения.

Задача 2.1.1. Пусть заданы отношения $r(ABC)$ и $s(BCD)$. Какова схема отношения $E = \pi_A\big(\sigma_{B=b}(s)\big) \bowtie \pi_B\big(\pi_{BC}(r) - \pi_{BC}(s)\big)$?

Задача 2.1.2. Пусть заданы отношения $r(R)$ и $s(S)$ со схемами $R = \{A, B, C\}$ и $S = \{B, C, D\}$, соответственно. Пусть $a \in dom(A)$ и $b \in dom(B)$ Какие из выражений: a) $\sigma_{A=a,B=b}(s)$; b) $\pi_B(r \bowtie s)$; c) $\pi_A(r) \bowtie \pi_D(s)$; d) $\pi_{AB}(r) \cup \pi_{BC}(s)$; e) $\pi_{BC}(r) \cap \pi_{BC}(r \bowtie s)$ f) $\sigma_{A=a,B=b}(r \bowtie s) - \sigma_{A=a,B=b}(r)$; имеют смысл?

Задача 2.1.3. Пусть X — подмножество R, и r, s — отношения со схемой R. Докажите или опровергните равенства: a) $\pi_X(r - s) = \pi_X(r) - \underline{\pi_X(s)}$; b) $\pi_X(r \cap s) = \pi_X(r) \cap \pi_X(s)$; c) $\pi_X(r \cup s) = \pi_X(r) \cup \pi_X(s)$; d) $\pi_X(\bar{r}) = \overline{\pi_X(r)}$.

Задача 2.1.4. r и r' — отношения со схемой R, где $X \subseteq R$. Пусть s, t — отношения со схемами S и T. Всегда ли верно, что
a) $\pi_X(r \cap r') = \pi_X(r) \cap \pi_X(r')$;
b) $(r \cap r') \bowtie s = (r \bowtie s) \cap (r' \bowtie s)$;
c) $(r \bowtie s) \bowtie t = r \bowtie (s \bowtie t)$?

Задача 2.1.5. Пусть r — отношение со схемой R, где $dom(R) = \mathbb{R}$ (множество действительных чисел). Выписать выражение, содержащее только операции реляционной алгебры, находящее наибольшее число в r (точнее отношение, содержащее только этот элемент).

Задача 2.1.6. Пусть r — список студентов III курса со следующими атрибутами: G — группа, N — ФИО, S — пол, Y — год рождения, A науч.рук., $M_1, ..., M_5$ — отметки за последнюю сессию, C — тема курсовой.

Выписать выражения реляционной алгебры для нахождения следующих отношений: a) Список тем курсовых у данного научного руководителя $a_0 \in dom(A)$. b) Список студентов, имеющих хотя бы одну задолженность; c) Список студентов, интересующих военкомат[1]. d) Пусть отношение s имеет один атрибут A и задает список всех сотрудников кафедры. Найти список сотрудников кафедры, не имеющих ни одного студента; e)* Список сотрудников кафедры, имеющих более одного студента;

Задача 2.1.7. В учебной части MIT[2] набор таблиц другой:
$r_1(CSG)$ `-- Course-StudentID-Grade`
$r_2(SNPE)$ `-- StudentID-Name-Phone-Email`
$r_3(CDHR)$ `-- Course-Day-Hour-Room`
a) Получить список телефонов студентов, которые посещают курс «A.I.» (Artificial Intelligence).
b) В аудитории $a \in Dom(R)$ в понедельник будет идти ремонт. Надо разослать сообщение о переносе занятий в другую аудиторию. Найдите адреса всех студентов, у которых в понедельник (в любое время) есть занятия в этой аудитории.
c) По заданному ID студента ($s_0 \in dom(S)$) определить, в какой аудитории у него в понедельник первая пара (т.е. $H = 1$).

Задача 2.1.8. Дано r — расписание авиарейсов Нью-Йорк–Москва с атрибутами A_1 — номер рейса, A_2 — время вылета, A_3 — время прибытия. и s — расписание электричек Москва—Васюки с атрибутами B_1 — номер поезда, B_2 — время отправления, B_3 — время прибытия. a) Составить список всех

[1] юноши призывного возраста, не сдавшие сессию
[2] У каждого студента свой набор курсов, которые надо посещать

возможных маршрутов из Нью-Йорка в Васюки (поезд должен отправляться позже, чем прилетит самолет); б) Выбрать из пункта а) те маршруты, в которых человек садится на ближайшую электричку (а не на любую после прилета самолета).

Задача 2.1.9. В отделе кадров предприятия в базе есть две таблицы: $r(NPS)$, где атрибуты N — ФИО (Name), P — Должность (Position), W — Зарплата(Wages) и $s(BS)$ с атрибутами B — Начальник (Boss), S — Подчиненный (Subordinate) (причем $dom(N) = dom(B) = dom(S)$). a) Как получить список рядовых сотрудников (т.е. тех, у кого нет подчиненных)? b) Как получить список начальников, у которых все непосредственные подчиненные получают зарплату не менее заданного w? c) Как получить список менеджеров среднего звена, т.е. начальников, у которых в подчинении только другие начальники (и нет рядовых сотрудников)?

Задача 2.1.10. В салоне красоты у секретаря есть таблицы:

- $r(CNP)$ с атрибутами C — ID клиента, N — ФИО, P — телефон;
- $s(MN'S)$ с атрибутами M — ID сотрудника, N' — ФИО сотрудника, S — специализация;
- $t(DTCM)$ с атрибутами D — дата, T — время, C — ID клиента, M — ID сотрудника.

a) Получить список ФИО всех маникюрщиц, которые вчера ($d_y \in dom(D)$) работали после 14-00. b) Парикмахер Анжела Пуговкина (А.П. $\in dom(N')$) заболела, надо обзвонить ее клиентов. Получить список клиентов (с телефонами), которых она должна была сегодня ($d_t \in dom(D)$) обслуживать

Задача 2.1.11. Пусть r — отношение со схемой (N, P), где N — название товара, а P — цена. Выписать оператор реляционной алгебры, который a) По названию товара $n_0 \in dom(N)$ находит названия всех товаров не дешевле этого товара n_0; b) Находит самый дорогой товар (если с такой ценой есть несколько товаров, то находит их все). c) Находит самый дешевый товар из тех, что дороже, чем n_0.

Задача 2.1.12. Пусть даны отношения $r(AN)$ и $s(NB)$, пусть $n_0 \in Dom(N)$. a) Докажите, что $\sigma_{N=n_0}(r \bowtie s) = \sigma_{N=n_0}(r) \bowtie \sigma_{N=n_0}(s)$; b) Пусть $|adom(N)| = L$ и для каждого $n \in adom(N)$ значение атрибута $N = n$ встречается в отношении r ровно M раз, а в s — ровно K раз. Сравните количество памяти требуемой при указанных в пункте a) схемах вычислений. Для упрощения можно считать, что каждый атрибут каждого кортежа занимает одну ячейку памяти, а затратами на хранение структуры базы можно пренебречь.

Определение. Пусть $r(R)$ и $s(S)$ — два отношения, для которых $R \cap S = \emptyset$. Пусть задано бинарное отношение $\theta : R \times S \to \{0, 1\}$. Тогда оператор θ-соединения задается как $t(AB) = r \bowtie_\theta s = \{t = (t_r, t_s) : t_r \theta t_s, t_r \in r, t_s \in s\}$. Т.е. результатом является множество пар, для которых указанное отношение θ истинно.

Задача 2.1.13. Выразить оператор θ-соединения через стандартные операции реляционной алгебры.

Определение. Пусть $r(R)$ и $s(S)$ — отношения со схемами $S \subseteq R$. Тогда можно определить операцию *реляционного деления*: $r \div s = \{x \in dom(R \setminus \setminus S) : \forall y \in s\big([x,y] \in r\big)\}$.

Задача 2.1.14. Пусть даны отношения $r(R)$ и $s(S)$, причем $R \cup S = \emptyset$. Докажите, что $(r \bowtie s) \div s = r$.

Задача 2.1.15. Докажите, что a) из $s \subseteq s'$ следует $(r \div s') \subseteq (r \div s)$; b) Докажите, что обратное неверно, т.е. из $(r \div s') \subseteq (r \div s)$ не следует, что $s \subseteq s'$;

Задача 2.1.16. Пусть $r(R)$ и $s(S)$ — отношения со схемами $S \subseteq R$ и $R' = R \setminus S$. Докажите, что a) $r \div s = \pi_{R'}(r) - \pi_{R'}\big((\pi_{R'}(r) \bowtie s) - r\big)$; b) Докажите, что $r \div s = \bigcap\limits_{t \in s} \pi_{R'}(\sigma_{S=t}(r))$.

Задача 2.1.17. Для данного отношения $q(RS)$ известно, что $|q| = |\pi_R(q)| \cdot |\pi_S(q)|$. a) Докажите, что $q = \pi_R(q) \bowtie \pi_S(q)$. b) Пусть задано отношение $r(ABC)$, найдите необходимые и достаточные условия того, что $r = \pi_{AB}(r) \bowtie \pi_{BC}(r)$.

Задача 2.1.18. Приведите пример тернарного отношения (т.е. отношения с 3 атрибутами), которое не может быть представлено в виде конъюнкции бинарных отношений без введения дополнительных атрибутов. *Подумайте, какая операция реляционной алгебры соответствует конъюнкции бинарных отношений?*

Задача 2.1.19. Пусть r и s — отношения со схемой R и ключом $K \subset R$. Какие из следующих отношений обязательно должны иметь суперключ K: a) $r \cup s$; b) $r \cap s$; c) $r - s$; d) $\tilde{r}$; e) $\pi_K(r)$; f) $r \bowtie s$.

Задача 2.1.20. Привести пример отношений $r(R)$ и $s(R)$ с ключом $K \subset R$, таких, что K не является ключом ни для $r \cap s$ ни для $r - s$ (хотя и является суперключом).

Ответы и указания.

2.1.1 $E(AB)$.

2.1.2 a) нет; b) да; c) да; d) нет; e) да; f) нет.

2.1.3 a,b) Опровергается примером $r = \left(\begin{array}{cc} A & B \\ \hline a & b \end{array}\right)$, $s = \left(\begin{array}{cc} A & B \\ \hline a & b' \end{array}\right)$. c) Верно. d) Опровергается примером $r = \left(\begin{array}{cc} A & B \\ \hline a & b \\ a_1 & b_1 \end{array}\right)$.

2.1.4 **Решение.** a) Неверно, легко построить контрпример, например $r = \{(x_1, y_1)\}$, $r' = \{(x_1, y_2)\}$, где $y_1 \neq y_2$. b) Верно, достаточно проверить для r со схемой (X, Y) и s со схемой (Y, Z)

2.1.5 $r - \pi_R\left(\sigma_{R<R'}\big(r \bowtie \rho_{R \to R'}(r)\big)\right)$.

2.1.6 a) $r_a = \pi_C(\sigma_{A=a_0}(r))$; b) $r_b = \pi_N(\sigma_{(M_1=2)\vee\ldots\vee(M_5=2)}(r))$; c) $r_c = \pi_N(\sigma_{(S=\text{Муж.})\vee(Y\leqslant 1995)}(r))\cap r_b$, где r_b — решение предыдущего пункта (Здесь 1995 выбрано как 2013-18); d) $r_d = s - \pi_A(r)$; e) Пусть $r_1 = \pi_{AN}(R)$, $r_2 = r_1 \bowtie \rho_{N\leftarrow N'}(r_1)$. Тогда $r_e = \pi_A(\sigma_{N\neq N'}(r_2))$ — решение.

2.1.7 a) Пусть $s = \pi_{CS}(r_1) \bowtie \pi_{SP}(r_2)$, тогда $p = \pi_P(\sigma_{C=\text{«A.I.»}}(s))$ — искомый список. b) Пусть $s_1 = \pi_C(\sigma_{D=Mon.}(r_3))$, $s_2 = \pi_{CS}(r_1) \bowtie \pi_{SE}(r_2)$. Тогда $\pi_E(s_1 \bowtie s_2)$ — искомый список адресов. c) Пусть $s = \pi_{CS}(r_1) \bowtie r_3$, тогда $\pi_R(\sigma_{D=Mon.,H=1}(s))$ — список аудиторий (их может быть несколько).

2.1.8 a) $r_a = \sigma_{A_3<B_2}(r \bowtie s)$; b) Пусть $r' = \rho_{B_2\to B'}(\pi_B(s))$, $r'' = \sigma_{(A_3<B')\&(B'<B_2)}(r_a \bowtie r')$. Тогда искомый список $r_a - \pi_{A_1A_2A_3B_1B_2B_3}(r'')$.

2.1.9 a) $r_a = \pi_S(s) - \rho_{B\to S}(\pi_B(s))$; b) Пусть $t = s \bowtie (\rho_{N\to S}(r))$, тогда ответ — $r_b = \pi_B(\sigma_{W\geqslant w}(t)) - \pi_B(\sigma_{W<w}(t))$. c) $\pi_B(s) - \pi_B(s \bowtie r_a)$, где r_a — ответ в пункте a).

2.1.10 a) $r' = \pi_M(\sigma_{D=d_y\&T>14:00}(t))$, тогда $r_a = \pi_{N'}(r' \bowtie \sigma_{S=\text{«маникюр»}}(s))$ — ответ. b) $r'' = (\sigma_{N'=\text{А.П.}}(s) \bowtie \sigma_{D=d_t}(t)$, тогда $r_b = \pi_{CP}(\pi_C(r'') \bowtie r)$ — ответ.

2.1.11 a) Пусть $p = \rho_{P\to P'}(\pi_P(\sigma_{N=n_0}(r)))$, тогда $r_a = \pi_N(\sigma_{P\geqslant P'}(r \bowtie p))$. b) Пусть $r' = r \bowtie (\rho_{N\to N'}(\pi_N(r)))$, тогда $r_b = \pi_N(r) - \pi_N(\sigma_{P<P'}(r'))$ — ответ. Альтеррнативный вариант — использовать операцию реляционного деления: $r_{b'} = \sigma_{N\geqslant N'}(r') \div \pi_{N'}(r')$. c) $r'' = \pi_{N',P'}(\sigma_{N=n_0,P'>P}(r \bowtie \rho_{N\to N',P\to P'}(r)))$. Далее — как в пункте b).

2.1.12 a) В обоих случаях получается множество кортежей вида (a_n, n, b_n), где $(a_n, n) \in r$, $(n, b_n) \in s$. b) Для хранения промежуточных результатов в первом случае (левая часть) требуется KLM, а во втором — $K + M$ ячеек. Для хранения ответа требуется KM ячеек — в обоих случаях.

2.1.13 $r \bowtie_\theta s = \pi_{R\theta S}(r \bowtie s)$.

2.1.14 Если $R \cup S = \emptyset$, то $r \bowtie s = r \times s$ и утверждение вытекает из определения операции реляционнного деления.

2.1.15 a) Рассмотрим произвольный $x \in r \div s$. Тогда при всех $y \in s$ кортеж (x, y) входит в r, т.е. при всех $y \in s'$ это тоже выполнено, следовательно, $x \in r \div s'$. b) Приведем контрпример: $r(AB) = \begin{pmatrix} a & 1 \\ a & 2 \\ b & 2 \end{pmatrix}$; $s(B) = (2)$ и $s'(B) = (1)$.

2.1.16 a) Если для всех $b \in s$ кортеж (a_0, b) содержится в r, то в $(\pi_{R'}(r) \bowtie s) - r$ не будет кортежей, содержащих a_0, следовательно, они останутся в $\pi_{R'}(r) - \pi_{R'}\big((\pi_{R'}(r) \bowtie s) - r\big)$. И наоборот, если при некотором b_0 кортеж $(a_0, b_0) \notin r$, то он будет в $(\pi_{R'}(r) \bowtie s) - r$ и, следовательно, в $\pi_{R'}(r) - \pi_{R'}\big((\pi_{R'}(r) \bowtie s) - r\big)$ не будет a_0. b) Если $x \in \pi_{R'}(\sigma_{S=t}(r))$, то кортеж (x, t) принадлежит r. Следовательно, пересечение таких множеств дает как раз определение операции реляционного деления.

2.1.17 а) Очевидно, что $q \subset \pi_R(q) \times \pi_S(q) = \pi_R(q) \bowtie \pi_S(q)$. Докажем обратное включение. Предположим, что найдутся $x \in \pi_R(q)$ и $y \in \pi_S(q)$, такие, что $(x, y) \notin q$. Тогда отношение q содержит менее $|\pi_R(q) \times \pi_S(q)| = |\pi_R(q)| \cdot |\pi_S(q)|$ кортежей, что противоречит условию задачи.

2.1.18 $r = \left(\begin{array}{ccc} A & B & C \\ \hline a_2 & b_1 & c_1 \\ a_1 & b_2 & c_1 \\ a_1 & b_1 & c_2 \end{array}\right)$. Если бы это отношение было конъюнкцией бинарных отношений, то там должен был бы присутствовать кортеж (a_1, b_1, c_1).

2.1.19 а) нет; b) да; c) да; d) нет; e) да; f) да.

2.1.20 $r = \left(\begin{array}{cc} A & B \\ \hline 1 & a \\ 1 & b \\ 2 & a \\ 2 & b \end{array}\right)$ и $s = \left(\begin{array}{cc} A & B \\ \hline 1 & a \\ 1 & b \\ 3 & a \\ 3 & b \end{array}\right)$

2.2. Информационно–графовый подход.

Основные обозначения и определения.

Пусть задано некоторое множество Y, будем называть его множеством объектов поиска или *множеством записей*, а элементы $y \in Y$, будем называть, *записями*.

Далее в задачах поиска всегда имеется *множество запросов*. будем обозначать его через X.

На декартовом произведении $X \times Y$ имеется бинарное отношение, которое позволяет устанавливать, когда запись из Y удовлетворяет запросу из X. Это отношение будем называть *отношением поиска*.

Тройку $S = \langle X, Y, \rho \rangle$, где X — множество запросов, Y — множество записей, ρ — отношение поиска, заданное на $X \times Y$, будем называть *типом задач информационного поиска* или сокращенно *типом ЗИП*.

Тройку $I = \langle X, V, \rho \rangle$, где X — множество запросов; V — некоторое конечное подмножество множества Y, в дальнейшем называемое *библиотекой*; ρ — отношение поиска, заданное на $X \times Y$, будем называть *задачей информационного поиска* (ЗИП) типа $S = \langle X, Y, \rho \rangle$. Содержательно будем считать, что задача $I = \langle X, V, \rho \rangle$ состоит в перечислении для произвольно взятого запроса $x \in X$ всех тех и только тех записей из V , которые находятся в отношении ρ с запросом x, то есть удовлетворяют запросу x.

Рассмотрим произвольную ЗИП $I = \langle X, V, \rho \rangle$. Алгоритм поиска решает ЗИП, если на любой запрос из множества запросов X он выдает все те и только те записи из V, которые удовлетворяют запросу. Возьмем произвольную запись $y \in Y$. Для нее можно ввести *характеристическую функцию*

$$\chi_{y,\rho}(x) = \begin{cases} 1, & \text{если } x\rho y \\ 0 & \text{в противном случае} \end{cases},$$

то есть она равна 1 на тех и только тех запросах, которым удовлетворяет запись y. Тогда можно сказать, что алгоритм, решающий ЗИП $I = \langle X, V, \rho \rangle$, где $V = \{y_1, \ldots, y_k\}$, — ни что иное как алгоритм, реализующий систему функций $\{\chi_{y_1,\rho}, \ldots, \chi_{y_k,\rho}\}$.

Рассмотрим ориетированную сеть, в ней выделен полюс, который называется ***корнем***, и каждая из характеристических функций записей $\chi_{y_i,\rho}$ $(i = \overline{1,k})$ реализуется как функция проводимости между корнем и своим полюсом, который отметим приписыванием ему записи y_i. Алгоритм поиска: считаем, что в начальный момент все вершины сети, кроме корня, неотмеченные, а некоторое упорядоченное множество вершин сети, которое назовем *множеством активных вершин*, содержит только корень сети. На каждом очередном шаге делаем следующее. Если множество активных вершин не пусто, то выбираем первую активную вершину и удаляем ее из множества активных вершин. Если выбранная вершина является полюсом, то соответствующую ей запись включаем в ответ на запрос x. Просматриваем в некотором порядке все ребра, исходящие из выбранной вершины, и если предикат, приписанный просматриваемому ребру, на запросе x принимает значение 1 и конец просматриваемого ребра неотмеченная вершина, то отмечаем конец просматриваемого ребра и включаем его в множество активных вершин (если мы включим его в начало множества активных вершин, то получим алгоритм обхода "сначала вглубь а если в конец — то "сначала вширь"). Алгоритм завершает работу в тот момент, когда множество активных вершин окажется пустым.

Пусть M — некоторое конечное множество. Через $|M|$ обозначим число элементов во множестве M, называемое *мощностью множества* M.

Через $\{\overline{1,m}\}$ договоримся обозначать множество $\{1, 2, \ldots, m\}$.

Некоторые оценки мы будем приводить с точностью до главного члена, поэтому введем обозначения, обычно принятые при описании асимптотических оценок.

Будем писать $\alpha(n) = \bar{o}(1)$, если $\lim\limits_{n\to\infty} \alpha(n) = 0$; $A(n) = \bar{o}(B(n))$, если $A(n) = B(n) \cdot \bar{o}(1)$. Скажем, что $A(n)$ ***асимптотически не превосходит*** $B(n)$ при $n \to \infty$ и обозначим $A \lesssim B$, если существует $\alpha(n) = \bar{o}(1)$ такое, что начиная с некоторого номера n_0, $A(n) \leqslant (1 + \alpha(n)) \cdot B(n)$. Если $A \lesssim B$ и $B \lesssim A$, то будем говорить, что A и B ***асимптотически равны*** при $n \to \to \infty$ и обозначать $A \sim B$. Будем писать $A \overset{<}{\asymp} B$, если существует такая положительная константа c, что, начиная с некоторого номера n_0, $A(n) \leqslant \leqslant c \cdot B(n)$. Если $A \overset{<}{\asymp} B$ и $B \overset{<}{\asymp} A$, то будем говорить, что A и B ***равны по порядку*** при $n \to \infty$ и обозначать $A \asymp B$ или $A = \underline{O}(B)$.

Через $\binom{n}{k}$ будем обозначать *число сочетаний из* n *элементов по* k. Если r — действительное число, то через $[r]$ будем обозначать максимальное целое, не превышающее r, а через $]r[$ — минимальное целое, не меньшее, чем r. Значок $\overset{\text{def}}{=}$ будем понимать как "по определению равно". Математическое ожидание будем обозначать значком $\mathbf{M}$, а значок $\mathbf{M}_x$ будем понимать как среднее значение при вариации переменной x.

Договоримся также о теоретико-графовой терминологии.

Пусть нам дан ориентированный граф. В ориентированном ребре (α, β) вершину α будем называть *началом ребра*, а β — *концом*. Скажем, что ориентированное ребро графа *исходит из вершины* β (*входит в вершину* β), если β — начало (конец) данного ребра. Скажем, что ребро *инцидентно* вершине, если эта вершина является одним из концов данного ребра. *Полустепенью исхода* (*захода*) вершины графа назовем число ребер, исходящих из данной вершины (входящих в данную вершину). *Степенью инцидентности вершины* назовем число инцидентных ей ребер. Вершину графа назовем *концевой*, если полустепень ее исхода равна 0. Остальные вершины графа назовем *внутренними*.

Последовательность ориентированных ребер графа

$$(\alpha_1, \alpha_2), (\alpha_2, \alpha_3), \ldots, (\alpha_{m-1}, \alpha_m)$$

назовем *ориентированной цепью* от вершины α_1 к вершине α_m.

Если f — одноместный предикат, определенный на X , то есть $f : X \to \{0, 1\}$, то множество $N_f = \{x \in X : f(x) = 1\}$ назовем *характеристическим множеством предиката* f.

Множество $O(y, \rho) = \{x \in X : x\rho y\}$ назовем *тенью записи* $y \in Y$.

Функцию $\chi_{y,\rho} : X \to \{0, 1\}$ такую, что $N_{\chi_{y,\rho}} = O(y, \rho)$, назовем *характеристической функцией записи* y.

В формальном определении понятия ИГ используются 4 множества:

- множество запросов X;
- множество записей Y;
- *множество F одноместных предикатов*, заданных на множестве X;
- *множество G одноместных переключателей*, заданных на множестве X (*переключатели* — это функции, область значений которых является начальным отрезком натурального ряда).

Пару $\mathcal{F} = \langle F, G\rangle$ будем называть *базовым множеством*.

Определение понятия ИГ разбивается на два шага. На первом шаге раскрывается структурная (схемная) часть этого понятия, на втором — функциональная.

Определение ИГ с точки зрения его структуры.

Пусть нам дана ориентированная многополюсная сеть.

Выделим в ней один полюс и назовем его *корнем*, а остальные полюса назовем *листьями*.

Выделим в сети некоторые вершины и назовем их *точками переключения* (полюса могут быть точками переключения).

Если β — вершина сети, то через ψ_β обозначим *полустепень исхода* вершины β.

Каждой точке переключения β сопоставим некий символ из G. Это соответствие назовем *нагрузкой точек переключения*.

Для каждой точки переключения β ребрам, из нее исходящим, поставим во взаимно однозначное соответствие числа из множества $\{\overline{1, \psi_\beta}\}$. Эти ребра назовем *переключательными*, а это соответствие — *нагрузкой переключательных ребер*.

Ребра, не являющиеся переключательными, назовем *предикатными*.

Каждому предикатному ребру сети сопоставим некоторый символ из множества F. Это соответствие назовем *нагрузкой предикатных ребер*.

Сопоставим каждому листу сети некоторую запись из множества Y. Это соответствие назовем *нагрузкой листьев.*

Полученную нагруженную сеть назовем *информационным графом* над базовым множеством $\mathcal{F} = \langle F, G \rangle$.

Определение функционирования ИГ.

Скажем, что предикатное ребро проводит запрос $x \in X$, если предикат, приписанный этому ребру, принимает значение 1 на запросе x; переключательное ребро, которому приписан номер n, проводит запрос $x \in X$, если переключатель, приписанный началу этого ребра, принимает значение n на запросе x; ориентированная цепочка ребер проводит запрос $x \in X$, если каждое ребро цепочки проводит запрос x; запрос $x \in X$ проходит в вершину β ИГ, если существует ориентированная цепочка, ведущая из корня в вершину β, которая проводит запрос x; запись y, приписанная листу α, попадает в ответ ИГ на запрос $x \in X$, если запрос x проходит в лист α. *Ответом ИГ U на запрос x* назовем множество записей, попавших в ответ ИГ на запрос x, и обозначим его $\mathcal{J}_U(x)$. Эту функцию $\mathcal{J}_U(x) : X \to 2^Y$ будем считать результатом функционирования ИГ U и называть *функцией ответа ИГ U*.

Понятие ИГ полностью определено.

Пример 2.5. Проиллюстрируем приведенное определение на примере одномерной задачи интервального поиска. В этом случае 4 множества, определяющие ИГ, имеют вид:

- множество запросов $X_{int} = \{(u, v) : 0 \leqslant u \leqslant v \leqslant 1\}$;
- множество записей $Y_{int} = [0, 1]$;
- множество предикатов
  $F = F_1 \bigcup F_2$, где $F_1 = \{f^1_{\leqslant,a} : a \in [0, 1]\}$, $F_2 = \{f^2_{\geq,a} : a \in [0, 1]\}$,

$$f^1_{\leqslant,a}(u, v) = \begin{cases} 1, & \text{если } u \leqslant a \\ 0, & \text{если } u > a \end{cases}, \qquad (2.1)$$

$$f^2_{\geq,a}(u, v) = \begin{cases} 1, & \text{если } v \geq a \\ 0, & \text{если } v < a \end{cases}, \qquad (2.2)$$

- множество переключателей
  $G = G_1 \bigcup G_2 \bigcup G_3$, где $G_1 = \{g_{\cdot,m} : m \in \mathbf{N}\}$, $G_2 = \{g_{-,m} : m \in \mathbf{N}\}$, $G_3 = \{g_{\leqslant,a} : a \in (0, 1]\}$,

$$g_{\cdot,m}(u, v) = \max(1,]u \cdot m[), \qquad (2.3)$$

$$g_{-,m}(u, v) = \begin{cases} 1, & \text{если } v - u < 1/m \\ 2, & \text{если } v - u \geq 1/m \end{cases}, \qquad (2.4)$$

$$g_{\leqslant,a}(u, v) = \begin{cases} 1, & \text{если } u \leqslant a \\ 2, & \text{если } u > a \end{cases}. \qquad (2.5)$$

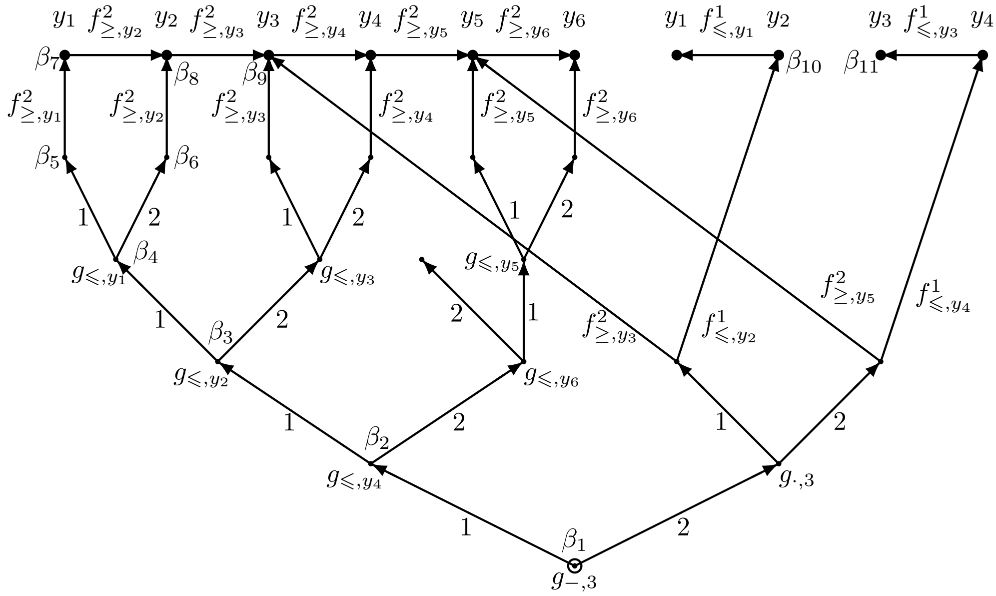


Рис. 2.1: Решение одномерной задачи интервального поиска

В информационном графе, приведенном на рисунке 2.1, корень изображен полым кружком. Листья изображены жирными точками, а записи, приписанные листьям, — это символы y с индексами. На рисунке имеется 8 переключательных вершин (им приписаны символы g с индексами) и 17 предикатных ребра (им приписаны символы f с индексами).

Если n — натуральное число, а $g(x)$ — некий переключатель, то через $\xi^n_g(x)$ обозначим предикат, определенный на X , такой, что

$$N_{\xi^n_g} = \{x \in X : g(x) = n\}.$$

Обозначим

$$\widehat{G} = \{\xi^n_g : g \in G, n \in \mathbf{N}\},$$

где $\mathbf{N}$ — множество натуральных чисел.

Если c — ребро ИГ, то через $[c]$ обозначим его нагрузку.

В соответствии с приведенными выше определениями введем функции проводимости.

Проводимостью ребра (α,β) назовем предикат, равный $[(\alpha,\beta)]$, если ребро предикатное, и $\xi_g^{[(\alpha,\beta)]}$, если ребро переключательное, где g — переключатель, соответствующий вершине α.

Проводимостью ориентированной цепи назовем конъюнкцию проводимостей ребер цепи.

Если зафиксировать запрос x, то цепь, проводимость которой на запросе x равна 1, назовем *проводящей цепью на запросе* x.

В ИГ по аналогии с контактными схемами введем для каждой пары вершин α и β *функцию проводимости* $f_{\alpha\beta}$ от вершины α к вершине β следующим образом:

- если $\alpha = \beta$, то $f_{\alpha\beta}(x) \equiv 1$ $(x \in X)$;

- если $\alpha \neq \beta$ и в ИГ не существует ориентированных цепей от α к β, то $f_{\alpha\beta}(x) \equiv 0$;
- если $\alpha \neq \beta$ и множество ориентированных цепей от α к β не пусто, то $f_{\alpha\beta}(x)$ равно дизъюнкции проводимостей всех ориентированных цепей от α к β.

Функцию проводимости от корня ИГ к некоторой вершине β ИГ назовем ***функцией фильтра вершины*** β и обозначим $\varphi_\beta(x)$.

Пример 2.6. Чему равна проводимость ребра (β_2, β_3) информационного графа, приведенного на рис. 2.1?

Решение. Проводимость ребра (β_2, β_3) равна $f_2(u,v) = \xi^1_{g_\leqslant, y_4}(u,v) = \begin{cases} 1, & \text{если } u \leqslant y_4 \\ 0, & \text{если } u > y_4 \end{cases}$.

Пример 2.7. Чему равна проводимость цепи информационного графа, приведенного на рис. 2.1, ведущей из вершины β_1 в вершину β_4, в предположении, что $y_2 < y_4$?

Решение. Проводимость ребра (β_1, β_2) равна $f_1(u,v) = \xi^1_{g_-, 3}(u,v) = \begin{cases} 1, & \text{если } v-u < 1/3 \\ 0, & \text{если } v-u \geq 1/3 \end{cases}$. Проводимость ребра (β_2, β_3) вычислена в предыдущем примере. Проводимость ребра (β_2, β_3) равна $f_3(u,v) = \xi^1_{g_\leqslant, y_2}(u,v) = \begin{cases} 1, & \text{если } u \leqslant y_2 \\ 0, & \text{если } u > y_2 \end{cases}$. Проводимость цепи, ведущей из вершины β_1 в вершину β_4, равна $f_4(u,v) = f_1(u,v) \& f_2(u,v) \& f_3(u,v) = \begin{cases} 1, & \text{если } v-u < 1/3 \text{ и } u \leqslant y_2 \\ 0, & \text{иначе} \end{cases}$.

Пример 2.8. Чему равна функция фильтра вершины β_8 информационного графа, приведенного на рис. 2.1, в предположении, что $y_1 < y_2 < y_4$?

Решение. В вершину β_8 ведут 2 цепи: $(\beta_1, \beta_2), (\beta_2, \beta_3), (\beta_3, \beta_4), (\beta_4, \beta_5), (\beta_5, \beta_7), (\beta_7, \beta_8)$ и $(\beta_1, \beta_2), (\beta_2, \beta_3), (\beta_3, \beta_4), (\beta_4, \beta_6), (\beta_6, \beta_8)$. Проводимость цепи, ведущей из вершины β_1 в вершину β_4, вычислена в предыдущем примере. Проводимость цепи $(\beta_4, \beta_5), (\beta_5, \beta_7), (\beta_7, \beta_8)$ равна $f_5(u,v) = \xi^1_{g_\leqslant, y_1}(u,v) \& f^2_{\geq, y_1} \& f^2_{\geq, y_2} = \begin{cases} 1, & \text{если } u \leqslant y_1 \text{ и } v \geq y_2 \\ 0, & \text{иначе} \end{cases}$. Проводимость цепи $(\beta_4, \beta_6), (\beta_6, \beta_8)$ равна $f_6(u,v) = \xi^2_{g_\leqslant, y_1}(u,v) \& f^2_{\geq, y_2} = \begin{cases} 1, & \text{если } u > y_1 \text{ и } v \geq y_2 \\ 0, & \text{иначе} \end{cases}$. Функция фильтра вершины β_8 равна $f_7 = f_4 \& (f_5 \vee f_6) = \begin{cases} 1, & \text{если } v-u < 1/3,\ u \leqslant y_2 \text{ и } v \geq y_2 \\ 0, & \text{иначе} \end{cases}$. Таким образом в вершину β_8 проходят запросы, длина которых меньше чем 1/3 и которые содержат точку y_2.

Через $\mathcal{R}(U), \mathcal{P}(U), \mathcal{L}(U)$ (или просто $\mathcal{R}, \mathcal{P}, \mathcal{L}$) обозначим множества вершин, точек переключения и листьев ИГ U соответственно.

Пусть $\mathcal{N}$ — некоторая подсеть (то есть произвольное подмножество вершин и ребер) ИГ U. Через $\langle \mathcal{N} \rangle$ обозначим множество записей, соответствующих листьям этой подсети (если α — некоторый лист ИГ U, то под $\langle \alpha \rangle$ будем понимать запись, соответствующую листу α).

Легко видеть, что функция ответа ИГ U определяется соотношением

$$\mathcal{J}_U(x) = \langle \{\alpha \in \mathcal{L}(U) : \varphi_\alpha(x) = 1\} \rangle.$$

Из определения функционирования ИГ видно, что ИГ как управляющая система может рассматриваться в качестве модели алгоритма поиска, работающего над данными, организованными в структуру, определяемую структурой ИГ.

В случае, когда базовое множество переключателей G пусто, то есть в графах нет переключателей, то ИГ называются *предикатными информационными графами* (ПИГ).

ПИГ, различным листьям которого соответствуют различные записи, называется *однозначным информационным графом* (ОИГ).

ОИГ, имеющий вид дерева, листья которого совпадают с концевыми вершинами дерева, назовем *информационным деревом* (ИД).

ИД удобны и интересны тем, что структуры данных, им соответствующие, практичны и их гораздо проще реализовать на ЭВМ.

Пример 2.9. Приведем пример еще одного ИГ. Пусть $I = \langle X, V, = \rangle$ — задача поиска идентичных объектов, где на множестве $V = \{y_1, \ldots, y_7\}$ задан линейный порядок и записи упорядочены в порядке возрастания, то есть $y_1 \leqslant y_1 \leqslant \cdots \leqslant y_7$. Пусть множество предикатов имеет вид

$$F = \{f_{=,a}(x) = \begin{cases} 0, & \text{если } x \neq a \\ 1, & \text{если } x = a \end{cases} : a \in X\}, \tag{2.6}$$

а множество переключателей — вид

$$G = \{g_{\leqslant,a}(x) = \begin{cases} 1, & \text{если } x \leqslant a \\ 2, & \text{если } x > a \end{cases} : a \in X\}. \tag{2.7}$$

Бинарным поиском или *методом деления пополам* (см., например, [3, 6, 13]) называется алгоритм поиска в упорядоченном массиве, при котором массив делится пополам, запрос сравнивается со средней точкой и в зависимости от результата сравнения поиск рекурсивно повторяется в одной из половин.

На рисунке 2.2 приведен ИГ над базовым множеством $\mathcal{F} = \langle F, G \rangle$, решающий ЗИП I, соответствующий бинарному поиску в версии Боттенбрука [19, с. 214], в которой вопрос о равенстве записи и запроса откладывается до самого последнего момента.

Из определения функционирования ИГ естественным образом вытекает, что каждому ИГ U можно сопоставить следующую процедуру поиска.

Предполагается, что эта процедура хранит в своей (внешней) памяти структуру ИГ U. Входными данными процедуры является запрос. Выходными данными является множество записей.

Пусть на вход процедуры поступил запрос x. Вводим понятие активного множества вершин и вносим в него в начальный момент корень ИГ U и помечаем его. Далее по очереди просматриваем вершины из активного множества и для каждой из них проделываем следующее:

- если рассматриваемая вершина — лист, то запись, приписанную вершине, включаем в ответ;

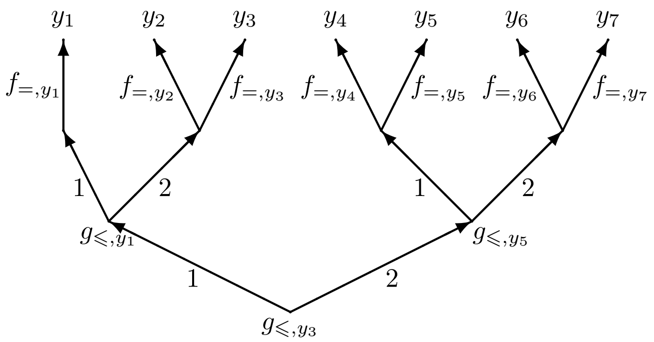


Рис. 2.2: Информационный граф бинарного поиска

- если рассматриваемая вершина переключательная, то вычисляем на запросе x переключатель, соответствующий данной вершине, и если конец ребра, исходящего из рассматриваемой вершины, нагрузка которого равна значению переключателя, непомеченная вершина, то помечаем его и включаем в множество активных вершин;

- если рассматриваемая вершина предикатная, то просматриваем по очереди исходящие из нее ребра и вычисляем значения предикатов, приписанных этим ребрам, на запросе x. Концы ребер, которым соответствуют предикаты со значениями, равными 1, если они непомеченные, помечаем и включаем в множество активных вершин;

- исключаем рассматриваемую вершину из активного множества.

Процедура завершается по исчерпании активного множества.

Заметим, что если ИГ решает задачу I, то множество, полученное на выходе процедуры, будет содержать все те и только те записи библиотеки $\langle U\rangle$, которые удовлетворяют запросу x. То есть полученная процедура решает ЗИП $I = \langle X, V, \rho\rangle$, где $V = \langle U\rangle$, и, значит, является алгоритмом поиска.

Таким образом, ИГ как управляющая система может рассматриваться как модель алгоритма поиска, работающего над данными, организованными в структуру, определяемую структурой ИГ.

В данном разделе мы введем понятия сложности ИГ, которые будут характеризовать такие общепринятые меры сложности алгоритмов поиска как объем памяти, время поиска в худшем случае и время поиска в среднем.

Отметим, что в большинстве работ, посвященных исследованию сложности алгоритмов поиска, под сложностью алгоритма понимается время поиска в худшем случае, и в сравнительно редких случаях исследуется среднее время поиска, хотя для задач поиска, используемых в базах данных, для которых характерны массовость и многократность, исследование среднего времени поиска представляется более актуальным. Некоторое объяснение крена в сторону изучения сложности в худшем случае можно найти в цитате из [16, стр.20]: "К сожалению, анализ поведения в среднем значительно более сложная вещь, чем анализ худшего случая, по двум причинам: во-первых существенные математические трудности возникают, даже если удачно выбрано исходное распределение; во-вторых, часто с трудом достигается согласие в том, что именно выбранное распределение является реальной моделью изучаемой ситуации. Вот почему преобладающее большинство результатов связано с анализом худших случаев."

Определим понятие сложности ИГ на запросе.

Будем считать, что время вычисления любого переключателя из G примерно одинаково и характеризуется числом a, а время вычисления любого предиката из F — числом b.

Пусть нам дан некий ИГ U и произвольно взятый запрос $x \in X$.

Сложностью ИГ U на запросе x назовем число

$$T(U,x) = a \cdot \sum_{\beta \in \mathcal{P}} \varphi_\beta(x) + b \cdot \sum_{\beta \in \mathcal{R} \setminus \mathcal{P}} \psi_\beta \cdot \varphi_\beta(x).$$

Величина $T(U,x)$ характеризует время работы описанной выше процедуры поиска, сопоставленной ИГ U, поскольку $T(U,x)$ равно количеству переключателей, вычисленных данной процедурой при подаче на ее вход запроса x, умноженное на a, плюс количество вычисленных предикатов, умноженное на b.

Сложность ИГ можно вводить двумя способами. Во-первых, как максимальную сложность на запросе

$$\widehat{T}(U) = \max_{x \in X} T(U,x)$$

(здесь мы берем max, а не sup, так как $T(U,x)$ принимает целые значения, и, значит, sup всегда достигается). Эта величина характеризует время поиска в худшем случае соответствующим ИГ алгоритмом и ее будем называть *В-сложностью ИГ* (верхней сложностью). Эта величина исследуется в большинстве работ, посвященных проблемам сложности задач поиска.

Во-вторых, можно вводить сложность ИГ как среднее значение сложности ИГ на запросе, взятое по множеству всех запросов. С этой целью введем *вероятностное пространство* над множеством запросов X, под которым будем понимать тройку $\langle X, \sigma, \mathbf{P} \rangle$, где σ — некоторая *алгебра подмножеств* множества X, $\mathbf{P}$ — *вероятностная мера* на σ, то есть аддитивная мера, такая, что $\mathbf{P}(X) = 1$.

В связи с тем, что мы ввели вероятностное пространство над множеством запросов, уточним понятие типа. А именно, под *типом* будем понимать тройку $S = \langle X, Y, \rho \rangle$, считая, что множество запросов X рассматривается вместе со своим вероятностным пространством $\langle X, \sigma, \mathbf{P} \rangle$. В тех же случаях, когда мы хотим явно выделить рассматриваемое вероятностное пространство над X, мы будем представлять тип пятеркой $S = \langle X, Y, \rho, \sigma, \mathbf{P} \rangle$.

Скажем, что базовое множество $\mathcal{F} = \langle F, G \rangle$ *измеримое*, если алгебра σ содержит все множества N_f, где $f \in F \cup \widehat{G}$.

Справедлива следующая лемма.

Лемма 2. *Если базовое множество $\mathcal{F} = \langle F, G \rangle$ измеримое, то для любого ИГ U над базовым множеством $\mathcal{F}$ функция $T(U,x)$, как функция от x, является случайной величиной.*

Далее всюду будем предполагать, что базовое множество измеримо.

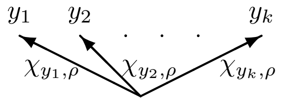


Рис. 2.3: Информационный граф переборного алгоритма

Сложностью ИГ U назовем математическое ожидание величины $T(U,x)$, то есть число
$$T(U) = \mathbf{M}_x\, T(U,x).$$

Если (β,α) — ребро ИГ, то *сложностью этого ребра* назовем число

- $b \cdot \mathbf{P}(N_{\varphi_\beta})$ — если (β,α) — предикатное ребро;
- $a \cdot \mathbf{P}(N_{\varphi_\beta})/\psi_\beta$ — если это ребро переключательное.

Если β — вершина ИГ, то число $\mathbf{P}(N_{\varphi_\beta})$ назовем *сложностью вершины* β.

Нетрудно показать, что сложность ИГ равна сумме сложностей ребер ИГ. В самом деле

$$\begin{aligned}
T(U) &= \mathbf{M}_x\, T(U,x) = \int_X T(U,x)\mathbf{P}(dx) = \\
&= \int_X (b \cdot \sum_{\beta\in\mathcal{R}\setminus\mathcal{P}} \psi_\beta \cdot \varphi_\beta(x) + a \cdot \sum_{\beta\in\mathcal{P}} \varphi_\beta(x))\mathbf{P}(dx) = \\
&= b \cdot \sum_{\beta\in\mathcal{R}\setminus\mathcal{P}} \psi_\beta \int_X \varphi_\beta(x)\mathbf{P}(dx) + a \cdot \sum_{\beta\in\mathcal{P}} \int_X \varphi_\beta(x)\mathbf{P}(dx) = \\
&= b \cdot \sum_{\beta\in\mathcal{R}\setminus\mathcal{P}} \psi_\beta \mathbf{P}(N_{\varphi_\beta}) + a \cdot \sum_{\beta\in\mathcal{P}} \mathbf{P}(N_{\varphi_\beta}).
\end{aligned}$$

Далее всюду будем предполагать, что $a = b = 1$.

Пусть нам дан ИГ U.

Объемом $Q(U)$ ИГ U назовем число ребер в ИГ U.

В качестве примера мы можем подсчитать сложность ИГ U, изображенного на рисунке 2.3. Легко видеть, что $Q(U) = k$ и $T(U) = k$, то есть объем графа минимально возможный, а время максимальное. Это и не удивительно, так как ИГ U соответствует переборному алгоритму поиска.

Пусть нам дана некая ЗИП I. *Сложностью задачи* I *при базовом множестве* $\mathcal{F}$ *и заданном объеме* q назовем число

$$T(I,\mathcal{F},q) = \inf\{T(U) : U \in \mathcal{U}(I,\mathcal{F}) \text{ и } Q(U) \leqslant q\},$$

где $\mathcal{U}(I,\mathcal{F})$ — множество всех ИГ над базовым множеством $\mathcal{F}$, решающих ЗИП I.

Соответственно *В-сложностью задачи* I *при базовом множестве* $\mathcal{F}$ *и заданном объеме* q назовем число

$$\widehat{T}(I,\mathcal{F},q) = \min\{\widehat{T}(U) : U \in \mathcal{U}(I,\mathcal{F}) \text{ и } Q(U) \leqslant q\}$$

(здесь мы берем min, а не inf, так как $\widehat{T}(U)$ принимает целые значения, и, значит, inf всегда достигается).

Число

$$T(I,\mathcal{F}) = \inf\{T(U) : U \in \mathcal{U}(I,\mathcal{F})\}$$

назовем *сложностью задачи I при базовом множестве $\mathcal{F}$*.

Соответственно *B-сложностью задачи I при базовом множестве $\mathcal{F}$* назовем число

$$\widehat{T}(I,\mathcal{F}) = \min\{\widehat{T}(U) : U \in \mathcal{U}(I,\mathcal{F})\}.$$

Если k — натуральное число, S — тип задач поиска, то обозначим

$$\mathcal{I}(k,S) = \{I = \langle X, V, \rho\rangle \in S : |V| = k\}.$$

Будем исследовать функции, характеризующие сложность класса ЗИП $\mathcal{I}(k,S)$, такие как функции Шеннона:

$$\widehat{\mathcal{T}}(k,S,\mathcal{F}) = \max_{I\in\mathcal{I}(k,S)} \widehat{T}(I,\mathcal{F}),$$

$$\mathcal{T}(k,S,\mathcal{F}) = \sup_{I\in\mathcal{I}(k,S)} T(I,\mathcal{F}),$$

(в первом случае мы берем max, а не sup, так как $\widehat{T}(I,\mathcal{F})$ принимает целые значения, и, значит, sup всегда достигается).

Если существует такой ИГ $U \in \mathcal{U}(I,\mathcal{F})$, что $T(U) = T(I,\mathcal{F})$, то ИГ U будем называть *оптимальным* для ЗИП I. Соответственно если $\widehat{T}(U) = \widehat{T}(I,\mathcal{F})$, то ИГ U будем называть *B-оптимальным* для ЗИП I.

Мощностная нижняя оценка

Пусть нам даны произвольные множества запросов X, записей Y и отношение поиска ρ на $X \times Y$. Причем на множестве запросов задано вероятностное пространство $\langle X, \sigma, \mathbf{P}\rangle$.

Скажем, что базовое множество $\mathcal{F}$ *допустимо для ЗИП I*, если существует ИГ над базовым множеством $\mathcal{F}$, который решает ЗИП I.

Следующий результат, называемый мощностной нижней оценкой, справедлив для любой ЗИП при минимальных ограничениях. Смысл этого результата заключается в том, что время поиска не может быть меньше чем время, необходимое на перечисление ответа.

Справедлива следующая теорема.

Теорема 4 (мощностная нижняя оценка). *Пусть $I = \langle X, V, \rho\rangle$ — произвольная ЗИП, такая, что существует такая запись $y \in V$, что $O(y,\rho) \neq \emptyset$, $\mathcal{F}$ — измеримое базовое множество, допустимое для I, тогда*

$$T(I,\mathcal{F}) \geq \max(1, \sum_{y\in V} \mathbf{P}(O(y,\rho))),$$

$$\widehat{T}(I,\mathcal{F}) \geq \max_{x\in X} |\mathcal{J}_I(x)|.$$

Упражнения.

Задача 2.2.1. Задача поиска идентичных объектов. Пусть $S_{\text{id}} = \langle X, X, =\rangle$, где $X = \{1, 2, \dots, 20\}$, заданы множество переключателей: $G = \{g_{<,a} \in X\}$, где $g_{<,a} = \begin{cases} 1, & \text{при } x < a; \\ 2, & \text{при } x \geq a \end{cases}$ и предикатов $G = \{f_{=,a} | a \in X\}$, где $f_{=,a}(x) \Leftrightarrow (a = x)$. Задана библиотека $V = \{2, 3, 5, 7, 11, 13\} \subset X$.

a) Приведите пример информационного графа над базовым множеством $\mathcal{F} = (F, G)$, разрешающего ЗИП $I = \langle X, V, =\rangle$.
b) Чему равна его В–сложность? Можно ли построить ИГ, обладающий меньшей В–сложностью?
c) На множестве запросов X задана равномерная вероятностная мера. Найти среднюю сложность полученного графа.

Задача 2.2.2. Задача о ближайшем соседе справа. Пусть $S_{\text{neigh}} = \langle X, Y, \rho\rangle$, где $X = Y = \{1, \dots, 20\}$, а отношение ρ задается условием: $x\rho y \Leftrightarrow ((y \in V)\&(x \leqslant y)\&(\nexists\, y' \in V : x \leqslant y' < y))$.Задано множество переключателей $G = \{g_a : a \in Y\}$, где $g_a(x) = \begin{cases} 1, & x < a \\ 2, & x = a \\ 3, & x > a \end{cases}$ Базовое множество $\mathcal{F} = \langle \emptyset, G\rangle$(предикатов нет).

a) Постройте ИГ, решающий ЗИП для $V = \{2, 3, 5, 10, 11, 17, 18, 20\}$
b) Найти сложность и среднюю сложность полученного графа (если на множестве запросов X задана равномерная вероятностная мера).
c) Подумайте, какие функции можно было бы добавить в базовое множество для снижения средней сложности?

Задача 2.2.3. Задача одномерного интервального поиска. Пусть заданы множество запросов $X_N = \{\overline{1, N}\}^2$, множество ответов $Y_N = \{\overline{1, N}\}$ (N — фиксированное число) и отношение поиска $(x_1, x_2)\rho_{\in[]} y \iff y \in [x_1, x_2]$. Пусть $S_{\text{int}} = \langle X_N, Y_N, \rho_{\in[]}\rangle$ — задача одномерного интервального поиска. Базовое множество состоит из множества переключателей $G = \{g_a^{(i)},\, a \in Y_N, i = 1, 2\}$, где $g_a^{(i)}(\mathbf{x}) = \begin{cases} 1, & x_i < a; \\ 2, & x_i = a; \\ 3, & x_i > a; \end{cases}$ и множества предикатов $F = \{f_a^{(i)},\, a \in Y_N, i = 1, 2\}$, где $f_a^{(1)}(\mathbf{x}) \Leftrightarrow (x_1 \leqslant a)$ и $f_a^{(2)}(\mathbf{x}) \Leftrightarrow (a \leqslant x_2)$. Постройте информационный граф над этим множеством, решающий ЗИП $I = \langle X_N, V, \rho_{\in[]}\rangle$ для $N = 100$ и библиотеки а $V = \{19, 20, 40, 60\}$.

Задача 2.2.4. Задача двоично–рационального поиска идентичных объектов. Будем обозначать $\mathbb{Q}_n^{(2)} = \{\frac{k}{2^n} : k = 0, 1, \dots, 2^n - 1\}$ — множество двоично–рациональных чисел заданной точности (n фиксировано). Пусть $S_{\text{rat}} = \langle X_n, Y_n, =\rangle$, где $X_n = Y_n = \mathbb{Q}_n^{(2)}$. Базовое множество состоит из предикатов $f_\alpha^{(s)}(x) \Leftrightarrow (\alpha_s = \alpha)$ и переключателей $g_s(x) = \alpha_s(x) + 1,\ s = 1, 2, \dots, n$, где $(0{,}\alpha_1 \dots \alpha_n)_2$ — двоичное разложение числа x, т.е. $x = \sum \alpha_i \cdot 2^{-i}$. Построить ИГ, решающий ЗИП для $n = 4$ и библиотеки $V_4 = \{0.0011; 0.0111; 0.1100; 0.1110; 0.1111\}$ (указана запись в двоичной системе счисления). б) Найти его среднюю сложность и В–сложность, если на $\mathbb{Q}_4^{(2)}$ задана равномерная вероятностная мера $\mathcal{P}(x) = \frac{1}{16}$.

Задача 2.2.5. Задача поиска идентичных объектов. Пусть $X = Y = [0,1]$, $S_{\text{id}} = \langle X, Y, = \rangle$, базовое множество состоит из переключателя $g(x) = [5x] + 1$, и предикатов $f_a(x) \Leftrightarrow (x = a)$. Задана библиотека $V = \{0.1, 0.33, 0.6, 0.75, 0.9, 1\}$, построить информационный граф, решающий ЗИП $I_{\text{id}} = \langle X, V, = \rangle$, b) Найти его среднюю и В–сложность (в случае равномерного распределения запросов).

Задача 2.2.6. Задача одномерного интервального поиска Пусть $S_{\text{int}} = \langle [0,1]^2, [0,1], \rho_{\in[]} \rangle$, где $(x_1, x_2)\rho_{\in[]} y \Longleftrightarrow y \in [x_1, x_2]$. Базовое множество состоит из множества переключателей $G = \{g_a^{(i)},\ a \in [0,1], i = 1, 2\}$, где $g_a^{(i)}(\mathbf{x}) = \begin{cases} 1, & x_i < a; \\ 2, & x_i = a; \\ 3, & x_i > a; \end{cases}$ и множества предикатов $F = \{f_a^{(i)},\ a \in [0,1], i = 1, 2\}$,
где $f_a^{(1)}(\mathbf{x}) \Leftrightarrow (x_1 \leqslant a)$, $f_a^{(2)}(\mathbf{x}) \Leftrightarrow (a \leqslant x_2)$.

a) Постройте информационный граф над этим множеством, решающий ЗИП $I = \langle X_{int}, V, \rho_{int} \rangle$ с библиотекой а $V = \{0.02;\ 0.11;\ 0.22;\ 0.33;\ 0.75;\ 0.99\}$.
b) Пусть задана библиотека $V = \{y_1, y_2, \ldots, y_k\} \subseteq [0,1]$. Найдите мощностную нижнюю оценку для ЗИП $I = \langle X_{int}, V, \rho_{int} \rangle$.

Задача 2.2.7. Задача одномерного интервального поиска. Пусть $X = \{(x_1, x_2) \in \{\overline{1, 100}\}^2 \ :\ x_1 \leqslant x_2\}$ — множество запросов, $Y = \{\overline{1, 100}\}$ — множество записей, задача интервального поиска $S_{\text{int1}} = \langle [0,1]^2, [0,1], \rho_{\in[]} \rangle$, где $(x_1, x_2)\rho_{\in[]} y \Longleftrightarrow y \in [x_1, x_2]$. Базовое множество состоит из переключателей

$$g_a(x_1, x_2) = \begin{cases} 1, & x_2 < a \\ 2, & x_2 = a \\ 3, & x_1 < a < x_2 \\ 4, & a = x_1 \\ 5, & a < x_1. \end{cases}$$

Постройте информационный граф над этим множеством, решающий ЗИП $I_{\text{int1}} = \langle X, V, \rho_{\in[]} \rangle$. для библиотеки $V = \{0.12;\ 0.13;\ 0.5;\ 0.6;\ 0.75;\ 0.99\}$.

Задача 2.2.8. Задача поиска в соответствии с регулярным выражением. Пусть множество записей $Y_n = \{0,1\}^n$ — слова из нулей и единиц длины n. Регулярным выражением[3] будем называть слово в алфавите $\{0, 1, *\}$, причем символу «$*$» может соответствовать как 0, так и 1. Более формально, $X_n = \{0, 1, *\}^n$, и отношение поиска $\mathbf{x}\rho_r\mathbf{y} \Leftrightarrow \big(\forall i \in \overline{1, n}\big((x_i = y_i) \vee (x_i = *)\big)\big)$. Пусть $S_{\text{bin-int}} = \langle X_n, Y_n, \rho_r \rangle$, где

[3]В этой задаче дается предельно упрощенное понятие регулярного выражения

Базовое множество состоит из предикатов $f_a^{(i)}(\mathbf{x}) \Leftrightarrow (x_i = a \vee x_i = *), i =$ $= 1, 2; a \in \{0, 1\}$ и переключателей $g^{(i)}(\mathbf{x}) = \begin{cases} 1, & x_i = 0 \\ 2, & x_i = * \\ 3, & x_i = 1 \end{cases}.$

a) Построить ИГ, решающий ЗИП для $n = 4$ и библиотеки $V_4 = \{0010; 0011; 1110; 1111\}$.

b) Найти его среднюю сложность и В–сложность, если на X_4 задана равномерная вероятностная мера.

c) Найти нижнюю мощностную оценку ЗИП.

Задача 2.2.9. Одномерная задача о доминировании. Пусть $X = Y = [0, 1]$, отношение $\rho(x, y) \Leftrightarrow (x \geqslant y)$. Задача информационного поиска $S_{\mathrm{dom1}} = \langle [0, 1], [0, 1], \geqslant, P, L\rangle$, где $\mathcal{L}$ — множество измеримых по–Лебегу подмножеств отрезка [0,1], P— мера Лебега. Базовое множество состоит из переключателей $G = \{g_a(x),\ a \in [0, 1]\}$, где $g_a(x) = \begin{cases} 1, & x \leqslant a; \\ 2, & x = a; \end{cases}.$

a) Постройте ИГ над ним, решающий ЗИП $I = \langle [0, 1], V, \geqslant\rangle$ с библиотекой $V = \{0.1, 0.5, 0.75, 0.9, 0.99\}$.

b) Найдите его В–сложность и среднюю сложность (запросы распределены равномерно на $X = [0, 1]$).

c) Найдите мощностную нижнюю оценку сложности.

Задача 2.2.10. Пусть $X = Y = \{\overline{1, 100}\}$, Задано базовое множество $\mathcal{F} =$ $= \langle F, G\rangle$, состоящее из переключателей $G = \{g_a : a \in Y\}$, где $g_a(x) =$ $= \begin{cases} 1, & x < a \\ 2, & x = a \\ 3, & x > a \end{cases}$ и предикатов $F = \{f_{=,a} : a \in Y\}$, где $f_{=,a}(x) \Leftrightarrow (x = a)$.
Запросы равномерно распределены на множестве X.

a) Построить ИГ, решающий ЗИП $S_{\mathrm{id}} = \langle X, V, =\rangle$, с библиотекой $V =$ $= \{1, 2, 3, 25, 45, 50, 90, 95, 100\}$.

b) Найти его сложность и В–сложность.

c) Пусть в базовое множество добавлен переключатель $h^{(0.05)}(x) =$ $= [0.05x]+1$. Постройте ИГ, решающий ту же задачу с использованием этого переключателя. Какова его средняя сложность?

Задача 2.2.11. Задача о ближайшем соседе справа. Пусть $S_{\mathrm{neigh}} = \langle X, Y, \rho\rangle$, где $X = Y = \{1, \dots, 25\}$, а отношение ρ задается условием: $x\rho y \Leftrightarrow ((y \in V)\&(x \leqslant y)\&(\nexists\, y' \in V : x \leqslant y' < y))$. Базовое множество состоит из переключателей $g_a(x)$, принимающих значения 1,2 и 3 при $x < a$, $x = a$ и $x > a$ соответственно.

a) Постройте ИГ, решающий ЗИП для $V = \{1, 2, 4, 8, 16\}$;

b) Найдите его среднюю сложность, если запросы распределены равномерно на X.

Задача 2.2.12. Задача поиска ближайшего соседа. Пусть $X = Y =$ $= \{1, 2, \dots, 100\}$, отношение $x\rho y \Leftrightarrow \forall y' \in V(|x - y| \leqslant |x - y'|)$. Базовое множество состоит из переключателя $g(x) = \left[\frac{x}{23}\right] + 1$ и предикатов вида $f_a^{(s)}(x) \Leftrightarrow |a - x| \leqslant s$, $a, s \in \mathbb{R}$. a) Построить ИГ, решающий ЗИП с библиотекой $V = \{1, 17, 33, 58, 80, 90\}$. Обратите внимание, что ближайший сосед может быть не один. b) Найти его среднюю и B-сложность (предполагается, что запросы распределены равномерно на X).

Задача 2.2.13. Задача поиска идентичных объектов. Пусть $X = \{1, 2, \ldots, 100\}$, $S_{\mathrm{id}} = \langle X, X, = \rangle$, базовое множество состоит из переключателей $g_{20}(x) = [\frac{x}{20}] + 1$ и предикатов $f_a(x) \Leftrightarrow (x = a)$. a) Построить информационный граф, решающий ЗИП $I = \langle X, V, = \rangle$, $V = \{15, 17, 21, 23, 27, 29, 43\}$. b) Найти его среднюю и B–сложность.

Задача 2.2.14. Задача о ближайшем соседе справа. Пусть $S_{\mathrm{neigh}} = \langle X, Y, \rho \rangle$, где $X = Y = [0, 1]$, а отношение ρ задается условием: $x\rho y \Leftrightarrow ((y \in V) \& (x \leqslant y) \& (\nexists\, y' \in V : x \leqslant y' < y))$. Базовое множество состоит из переключателей $g_a(x)$, принимающих значения 1,2 и 3 при $x < a$, $x = a$ и $x > a$ соответственно. Постройте ИГ, решающий ЗИП для $V = \{0.1, 0.25, 0.33, 0.5, 0.7\}$;

Задача 2.2.15. Задача о ближайшем соседе справа. Пусть $X = Y = \{1, \ldots, 30\}$. Отношение ρ задается условием: $x\rho y \Leftrightarrow ((y \in V) \& (x \leqslant y) \& (\nexists\, y' \in V : x \leqslant y' < y))$. Пусть $g_a(x)$ равно 1, 2, 3 при $x < a$, $x = a$ и $x > a$, соответственно. Пусть $G = \{g_a : a \in Y\}$, $\mathcal{F} = \langle \emptyset, G \rangle$. На множестве запросов X задано вероятностное пространство $\mathcal{X} = \langle X, \sigma, P \rangle$, где $\sigma = 2^X$, $P(M) = |M|/|X|$ — равномерная вероятностная мера. а) Постройте ИГ, решающий ЗИП для $V = \mathbb{P} \cap [1, 30]$ (где $\mathbb{P}$ — множество простых чисел). б) Найдите его сложность и B–сложность.

Задача 2.2.16. Задача поиска идентичных объектов. Пусть $S_{\mathrm{id}} = \langle X, X, = \rangle$, где $X = \{0, 1, 2, \ldots, 1023\}$, заданы множество переключателей: $G = \{g_{<,a} | a \in X\}$, где $g_{<,a}(x) = \begin{cases} 1, & \text{при } x < a; \\ 2, & \text{при } x \geq a \end{cases}$ и предикатов $G = \{f_a | a \in X\}$, где $f_a(x) \Leftrightarrow (a = x)$, задана библиотека $V = X$.

a) Опишите схему построения какого–либо ИГ, решающего эту ЗИП. Какова его B-сложность?
b) Докажите, что невозможно построить ИГ, решающий эту ЗИП, B-сложность которого меньше 10;
c) Пусть в базовое множество добавлены переключатели: $h_{16}(x) = 1 + (x \mod 16)$, (где $x \mod 16$ — остаток от деления x на 16). Описать схему построения ИГ, у которого B-сложность меньше 10.
d) Пусть в базовое множество добавлены переключатели: $h_a(x) = 1 + (x \mod a)$, $a = 2, 3, ..., 30$ (т.е. остаток от деления на a). Описать ИГ, у которого B–сложность не превосходит 3.
e) Доказать, что для базового множества из пункта d) невозможно построить ИГ, B–сложность которого равна 2.
f) Допустим $X = Y = V = \{1, ..., N\}$, где $N > 1024$. Приведите нижнюю оценку на B-сложность ИГ.
g) Предложите какую-либо схему построения ИГ при больших N. Как будет меняться B-сложность при $N \to \infty$?

Задача 2.2.17. Задача двоично-рационального интервального поиска. Будем обозначать $\mathbb{Q}_n^{(2)} = \{\frac{k}{2^n} : k = 0, 1, \ldots, 2^n - 1\}$ — множество двоично-рациональных чисел заданной точности (как в задаче 2.2.4). Пусть $X = \{\mathbf{x} = (x_1, x_2) \in \mathbb{Q}_n^{(2)} : x_1 \leqslant x_2\}$. Отношение $\mathbf{x}\rho y \Leftrightarrow x_1 \leqslant y \leqslant x_2$ (знак «$\leqslant$» понимается как сравнение действительных чисел). Базовое множество состоит из предикатов $f_{i,\alpha}^{(s)}(x) \Leftrightarrow (\alpha_s^{(i)} = \alpha)$ где $x_i = (0{,}\alpha_1^{(i)} \ldots \alpha_n^{(i)})_2$, $i = 1, 2$; и переключателей $g_q^{(i)}(x) = \begin{cases} 1, & x_i < q; \\ 2, & x_i = q;, \\ 3, & x_i > q. \end{cases}$ где $q \in \mathbb{Q}_n^{(2)}$, $i = 1, 2$. а) Построить ИГ, решающий ЗИП для $n = 3$ и $V = \{0.001; 0.010; 0.111\}$. б) Найти его сложность и В-сложность, если на X задана равномерная вероятностная мера $\mathcal{P}(x = a) = \frac{1}{C_{2^n}^2 + 2^n}$. в) Найти нижнюю мощностную оценку ЗИП.

Задача 2.2.18. Задача о поиске идентичных объектов.

Пусть $X = Y = \{1, \ldots, 60\}$, базовое множество состоит из предикатов $f_a(x) \Leftrightarrow (a = x)$ и переключателей $g_a(x) = \begin{cases} 1, & x < a \\ 2, & x = a \\ 3, & x > a \end{cases}$. На множестве запросов задана равномерная вероятностная мера: $P(x = a) = \frac{1}{60}$.

a) Построить ИГ, решающий ЗИП $\langle X, V, = \rangle$, где $V = \{1, 2, 3, 25, 45, 48, 49, 50\}$.
b) Найти его сложность и В–сложность.
c) Пусть в базовое множество добавлен переключатель $h_{12}(x) = \lceil \frac{x}{12} \rceil$ (через $\lceil a \rceil$ обозначается a округленное вверх до ближайшего целого). Постройте ИГ меньшей сложности, решающий ту же задачу.

Задача 2.2.19. Задача одномерного интервального поиска. Пусть $S_{int} = \langle X_{int}, Y_{int}, \rho_{int} \rangle$ — тип задачи одномерного интервального поиска, где $Y_{int} = [0; 1]$, $X_{int} = \{(u, v) : 0 \leqslant u \leqslant v \leqslant 1\}$ и $(x_1, x_2)\rho_{int}y \Longleftrightarrow x_1 \leqslant y \leqslant x_2$. Пусть задана библиотека $V = \{y_1, y_2, \ldots, y_k\} \subseteq [0, 1]$ и на множестве запросов $X_{int} = \{(u, v) : 0 \leqslant u \leqslant v \leqslant 1\}$ задана равномерная вероятностная мера. a) Приведите мощностную нижнюю оценку для ЗИП $I = \langle X_{int}, V, \rho_{int} \rangle$. b) При каких $\{y_1, y_2, \ldots, y_k\}$ оценка, полученная в пункте a) будет максимальна?

Задача 2.2.20. Задача поиска идентичных объектов Пусть $X = Y = \{1, 2, \ldots, 100\}$, $\rho(x, y) \leftrightarrow (x = y)$, на множестве X задана равномерная вероятностная мера $P(i) = \frac{1}{100}$. Базовое множество множество состоит из предикатов $f_a(x) \Leftrightarrow (a = x)$ и переключателя $g_{10}(i) = (i \mod 10) + 1$ (остаток от деления на 10). Пусть задана библиотека $V = \{y_1, y_2, \ldots, y_k\} \subset Y$. Найдите наименьшую возможную среднюю сложность ИГ, решающего эту задачу.

Задача 2.2.21. Пусть $S_{dom1} = \langle [0, 1], [0, 1], \geqslant, P, L \rangle$ — одномерная задача о доминировании. Пусть $V = \{y_1, y_2, \ldots, y_k\} \subset [0, 1]$. L — множество измеримых по-Лебегу подмножеств отрезка [0,1], P — мера Лебега. Приведите мощностную нижнюю оценку для ЗИП S_{dom1}.

Задача 2.2.22. Пусть $X = Y = V = \{1, ..., 1000\}$, ЗИП $S_{id} = \langle X, Y, = \rangle$ — задача поиска идентичных объектов. Базовое множество $\langle \emptyset, G \rangle$ состоит из переключателей $G = \{g_i, i = 2, 3, ..., 9\}$, где $g_i(x) = (x \mod i) + 1$. a) Найдите наименьшую возможную В-сложность ИГ, решающего эту ЗИП; b) Тот же вопрос для наименьшей средней сложности (если на X задана равномерная вероятностная мера).

Задача 2.2.23. Дано 9 конфет «Фундук в шоколаде», причем одна из них, бракованная, не содержит орешка внутри (т.е. легче остальных). Надо найти эту конфету за два взвешивания на чашечных весах без гирь.

a) Сформулировать задачу в виде ЗИП и задать базовое множество.
b) Построить ИГ, соответствующий алгоритму взвешивания.
c) Построить аналогичный ИГ для 16 конфет. Какова его средняя сложность (бракованная конфета одна, выбирается случайно)?

Задача 2.2.24. На столе в ряд лежат четыре монеты. Среди них обязательно есть как настоящие, так и фальшивые (которые легче настоящих). Известно, что любая настоящая монета лежит левее любой фальшивой. Надо за одно взвешивание на чашечных весах без гирь определить тип каждой монеты, лежащей на столе.

a) Сформулировать задачу в виде ЗИП и указать базовое множество.
b) Описать алгоритм решения и построить сответствующий ИГ.

Задача 2.2.25. Имеются 5 запертых замков и 5 ключей к ним. Надо за наименьшее число проверок определить, какому замку соответствует какой ключ.

a) Сформулировать задачу в виде ЗИП и задать базовое множество.
b) Описать схему ИГ наименьшей В-сложности.

Задача 2.2.26. *(Козлова Е.Г., Сказки и подсказки)* Известно, что медные монеты[4] достоинством в 1, 2, 3, 5 коп. весят соответственно 1, 2, 3, 5 г. Среди четырех медных монет (по одной каждого достоинства) есть одна бракованная, отличающаяся весом от нормальной (причем неизвестно в какую сторону). Надо с помощью взвешиваний на чашечных весах без гирь определить бракованную монету.

a) Сформулировать в виде ЗИП и указать соответствующее базовое множество.
b) Построить ИГ, решающий эту ЗИП.
c) Какова В-сложность полученного графа? Почему не существует ИГ меньшей В-сложности?

Задача 2.2.27. Лиса Алиса и Кот Базилио — фальшивомонетчики. Базилио делает монеты тяжелее настоящих, а Алиса — легче. У Буратино есть 15 одинаковых по внешнему виду монет, но какая-то одна — фальшивая. Надо за два взвешивания определить, кто сделал фальшивую монету — Кот Базилио или Лиса Алиса?

a) Сформулировать соответствующую задачу информационного поиска и задать базовое множество.

[4] образца 1961 года

b) Решить задачу и построить соответствующий информационный граф. Какова его B-сложность?

Задача 2.2.28. Из 11 шаров 2 радиоактивны. Про любой набор шаров за одну проверку можно узнать, имеется ли в нем хотя бы один радиоактивный шар (но нельзя узнать, сколько их). Требуется найти оба радиоактивных шара.

a) Сформулировать соответствующую ЗИП и указать базовое множество.
b) Описать схему построения ИГ, обладающего B–сложностью $T_B = 7$.
c) Доказать, что невозможно построить ИГ, обладающий B–сложностью $T_B \leqslant 5$.
d) * Доказать, что невозможно построить ИГ, обладающий B–сложностью $T_B = 6$.

Задача 2.2.29. * Дано 4 монеты, одна из них — фальшивая, но неизвестно, тяжелее или легче настоящей. Надо найти фальшивую монету за наименьшее количество взвешиваний.

a) Сформулировать задачу в виде ЗИП и указать базовое множество.
b) Описать алгоритм решения и построить ИГ B-сложности, равной 2.
c) Допустим, надо не только найти фальшивую монету, но и определить, тяжелей или легче она, чем настоящая. Как изменится формулировка ЗИП?
d) Доказать, что не существует информационного графа с B-сложностью, равной 2, решающего задачу из предыдущего пункта.

Задача 2.2.30. * Дано 12 монет, одна из них — фальшивая, но неизвестно, тяжелее или легче настоящей. Надо найти фальшивую монету не более чем за 3 взвешивания.

a) Сформулировать задачу в виде ЗИП и указать базовое множество.
b) Описать алгоритм решения и построить сответствующий ИГ.
c) Найти его B-сложность и среднюю сложность при условии, что фальшивая монета может быть легче или тяжелей настоящей с $p = \frac{1}{2}$.

Задача 2.2.31. Из пяти монет — две фальшивые. Одна из фальшивых монет легче настоящей, а другая — на столько же тяжелее настоящей. Надо найти обе фальшивые монеты. a) Сформулировать ЗИП; b) Построить ИГ B-сложности 3, решающий ее. c) Доказать, что невозможно решение с меньшим числом взвешиваний.

Задача 2.2.32. Дан массив длины 5, содержащий различные числа. Надо упорядочить его по возрастанию, если разрешается сравнивать любые два элемента. a) Сформулировать соответствующую задачу ЗИП b) Показать, что существует ИГ B–сложности 7, решающий ее (*строить сам граф не надо, достаточно описать схему построения*). c) Доказать, что не существует графа B-сложности 6, решающего эту ЗИП. d)* Доказать, что для задачи упорядочивания массива длины N минимальная B-сложность ИГ имеет порядок $O(N \log N)$ (при $N \to \infty$).

Задача 2.2.33. Для ускоренной проверки образцов крови на наличие какого-либо вируса медики применяют следующий метод: Пусть есть $N = kn$ образцов крови, каждый из которых содержит некоторый вирус с вероятностью p (которая обычно очень мала). Разобьем образцы на k групп по n образцов в каждой и сольем в одну пробирку по части каждого образца из группы. Если в смеси содержится вирус, то надо проверить каждый образец в группе (а если нет, то в группе нет инфицированных). а) Описать соответствующую информационно–графовую модель (сам граф информационного поиска можно не строить). б) Вычислить средную сложность при заданной вероятности p. Как она зависит от n при фиксированном N? с) Пусть $N = 1600$, $p = 10^{-4}$. При каких n средняя сложность минимальна?

Ответы и указания.

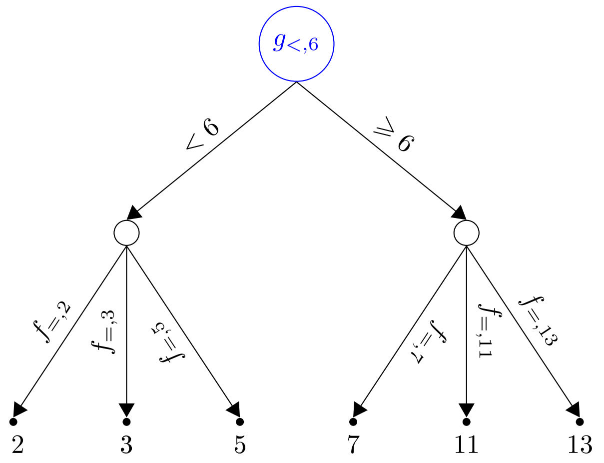


Рис. 2.4: К задаче 2.2.1

2.2.1 a) Граф указан на рис. 2.4. b), c) Сложность и B-сложность равны 4.

2.2.2 a) См. рис. 2.5. b) $T_b = 3$, $\hat{T} = \frac{1}{20} \cdot 1 + \frac{15}{20} \cdot 2 + \frac{4}{20} \cdot 3 = \frac{43}{20} = 2.15$.

2.2.3 См. рис. 2.6.

2.2.4 См. рис. 2.7.

2.2.5 а) См. рис. б) $\widehat{T} = 2$, $T_B = 3$.

2.2.6 а) См. рис. б) $\hat{T} \geqslant \sum\limits_{i=1}^{k} 2y_i(1-y_i)$.

2.2.7 Решение аналогично задаче 2.2.6.

2.2.8 a) См. рис. b) $\bar{T} = 1 + \frac{2}{3} + \left(\frac{2}{3}\right)^2 \cdot 2 + \left(\frac{2}{3}\right)^2 \cdot 2 = \frac{31}{9}$. c) $T \geqslant 4 \cdot \left(\frac{2}{3}\right)^4 = \frac{64}{81}$.

2.2.9 a) См. рис. b) $T_B = 7$ (т.к. учитывается время выдачи ответа). $\hat{T} = $ $= 0,1 \cdot 7 + (0,5 - 0,1) \cdot 6 + (0,75 - 0,5) \cdot 4 + (0,9 - 0,75) \cdot 3 + (0,99 - 0,9) \cdot 3 +$ $+ (1 - 0,99) \cdot 3 = 4,85$.

2.2.10 a), b) Решается аналогично задаче 2.2.1. c) См. рис. 2.12. Средняя сложность этого ИГ равна $\hat{T} = \frac{2}{100} \cdot 3 + \frac{1}{100} \cdot 2 + \frac{16}{100} \cdot 2 + \frac{20}{100} \cdot 2 + \frac{20}{100} \cdot 2 +$ $+ \frac{20}{100} \cdot 1 + \frac{11}{100} \cdot 2 + \frac{9}{100} \cdot 3 + \frac{1}{100} \cdot 1 = 1.9$.

2.2.11 Решается аналогично задаче 2.2.2.

2.2.12 а) См. рис. б) $\widehat{T} = 1 + \frac{22}{100} \cdot 2 + \frac{23}{100} \cdot 3 = 2.13$, $T_B = 4$.

2.2.13 решается аналогично задаче 2.2.10.c).

2.2.14 Решается аналогично задаче 2.2.2.

2.2.15 Решается аналогично задаче 2.2.2.

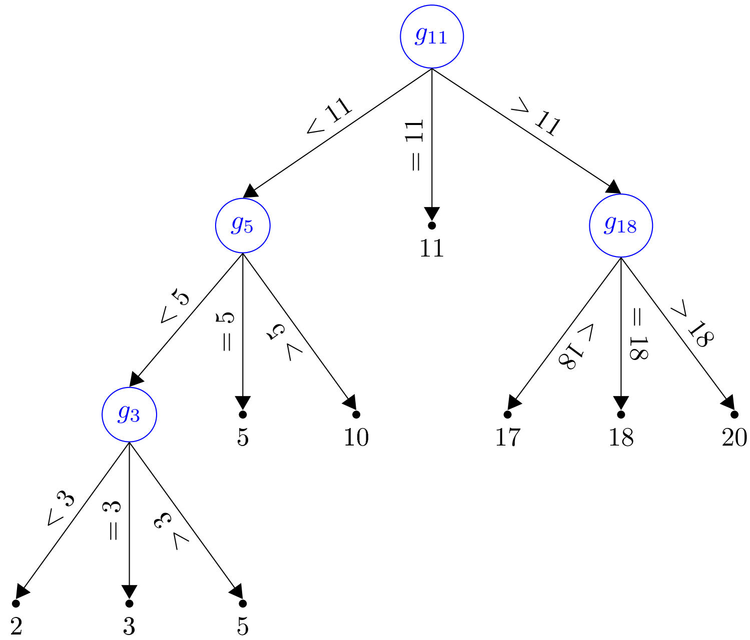


Рис. 2.5: К задаче 2.2.2

2.2.16 Не будем приводить сам граф, т.к. он слишком громоздок, опишем лишь схему его построения.

a) Схема представляет собой ярусное двоичное дерево. На первом ярусе происходит сравнение с $2^9 = 512$, что позволяет определить цифру, стоящую в 9-й позиции двоичного разложения искомого числа. На 2 ярусе происходит сравнение с 2^8 или $2^9 + 2^8$, и т. д. На каждом ярусе определяется один разряд,так что на последнем, 10-м ярусе будет определен младший разряд, т.е. найдено соответствующее число.

b) Дерево, содержащее 9 ярусов может иметь не более 512 листьев.

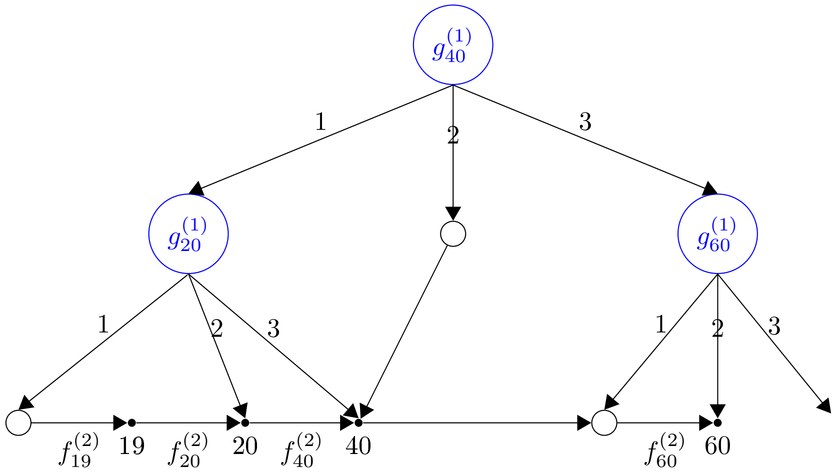
c) Последние 4 яруса могут быть заменены на переключатели, что дает B-сложность, равную 7.

d) Заметим, что достаточно взять остатки от деления на числа 25, 26 и 27 — они попарно взаимно просты. По китайской теореме об остатках любое число от 0 до $25 \times 26 \times 27 - 1 = 17\,549$ может быть восстановлено однозначно по остаткам. Т.е. достаточно рассмотреть ярусный граф, у которого в первом ярусе стоят переключатели h_{25}, во втором — h_{26} и в третьем — h_{27}.

e) Максимальное ветвление равно 30, значит у графа глубины 2 будет не более $30^2 = 900$ листьев.

f) $\log_{30} N$.

g) Выберем взаимно простые числа, не превосходящие 30, так, чтобы их произведение было максимальным: 29, 27, 25, 23, 19, 17, 13,11, 7. Пусть
$$P = 29 \cdot 27 \cdot 25 \cdot 23 \cdot 19 \cdot 17 \cdot 13 \cdot 11 \cdot 7.$$



К задаче 2.2.3

По китайской теореме об остатках за 9 проверок можно определить остаток от деления числа N на P. Для того, чтобы определить частное (методом деления пополам) потребуется еще $\lceil \log_2(N/P) \rceil$ проверок. Итого получаем (воспользовавшись калькулятором, можно найти $\log_2(P) \approx 37,08...$) $T_B \leqslant \log_2 N - 28$.

2.2.17 Задача решается аналогично задаче 2.2.6.

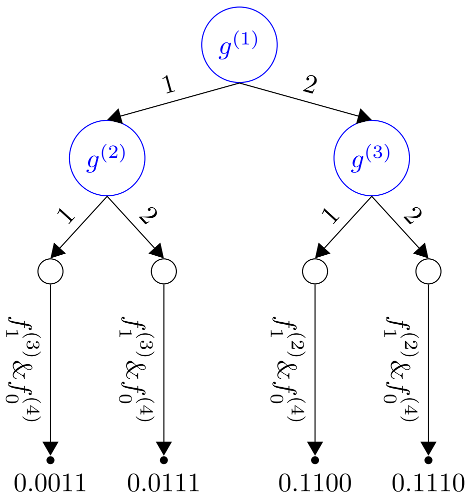
2.2.18 a),b) Решение аналогично задаче 2.2.1. c) См. рис. 2.14. Его средняя сложность равна $2\frac{11}{30}$.

2.2.19 a) Множество запросов, в ответ на которые должно быть включено y_i представляет собой прямоугольник $1 \leqslant x_1 \leqslant y_i \leqslant x_2 \leqslant 1$ (см.рис. 2.15), его площадь равна $y_i \cdot (1 - y_i)$, а вероятность такого запроса — $2y_i \cdot (1 - y_i)$. Следовательно, мощностная нижняя оценка сложности равна $2 \sum\limits_{i=1}^{k} y_i \cdot (1 - - y_i)$. b) Максимума не существует. Точная верхняя грань получается, если устремить все $y_i \to \frac{1}{2}$, она равна $\frac{k}{2}$.

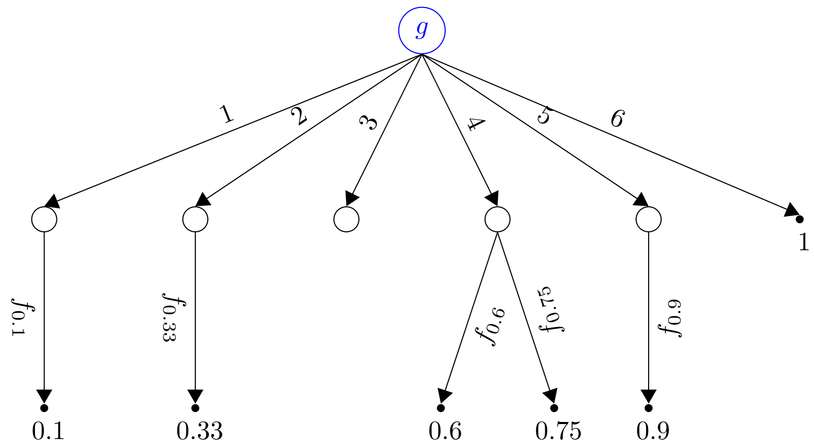
2.2.20 $k/10+1$. Заметим, что вероятность каждого остатка ровно 0.1, следовательно, любой элемент в библиотеке прибавляет столько к средней сложности. Кроме того надо учесть сам переключатель.

2.2.21 Решается аналогично задаче 2.2.19.

2.2.22 a) Граф состоящий из 3 ярусов имеет не более $7 \cdot 8 \cdot 9 = 504$ листьев, следовательно, он не может решать указанную ЗИП. Построим граф из 4 ярусов, состоящий из переключателей g_5, g_7, g_8, g_9. Числа 5,7,8,9 попарно взаимно просты, значит по китайской теореме об остатках любое число от 1 до $5 \cdot 7 \cdot 8 \cdot 9$ однозначно восстанавливается по остаткам от деления на эти числа. Ответ: 4. b) Воспользуемся пунктом a) и заметим, что числа от 501 до 504 могут быть однозначно восстановлены по остаткам от деления на 7,8,9. Следовательно, можно построить граф, который на этих запросах



К задаче 2.2.4

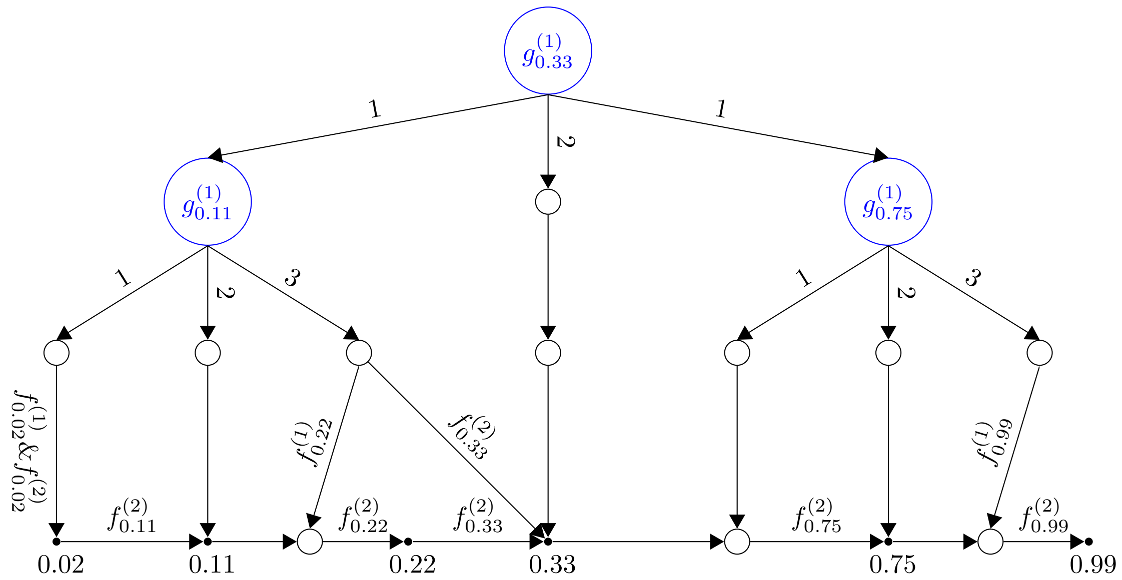


К задаче 2.2.5

имеет сложность 3 (а на остальных 4, как в пункте а). Средняя сложность равна $3 \cdot 0.004 + 4 \cdot 0.996 = 3.996$.

2.2.23 а) $X = Y = \{\overline{1,9}\}$, отношение поиска $x\rho y \Leftrightarrow (x = y)$. Базовое множество содержит только переключатели, соответствующие взвешиванию конфет с номерами $i_1, \ldots, i_k$ на левой чаше и $j_1, \ldots, j_k$ — на правой. $g_{i_1,\ldots,i_k \wedge j_1,\ldots,j_k}(x) = \begin{cases} 1, & x \in \{i_1, \ldots, i_k\} \\ 2, & x \notin \{i_1, \ldots, i_k\} \cup \{j_1, \ldots, j_k\} \\ 3, & x \in \{j_1, \ldots, j_k\} \end{cases}$. b) См. рис. b) См. рис. Средняя сложность $\hat{T} = \frac{4}{16} \cdot 2 + \frac{12}{16} \cdot 3 = 2.75$.

2.2.24 а) Самая левая фальшивая монета может занимать 2,3, или 4 позицию. Пусть $Y = \{2, 3, 4\}$, $X = \{\mathbf{x} = (x_1, x_2, x_3, x_4) \in \{0,1\}^4 : x_1 = 0, x_2 \leqslant x_3, x_4 = 1\} = \{(0,1,1,1); (0,0,1,1); (0,0,0,1)\}$ — считаем, что вес



К задаче 2.2.6

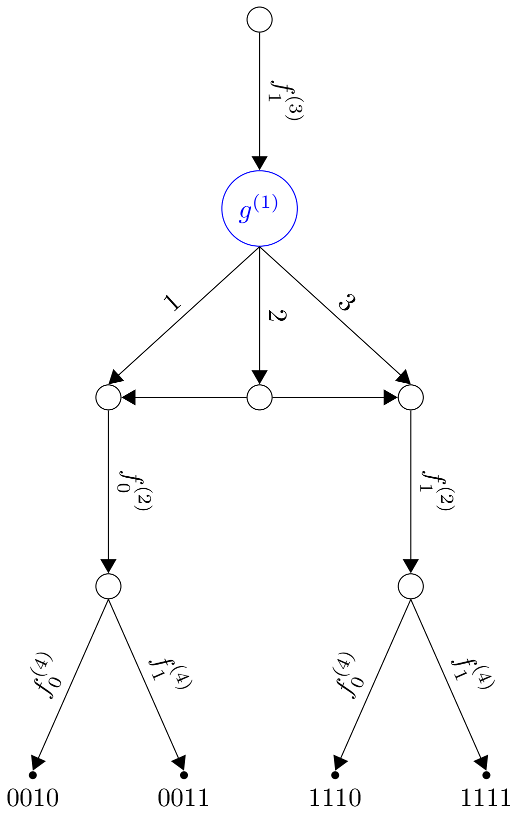
фальшивой монеты равен 0, а настоящей — 1. Базовое множество состоит из переключателей вида

$$g_{i_1,\ldots,i_k\wedge j_1,\ldots,j_k}(\mathbf{x}) = \begin{cases} 1, & x_{i_1}+\ldots+x_{i_k} < x_{j_1}+\ldots+x_{j_k}; \\ 2, & x_{i_1}+\ldots+x_{i_k} = x_{j_1}+\ldots+x_{j_k}; \\ 3, & x_{i_1}+\ldots+x_{i_k} > x_{j_1}+\ldots+x_{j_k}, \end{cases}$$

таких, что $\{i_1,\ldots,i_k\},\{j_1,\ldots,j_k\} \subset \{\overline{1,4}\}$ и $\{i_1,\ldots,i_k\}\cap\{j_1,\ldots,j_k\} = \emptyset$. b) Граф, решающий задачу указан на рисунке:

2.2.25 a) Будем считать, что соответствие между ключам и замками задается перестановкой $\sigma \in S_5$, т.е. i-й замок открывается $\sigma(i)$-м ключом. Тогда $X = Y = S_5$ — множество запросов и записей, $x\rho y \Leftrightarrow x = y$ — отношение поиска. Базовое множество состоит из переключателей (проверяющих, открывает ли i замок j-й ключ) $g_{i,j} = \begin{cases} 1, & j = x(i) \\ 2, & j \neq x(i), \end{cases}$ где $i,j \in \{\overline{1,5}\}$. b) Проверяем 1-й ключ — 4 проверки, 2-й — 3 проверки, 3-й — 2 проверки, 4-й — одна. Итого 4+3+2+1=10 проверок.

2.2.26 a) $Y = \{1,2,3,5\}$ — множество записей, $X = \{+,-\}\times Y = \{(\varepsilon,n) : \varepsilon = \pm, n \in Y\}$ — множество запросов (n — номер дефектной монеты), $(\varepsilon,n)\rho y \Leftrightarrow (n = y)$ — отношение поиска. Базовое множество состоит из переключателей вида $g_{S'\wedge S''}(x) = \begin{cases} 1, \sum_{S'} m_x(s) < \sum_{S''} m_x(s) \\ 2, \sum_{S'} m_x(s) = \sum_{S''} m_x(s) \\ 3, \sum_{S'} m_x(s) > \sum_{S''} m_x(s) \end{cases}$, где $m_x(s)$ — масса монеты достоинства s (при условии, что $x = (\varepsilon,n)$ — дефектная) и S', S'' — непересекающиеся подмножества Y. b) См. рис. c) Сложность по-



К задаче 2.2.8

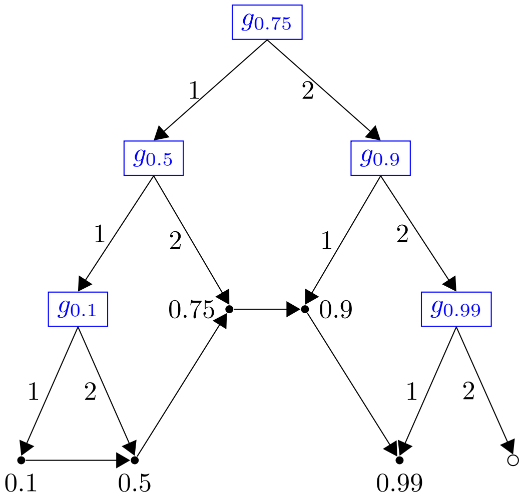
лученного графа равна 2, очевидно, что ИГ сложности 1 не может решить указанную задачу (не может быть более 3 листьев, а нужно, как минимум 4).

2.2.27 a) Пусть $W = \{\mathbf{w} = (w_1, \ldots, w_{15})\}$ — веса монет. Не ограничивая общности, можно считать, что веса нормированы так, чтобы вес настоящих монеты был равен 0, а вес фальшивой — ± 1. Тогда $X = W$, $Y = \{A, B\}$, отношение поиска $((x_1, \ldots, x_{15})\rho y) \Leftrightarrow \big((\exists x_i = -1 \& y = A) \vee (\exists x_i = 1 \& y = B)\big)$. Базовое множество состоит из переключателей вида

$$g_{i_1,\ldots,i_k \wedge j_1,\ldots,j_k}(\mathbf{x}) = \begin{cases} 1, & x_{i_1} + \ldots + x_{i_k} < x_{j_1} + \ldots + x_{j_k}; \\ 2, & x_{i_1} + \ldots + x_{i_k} = x_{j_1} + \ldots + x_{j_k}; \\ 3, & x_{i_1} + \ldots + x_{i_k} > x_{j_1} + \ldots + x_{j_k}, \end{cases}$$

таких, что $\{i_1, \ldots, i_k\}, \{j_1, \ldots, j_k\} \subset \{\overline{1, 15}\}$ и $\{i_1, \ldots, i_k\} \cap \{j_1, \ldots, j_k\} = \emptyset$.
b) См. рис. B-сложность равна 2.

2.2.28 a) Множество запросов и записей совпадают $X = Y = \{(y_1, y_2) \in \{\overline{1, 11}\}^2 \,|\, y_1 < y_2\}$. Отношение поиска $x \rho y \Leftrightarrow (x = y)$, т.е. это задача поиска идентичных объектов. Множество переключателей



К задаче 2.2.9

$g_S(y_1, y_2) = \begin{cases} 1, & \{y_1, y_2\} \cap S = \emptyset; \\ 2, & \{y_1, y_2\} \cap S \neq \emptyset, \end{cases}$ (где $S \neq \emptyset$); предикатов в базовом множестве нет. b) Поскольку итоговый граф будет слишком громоздкий не будем приводить его здесь, только опишем схему построения. Разобьем шары на 5 групп по два шара и одна — оставшийся шар. Проверяем каждую группу из двух шаров — 5 проверок. Возможны варианты: либо одна, либо две группы радиоактивны. В первом случае проверяем оба шара из группы — если радиоактивен только один, то оставшийся шар (11-й) — радиоактивен. Во втором случае проверяем по одному шару из каждой радиоактивной группы. Соответствующий ИГ строится следующим образом: первые 5 ярусов строим полное бинарное дерево. Из листьев выбираем 5 таких, где только одна проверка (из 5) положительна и 10 таких, где ровно две проверки положительны. К ним пристраиваем снизу соответствующие проверочные деревья. c) Двоичное дерево с 5 ярусами не может иметь более $2^5 = 32$ листьев. А множество записей состоит из $C_{11}^2 = 55$ элементов. d) Если в первой проверке участвует k шаров, то множество записей распадается на два подмножества: те записи, где среди этой группы нет радиоактивных шаров (их C_{11-k}^2) и где есть (их $C_{11}^2 - C_{11-k}^2$). Если можно построить ИГ с 6 ярусами, то оба этих множества должны содержать не более $2^5 = 32$ элементов. Составим таблицу

k	C_{11-k}^2	$C_{11}^2 - C_{11-k}^2$
1	45	10
2	36	19
3	28	27
4	21	34
5	15	40
6	10	45
...	...	...

Видно, что подходит только $k = 3$, предположим, что при проверке 3 шаров радиоактивных в группе нет. Тогда оба радиоактивных остались в кучке из 8 шаров. Надо выделить 2 радиоактивных за 5 проверок. Аналогично, пусть в первой про-

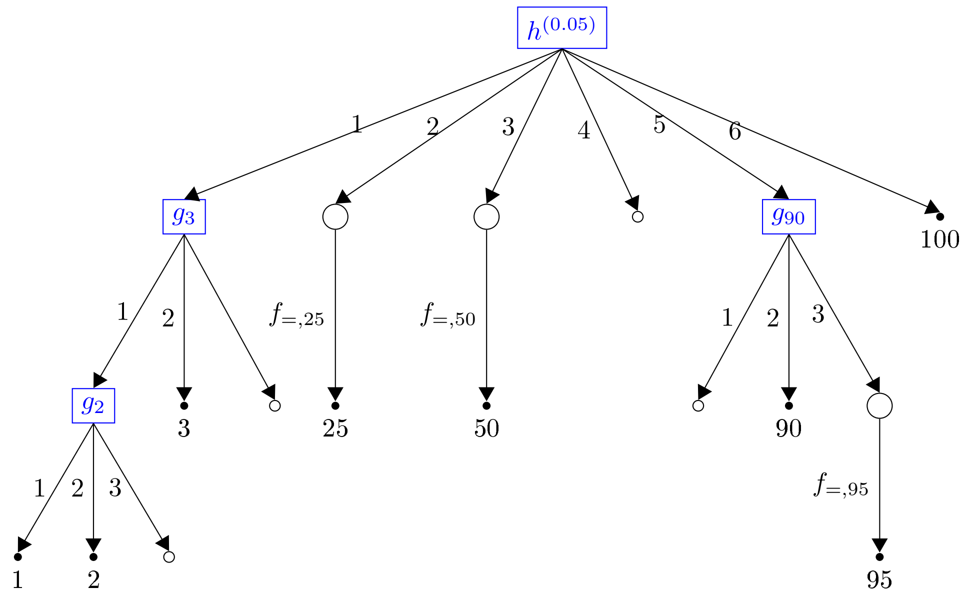


Рис. 2.12: К задаче 2.2.10.c)

верке участвует m шаров, разобьем множество соответствующих записей на два подмножества размером C_{8-m}^2 и $C_8^2 - C_{8-k}^2$. Составим такую же таблицу

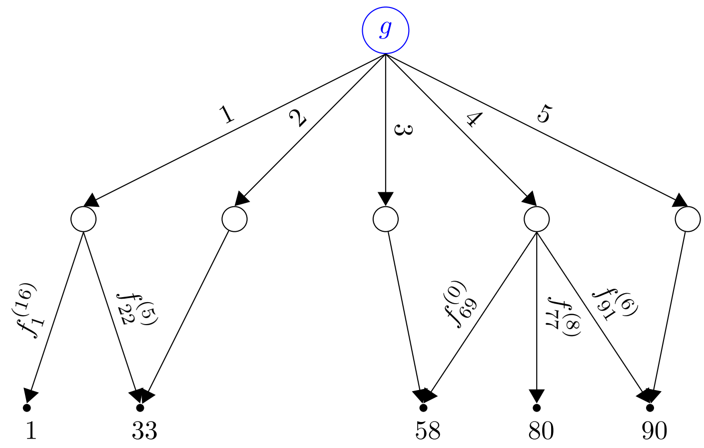
k	C_{8-k}^2	$C_8^2 - C_{8-k}^2$
1	21	7
2	15	13
3	10	18
4	6	22
...	...	...

Видно, что подходит только $k = 2$. Предположим, проверено 2 шара и из них нет радиоактивных. Остается 6 шаров. Для них построим такую же таблицу.

k	C_{6-k}^2	$C_6^2 - C_{6-k}^2$
1	10	5
2	6	9
3	3	12
4	1	14

. Видно, что не получается разбить на два подмножества, каждое из которых меньше 8, следовательно, такой ИГ построить нельзя.

2.2.29 а) Пусть заданы веса монет $\mathbf{w} = (w_1, \ldots, w_4)$. Нормируем веса монет, так, чтобы вес настоящих монет был равен 0, а вес фальшивой монеты ± 1 (в зависимости от того, легче она или тяжелее настоящей). Обозначим $S = \{(s_1, \ldots, s_4) : \exists j : s_i = \pm\delta_{ij}\}$, где $\delta_{ij} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}$. Пусть $X = S$, $Y = V = \{\overline{1,4}\}$ отношение $\mathbf{x}\rho i \Leftrightarrow (x_i \neq 0)$. Базовое множество состоит из переключателей вида $g_{I\wedge J}(\mathbf{s}) = \begin{cases} 1, & \sum_{i\in I} s_i < \sum_{j\in J} s_j; \\ 2, & \sum_{i\in I} s_i = \sum_{j\in J} s_j; \\ 3, & \sum_{i\in I} s_i > \sum_{j\in J} s_j, \end{cases}$ где $I, J \subset \{\overline{1,4}\}$ и



К задаче 2.2.12

$I \cap J = \emptyset$. Предикатов нет. b) См. рис. c) Тогда $X = Y = S$ и $x\rho y \leftrightarrow x = y$ — задача поиска идентичных объектов. d) Если взвешивать в начале по одной монете, то средняя ветка должна содержать 4 записи. Если сначала взвешивать по две монеты, то средней ветки не будет, а боковые содержат по 4 записи. В обоих случаях один переключатель не позволяет получить более 3 листьев.

2.2.30 a) Пусть заданы веса монет $\mathbf{w} = (w_1, \ldots, w_{12})$. Нормируем веса монет, так, чтобы вес настоящих монет был равен 0, а вес фальшивой монеты ± 1 (в зависимости от того, легче она или тяжелее настоящей). Обозначим $S = \{(s_1, \ldots, s_2) : \exists j : s_i = \pm\delta_{ij}\}$, где $\delta_{ij} = \begin{cases} 1, & i = j \\ 0, & i \neq j \end{cases}$. Пусть $X = Y = V = S$, отношение $x\rho y \Leftrightarrow (x = y)$ — т.е. задача поиска идентичных объектов. Заданы переключатели $g_{i_1,\ldots,i_k|j_1,\ldots,j_k}(\mathbf{s}) = \begin{cases} 1, & s_{i_1} + \ldots + s_{i_k} < s_{j_1} + \ldots + s_{j_k}; \\ 2, & s_{i_1} + \ldots + s_{i_k} = s_{j_1} + \ldots + s_{j_k}; \\ 3, & s_{i_1} + \ldots + s_{i_k} > s_{j_1} + \ldots + s_{j_k}, \end{cases}$ где $\{i_1, \ldots, i_k\}, \{i_1, \ldots, i_k\} \subset \{\overline{1, 12}\}$ и $\{i_1, \ldots, i_k\} \cap \{i_1, \ldots, i_k\} = \emptyset$. Предикатов нет. b) См. рис. Для краткости, вместо вектора $(s_1, ..., s_{12})$ на рисунке указаны только номер фальшивой монеты и соответствующий знак. c) $T_B = \widehat{T} = 3$.

2.2.31 a) В качестве X рассмотрим множество векторов $\{(x_1, \ldots, x_5) : x_i = 0, 1, -1\}$, у которых ровно одна компонента равна 1, ровно одна -1, а оставшиеся три — нули. В качестве Y выберем множество пар (y_+, y_-), а отношение поиска зададим как $(x_1, ..., x_5)\rho(y_+, y_-) \Leftrightarrow (x_{y_+} = 1)\&(x_{y_-} = -1)$. В качестве базового множества рассмотрим переключатели

$$g_{i_1\ldots i_k;j_1\ldots j_k}(\bar{x}) = \begin{cases} 1, & x_{i_1} + \ldots + x_{i_k} > x_{j_1} + \ldots + x_{j_k}; \\ 2, & x_{i_1} + \ldots + x_{i_k} = x_{j_1} + \ldots + x_{j_k}; \\ 3, & x_{i_1} + \ldots + x_{i_k} < x_{j_1} + \ldots + x_{j_k}. \end{cases}$$

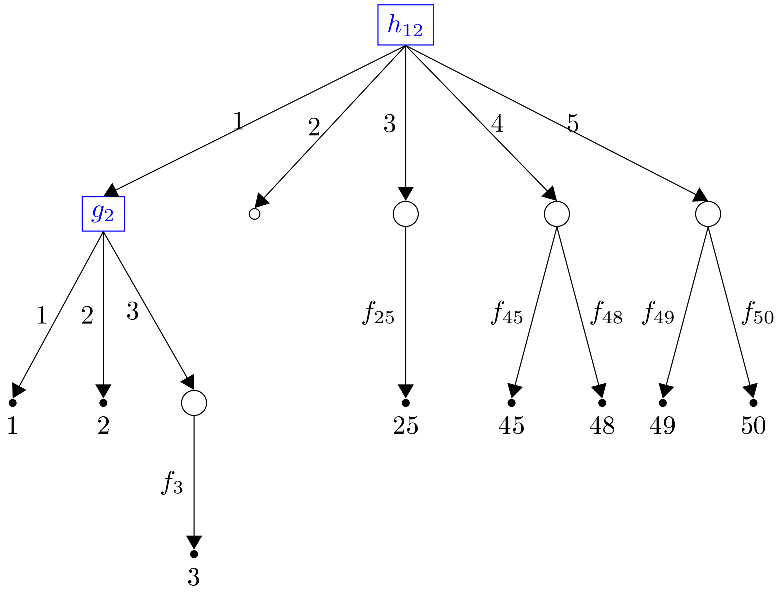


Рис. 2.14: К задаче 2.2.18

b) Поскольку информационный граф довольно громоздок, приведем словесное описание: Первый раз положим на чаши весов первую и вторую монеты, а второй раз — третью и четвёртую. Возможны только два случая. 1) Один раз весы были в равновесии (пусть при первом взвешивании; при этом на чашах настоящие монеты), а другой раз — нет. Возьмем настоящую монету из первого взвешивания и сравним её с той, что оставалась на столе. Если их веса равны, то последняя монета настоящая, а фальшивые - те, что участвовали во втором взвешивании. Иначе, монета со стола — фальшивая, и мы знаем, легче она настоящей или тяжелее, а потому знаем, лёгкая или тяжёлая фальшивая монета участвовала во втором взвешивании. 2) Оба раза весы были не в равновесии. Тогда на весах каждый раз была одна фальшивая монета, а на столе осталась настоящая. Взвесим её с лёгкой монетой из первого взвешивания. Если веса равны, то в первом взвешивании фальшивой была более тяжёлая, а во втором - более лёгкая. Если же более лёгкая монета из первого взвешивания оказалась легче, то она фальшивая, а из второго взвешивания фальшивая — более тяжёлая. c) Граф сложности 2 не может дать более 9 различных ответов, а в данной задаче $|Y| = 20$.

2.2.32 a) Пусть S_5 — множество всех перестановок из 5 элементов. Для массива существует перестановка, которая переставляет его элементы в правильном порядке (т.е. $x_i < x_j \Leftrightarrow \sigma(i) < \sigma(j)$). Например, для массива $(10, 1, 3, 2, 8)$ такой перестановкой будет $\sigma = (5, 1, 3, 2, 4)$. Тогда задача упорядочивания массива может быть сформулирована как $I = \langle S_5, S_5, = \rangle$ — задача поиска идентичных объектов. В качестве базового множества возьмем множество переключателей $g_{ij}(x) = \begin{cases} 1, & x_i < x_j; \\ 2, & x_i > x_j. \end{cases}$, где $i, j = 1, ..., 5$, $i \neq j$. b) Решением будет ИГ содержащий $5! = 120$ листьев. Опишем ал-

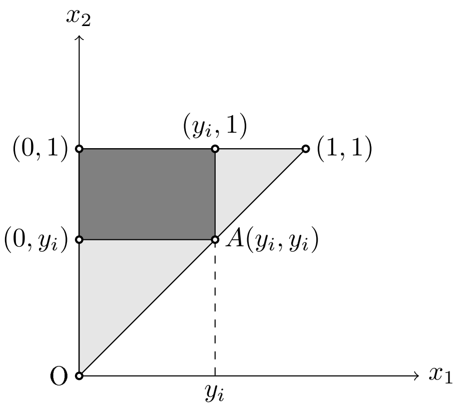
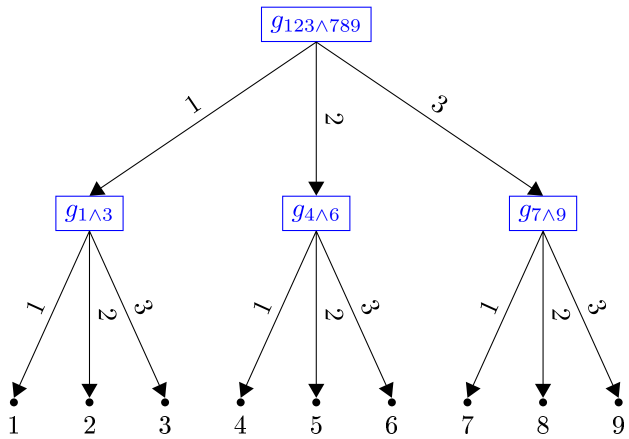


Рис. 2.15: К задаче 2.2.19

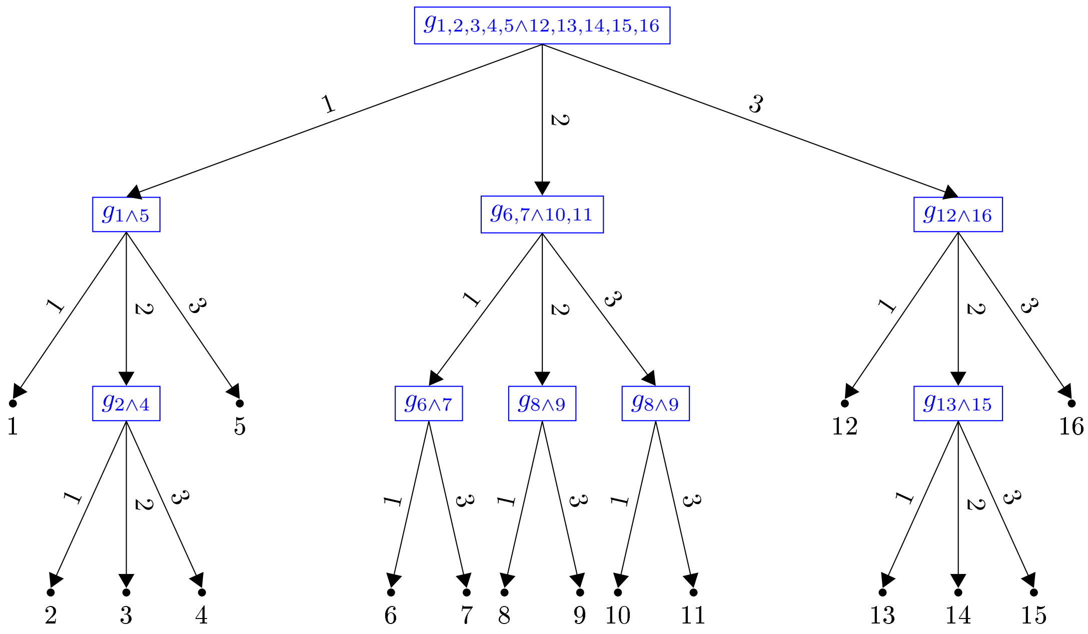


К задаче 2.2.23.b)

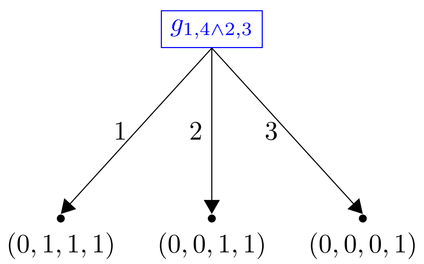
горитм взвешивания, соответствующий этому графу. Сначала упорядочим x_1 и x_2 (1-е взвешивание), потом x_3 и x_4 (2-е взвешивание). Таким образом $x_1 < x_2$ и $x_3 < x_4$. Сравним x_1 и x_3 (3-е взвешивание), упорядочим, так, что $x_1 < x_3$ (если надо то x_2 и x_4 тоже поменяем местами). Потом сравним x_3 и x_5 (4-е взвешивание).

I) Допустим, что $x_3 < x_5$. Тогда сравним x_4 и x_5(5-е взвешивание). Если $x_4 < x_5$, то получаем: $x_1 < x_3 < x_4 < x_5$ и $x_1 < x_2$. Для того, чтобы установить позицию x_2 достаточно еще 2 взвешиваний. Случай $x_4 > x_5$ рассматривается аналогично.

II) Допустим, что $x_3 > x_5$. Тогда сравним x_2 и x_3 (5-е взвешивание). Если $x_2 < x_3$, то получаем в итоге $x_1 < x_2 < x_3 < x_4$ и $x_5 < x_3$, остается два взвешивания, чтобы поместить x_5 на нужное место. Если же $x_2 > x_3$, то сравним x_2 и x_4 (6-е взвешивание) и x_1 с x_5 (7-е взвешивание). По результатам упорядочиваем массив. c) Граф сложности 6 не может содержать более $2^6 = 64$ концевых вершин. d) Очевидно, что если граф содержит $N!$ концевых вершин, то его глубина не менее $\log N! \sim \log\left(\frac{n}{e}\right)^n \sim N\log N$. С другой стороны, можно показать (индукцией по n), что массив размера 2^n можно



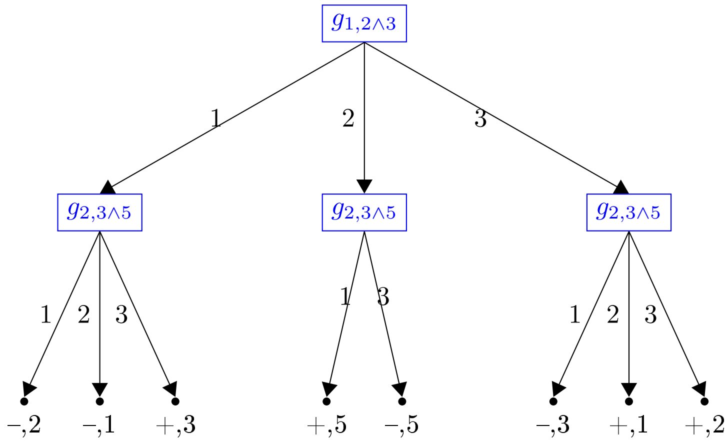
К задаче 2.2.23.c)



К задаче 2.2.24

отсортировать за $O(n2^n)$ сравнений.

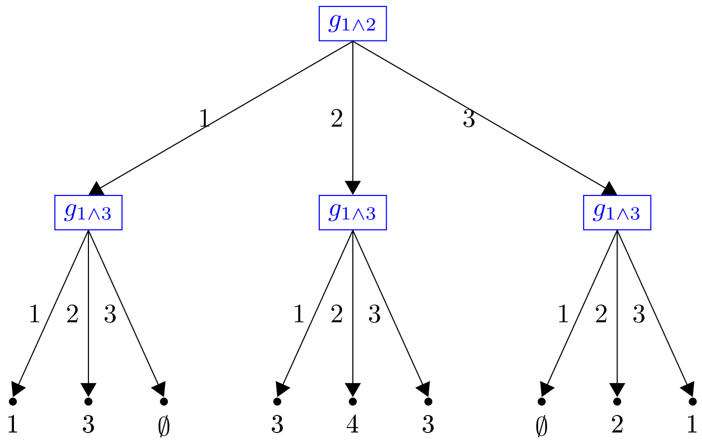
2.2.33 a) Обозначим $E_N = \{0, 1, \ldots, N-1\}$. В качестве множества запросов и множества ответов выберем $X = Y = 2^{E_N}$ — множество всех подмножеств E_N. В качестве отношения ρ возьмем равенство — получим задачу поиска идентичных объектов. В качестве базового множества выберем множество предикатов вида $f_A(B) \Leftrightarrow (A \cap B \neq \emptyset)$, где $A, B \subset E_N$. Этот предикат соответствует проверке образцов из множества A и дает положительную реакцию, если хотя бы один содержит вирус. Вероятностная мера на X задана следующим образом: $\mathcal{P}(A) = p^{|A|} \cdot (1-p)^{N-|A|}$. b) Очевидно, для проверки каждой группы нужно вычислить один предикат, т.е. сложность k. Если группа дала положительную реакцию, то надо проверить каждый образец в ней — сложность равна n. Вероятность этого равна $1-(1-p)^n$. Таким образом, получаем среднюю сложность $S = k + n(1-(1-p)^n)$. c) Зафиксируем N и будем менять n. Получим $S(n) = \frac{N}{n} + n - n \cdot (1-p)^n$. Найдем производ-



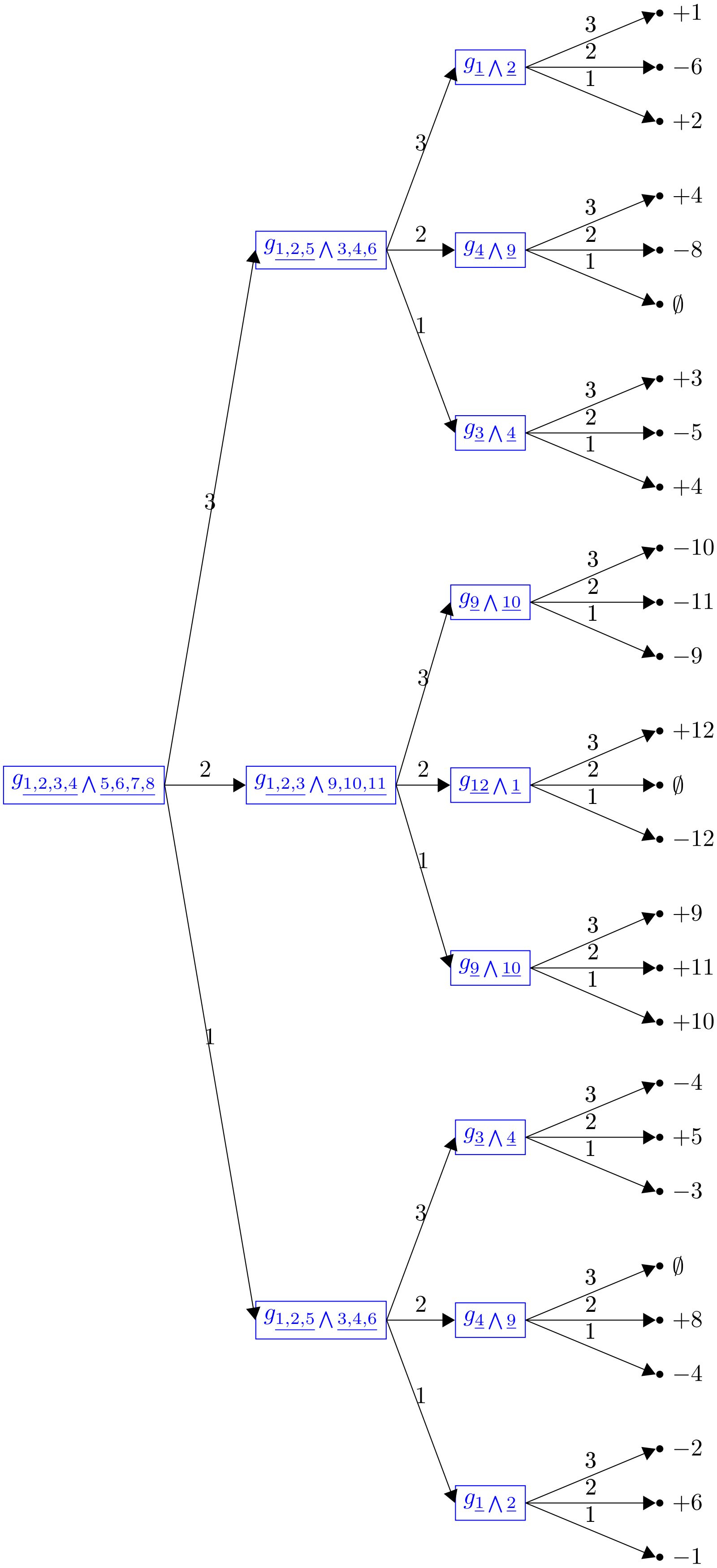
К задаче 2.2.26)

К задаче 2.2.27)

ную: $S'(x) = -\frac{N}{x^2} + 1 - (1-p)^x\,(1 + x\ln(1-p))$. Поскольку p мало, можно разложить в ряд Тейлора до 2-го члена: $S'(x) = -\frac{N}{x^2} + 2xp + \bar{o}(p)$. Приравнивая к нулю главную часть разложения, получим $x = \sqrt[3]{N/2p}$. Итак, наименьшее значение достигается при $n = \left[\sqrt[3]{\frac{N}{2p}}\right]$. Подставляя $N = 1600$, $p = 10^{-4}$, получим $n = 200$. Т.е. надо смешивать по 200 образцов.



К задаче 2.2.29



К задаче 2.2.30

Глава 3

Логический подход

3.1. Вывод из аксиом

Основные обозначения и определения

Существует много различных дедуктивных систем для описания общезначимых формул логики высказываний, отличающихся множеством используемых логических связок и выбором аксиом; приведем здесь одну из простейших таких систем в которой рассматриваются лишь две логические связки — $\neg$, $\rightarrow$. Остальные логические связки легко могут быть выражены через них и трактоваться как своего рода "сокращенные обозначения"для соответствующих формул с $\neg$ и $\rightarrow$. Аксиомы данной дедуктивной системы задаются при помощи так называемых *схем аксиом.*

A1) Если f, g — формулы, то формула

$$(f \rightarrow (g \rightarrow f))$$

есть аксиома.

A2) Если f, g, h — формулы, то формула

$$((f \rightarrow (g \rightarrow h)) \rightarrow ((f \rightarrow g) \rightarrow (f \rightarrow h)))$$

есть аксиома.

A3) Если f, g — формулы, то формула

$$((\neg f \rightarrow \neg g) \rightarrow ((\neg f \rightarrow g) \rightarrow f))$$

есть аксиома.

В действительности каждая из схем аксиом A1, A2, A3 определяет бесконечное множество аксиом, отличающихся выбором формул f, g, h.

Правило вывода в этой дедуктивной системе одно — если уже получены общезначимые формулы f и $(f \rightarrow g)$, то из них выводится общезначимая формула g (согласно классификации предложенной еще Аристотелем, это правило называется modus ponens).

Чтобы проиллюстрировать технику, применяемую при изучении дедуктивных систем, приведем схематические наброски доказательств некоторых важных свойств только, что определенной дедуктивной системы.

Выводом формулы f из списка формул $\Gamma = g_1, \dots, g_n$ будем называть произвольную последовательность формул $f_1, f_2, \dots, f_m = f$, такую, что каждое $f_i(i = 1, \dots, m)$ есть либо аксиома, либо элемент списка Γ, либо получено по правилу modus ponens из некоторых f_j, f_k, при $j, k \in \{1, \dots, i-1\}$. Если существует вывод f из Γ, то обозначаем это обстоятельство посредством $\Gamma \vdash f$ (в частности, список Γ может быть пуст; те формулы, которые выводимы из пустого Γ, как легко можно видеть, образуют класс всех *выводимых формул* нашей дедуктивной системы). Следующая лемма есть пример выводимой формулы.

Лемма 3. *Если f — формула, то* $\vdash (f \to f)$.

Имеет место следующее важное утверждение, известное как теорема дедукции (Эрбран).

Теорема 5 (О дедукции)**.** *Если Γ — список формул и f, g — формулы, то из $\Gamma, f \vdash g$ вытекает $\Gamma \vdash (f \to g)$.*

В дальнейшем для краткости будем обозначать ее «ТоД».

Из нее вытекает следующая лемма:

Лемма 4 (Доказательство от противного)**.** *Пусть Γ — список формул и f, g — формулы. Если Γ, $\neg f \vdash g, \neg g$, то $\Gamma \vdash f$.*

Очевидно, что любая формула, выведенная из A1, A2, A3 является общезначимой. Оказывается, верно и обратное утверждение:

Теорема 6 (О полноте исчисления высказываний)**.** *Формула исчисления высказываний общезначима тогда и только тогда, когда она выводима в дедуктивной системе со схемами аксиом A1, A2, A3 и правилом вывода* modus ponens.

Упражнения.

Задача 3.1.1. Выразить следующие формулы используя только операции импликации ($\to$) и отрицания ($\neg$).

a) $a \vee b$;
b) $a \& b$;
c) $a \leftrightarrow (b \vee c)$;
d) $m(a, b, c) = ab \oplus ac \oplus bc$.

Задача 3.1.2. Показать выводимость $\Gamma, A_1, A_2, A_3 \vdash F$, используя теорему о дедукции:

a) $\Gamma = \{a \to b, \neg b\}$, $F = \neg a$ (*reductio ad absurdum*);
b) $\Gamma = \{a \to b, b \to c\}$, $F = a \to c$ (правило силлогизма);
c) $\Gamma = \{a \to b, b \to c, \neg c\}$, $F = \neg a$;

Задача 3.1.3. Построить явный вывод (т.е. не используя ТоД) $\{A_1, A_2, A_3, a \to b, b \to c\} \vdash a \to c$.

Задача 3.1.4. Построить вывод из аксиом и леммы о выводе $f \to f$ не используя теорему о дедукции. В случае необходимости предварительно привести к виду, содержащему только импликации и отрицания.

a) $\{A \vee B, \neg A\} \vdash B$ (разбор случаев);
b) $\vdash (\neg A \to A) \to A$ (доказательство от противного);

Задача 3.1.5. Доказать, используя ТоД: $a \to (b \to c) \vdash b \to (a \to c)$;

Задача 3.1.6. Доказать, используя используя ТоД и ЛДоП:

a) $\neg\neg B \to B$ (снятие двойного отрицания);
b) $B \to \neg\neg B$ (навешивание двойного отрицания).

Задача 3.1.7. Доказать (используя ТоД и ЛДоП):

a) $\neg a \to (a \to b)$;
b) $(\neg b \to \neg a) \to (a \to b)$;
c) $(a \to b) \to (\neg b \to \neg a)$;
d) $(\neg(a \to b)) \to a$.

Задача 3.1.8. Доказать используя ТоД и ЛДоП (предварительно привести к виду, содержащему только импликации и отрицания):

a) $\{a, b\} \vdash a\&b$;
b) $a\&b \vdash \{a, b\}$.

Задача 3.1.9. Показать выводимость $\Gamma, A_1, A_2, A_3 \vdash F$, используя ТоД и ЛДоП (предварительно преобразовав к виду, содержащему только импликации и отрицания):

a) $\Gamma = \{a \to b, b \to c\}$, $F = a \to (b\&c)$;
b) $\Gamma = \{a \to b, a \oplus b\}$, $F = b$;
c) $\Gamma = \{a \oplus b, a\}$, $F = \neg b$ (разбор взаимоисключающих случаев);

Задача 3.1.10. Доказать, используя ТоД и ЛДоП.

a) $a \to (\neg b \to \neg(a \to b))$;
b) $(a \to b) \to ((\neg a \to b) \to b)$.

Задача 3.1.11. Привести формулу $a \to (b \to (a\&b))$ к виду, содержащему только импликации и отрицания и построить вывод, используя ТоД.

Задача 3.1.12. Привести к виду, содержащему только импликации и отрицания и доказать с помощью ТоД и ЛДоП: $\big((a \to b)\&(b \to c)\big) \to (a \to c)$ (правило силлогизма).

Задача 3.1.13. Доказать выводимость $\Gamma, A_1, A_2, A_3 \vdash F$ или показать невозможность такого вывода с помощью теорем о дедукции и о полноте исчисления высказываний.

a) $\Gamma = \{a \to b, \neg a\}$, $F = \neg b$;
b) $\Gamma = \{a \to b, \neg a \to \neg b\}$, $F = a \leftrightarrow b$;
c) $\Gamma = \{a \to b, \neg b \to \neg a\}$, $F = a \leftrightarrow b$;
d) $\Gamma = \{a \to b, a \vee b\}$, $F = a$;

e) $\Gamma = \{a \to b, b \to a, a \vee b\}$, $F = a\&b$;

Задача 3.1.14. Проверить выводимость или невыводимость $\Gamma, A_1, A_2, A_3 \vdash \vdash F$.

a) $\Gamma = \{a \to (b \to c), (a \to b) \to c\}$, $F = b \to c$;
b) $\Gamma = \{a \to (b \to c), (a \to b) \to c\}$, $F = a \to c$;
c) $\Gamma = \{a \to b\&c, b \to c\&a, c \to a\&b, a \vee b \vee c\}$, $F = a\&b\&c$;
d) $\Gamma = \{(a \vee b) \to c, (b \vee c) \to a, (c \vee a) \to b, a \vee b \vee c\}$, $F = a\&b\&c$;

Задача 3.1.15. Пусть формула F не общезначима. Возьмем ее в качестве схемы аксиом $\hat{A}$, т.е. будем считать аксиомами все формулы, которые получаются из F подстановками формул на место пропозициональных переменных.

a) Доказать, что теория, построенная на основе $A_1, A_2, A_3, \hat{A}$ будет противоречивой (т.е. возможен вывод формул B и $\neg B$).
b) Доказать, что в такой теории выводима любая формула.

Задача 3.1.16. Доказать независимость аксиомы A_3 от A_1 и A_2.

Ответы и указания.

3.1.1 a) $a \vee b = (\neg a) \to b$; b) $a\&b = \neg(a \to \neg b)$; c) $a \leftrightarrow (b \vee c) = \neg\big((a \to (\neg b \to \to c)\big) \to \neg\big((\neg b \to c) \to a)\big)$. d) $ab \oplus ac \oplus bc = ((\neg a \to b) \to \neg c) \to \neg(a \to \neg b)$.

3.1.2 a) Заметим, что $\Gamma, \neg\neg a \vdash b, \neg b$ и применим теорему о дедукции, получим, что $\Gamma \vdash \neg\neg a \to b, \neg\neg a \to \neg b$. Тогда верна следующая цепочка вывода: $\neg\neg a \to b$ (т. о дедукции); $\neg\neg a \to \neg b$ (т. о дедукции); $(\neg\neg a \to \neg b) \to ((\neg\neg a \to \to b) \to \neg a)$ (схема A_3); $(\neg\neg a \to b) \to \neg a$ (М.Р.); $\neg a$ (М.Р.). b) Заметим, что $\Gamma, a \vdash c$, действительно, a(дано); $a \to b$ (дано); b (М.Р.); $b \to c$ (дано); c (М.Р.). Следовательно, по т. о дедукции $\Gamma \vdash a \to c$. c) Получается комбинацией пунктов «a» и «b». Действительно, в пункте «b» показано, что $a \to b, b \to c \vdash a \to c$. Применяя пункт «a», получим, что $a \to c, \neg c \vdash \neg a$.

3.1.3 Вывод:
(1) $(b \to c) \to (a \to (b \to c))$ (схема A_1, $f = (b \to c)$, $g = a$);
(2) $b \to c$ (дано);
(3) $a \to (b \to c)$ (М.Р. 1-2);
(4) $(a \to (b \to c)) \to ((a \to b) \to (a \to c))$ (схема A_2);
(5) $(a \to b) \to (a \to c)$ (М.Р. 3-4);
(6) $a \to b$ (дано);
(7) $a \to c$ (М.Р. 5-6).

3.1.4 a) Напомним, что $A \vee B = (\neg A) \to B$. Тогда цепочка вывода: $(\neg A) \to \to B$ (дано), $\neg A$ (дано), B (М.Р.). b) Верна следующая цепочка вывода: $(\neg A \to \neg A) \to ((\neg A \to A) \to A)$ (A_3, при $f = g = A$); $\neg A \to \neg A$ (лемма); $(\neg A \to A) \to A)$ (М.Р.).

3.1.5 Поскольку $a, b, a \to (b \to c) \vdash c$, то по ТоД $b, a \to (b \to c) \vdash a \to c$. Применим ТоД еще раз, получим требуемое утверждение.

3.1.6 a) Очевидно, что $\neg\neg B, \neg B \vdash \neg B, \neg\neg B$. Применив лемму о доказательстве от противного, получим $\neg\neg B \vdash B$. По теореме о дедукции получаем $\neg\neg B \to B$. b) Воспользуемся пунктом «a», применив его к $\neg B$, получаем $\neg\neg\neg B \to \neg B$, т.е. $B, \neg\neg\neg B \vdash B, \neg B$. Тогда по лемме о доказательстве от противного $B \vdash \neg\neg B$. применив теорему о дедукции, получаем $B \to \neg\neg B$.

3.1.7 a) $a, \neg a, \neg b \vdash a, \neg a$. По лемме о доказательстве от противного $a, \neg a \vdash b$. Применяя два раза теорему о дедукции, получим $\neg a \to (a \to b)$;
b) Рассмотрим систему $(\neg b \to \neg a)$, $\neg b$, a. Из нее выводимо a и $\neg a$, следовательно, по лемме от противного из $(\neg b \to \neg a)$, a выводимо b. Применим теорему о дедукции, получим $(\neg b \to \neg a) \vdash a \to b$. Применяя теорему о дедукции еще раз, получим $\vdash (\neg b \to \neg a) \to (a \to b)$.
c) Решается аналогично пункту «b», с применением формулы $a \to \neg\neg a$.
d) Рассмотрим систему $(\neg(a \to b)), \neg a$. По пункту «a» и modus ponens выводим $a \to b$. Применяя ЛДоП, получим $(\neg(a \to b)) \vdash a$, откуда по ТоД $\vdash (\neg(a \to b)) \to a$.

3.1.8 a) Напомним, что $a\&b$ есть сокращение для $\neg(a \to \neg b)$. Из системы $a, b, \neg\neg(a \to \neg b)$ вывести b и $\neg b$, следовательно, по лемме о доказательстве от противного $a, b \vdash \neg(a \to \neg b)$. b) Тот факт, что $\neg(a \to \neg b) \vdash a$ вытекает из пункта «d» задачи 3.1.7. Рассмотрим систему $\Gamma = \{\neg(a \to \neg b), \neg b\}$. Построим вывод $\Gamma \vdash \neg b \to (a \to \neg b)$ (A1); $(a \to \neg b)$ (M.P.). Итак, вывели $(a \to \neg b)$ и $\neg(a \to \neg b)$, следовательно, по ЛДОП $\neg(a \to \neg b) \vdash b$.

3.1.9 a) Преобразуем $a \to (b\&c) = a \to \neg(b \to \neg c)$. Рассмотрим систему: $\Gamma' = \{a, a \to b, b \to c, \neg\neg(b \to \neg c)\}$. Тогда $\Gamma' \vdash c$ и $\Gamma' \vdash \neg c$, следовательно, по ЛДоП $a, a \to b, b \to c \vdash \neg(b \to \neg c)$. По ТоД получаем $a \to b, b \to c \vdash a \to$ $\to \neg(b \to \neg c)$.
b) Преобразуем $a \oplus b = (a \to b) \to \neg(b \to a)$. Тогда $(a \to b), (a \to b) \to \neg(b \to$ $\to a) \vdash \neg(b \to a)$ (M.P.). Воспользуемся пунктом «d» задачи 3.1.7 (только поменяем a и b местами), получим b.
c) Преобразуем $a \oplus b = (a \to b) \to \neg(b \to a)$. Рассмотрим систему $\Gamma_1 = \{(a \to$ $\to b) \to \neg(b \to a), a, \neg\neg b\}$. Выводим b (M.P.), $a \to (b \to a)$ (A1), $(b \to a)$ (M.P.), $b \to (a \to b)$ (A1), $a \to b)$ (M.P.), $\neg(b \to a)$ (M.P.). Итак, мы вывели $b \to a$ и $\neg(b \to a)$, следовательно, по ЛДоП $\{(a \to b) \to \neg(b \to a), a\} \vdash \neg b$.

3.1.10 a) Из системы a, $\neg\neg(a \to b)$ выводимо b. Применив к системе a, $\neg b$, $\neg\neg(a \to b)$ лемму о доказательстве от противного, получим a, $\neg b \vdash \neg(a \to b)$. Применяя два раза теорему о дедукции, получим $\vdash a \to (\neg b \to \neg(a \to$ $\to b))$. b) Рассмотрим систему $\{(a \to b), (\neg a \to b), \neg b\}$. Как показано в пункте «c» задачи 3.1.7 $(a \to b) \vdash (\neg b \to \neg a)$. Применяя M.P., выводим $\neg a$, применяя еще раз, получим b. Итак, получены b и $\neg b$, слеодвательно, по ЛДОП $\{(a \to b), (\neg a \to b) \vdash b\}$. Применяя два раза ТоД, получаем требуемое утверждение.

3.1.11 Приведем формулу к виду $a \to (b \to \neg(a \to \neg b))$. Воспользуемся задачей 3.1.8, т.е. $a, b \vdash \neg(a \to \neg b))$. Применяя два раза ТоД, получаем требуемую формулу.

3.1.12 Приведем к стандартному виду: $\neg\big((a \to b) \to \neg(b \to c)\big) \to (a \to c)$. Воспользуемся пунктом «b» задачи 3.1.8, получим $\neg\big((a \to b) \to \neg(b \to$ $\to c)\big) \vdash \{(a \to b), (b \to c)\}$. Применив пункт «b» задачи 3.1.2, получаем

$\neg\big((a \to b) \to \neg(b \to c)\big) \vdash a \to c$. Применив ТоД, получим требуемую формулу.

3.1.13 a) Если бы вывод был возможен, то (по т. о дедукции) была бы выводима формула $(a \to b)\&(\neg a) \to \neg b$. Эта формула не общезначима (можно взять $a =$ Л, $b =$ И), следовательно, вывод невозможен. b) Поскольку, как легко проверить, формула $(a \to b) \to ((\neg a \to \neg b) \to (a \leftrightarrow b))$ общезначима, то она (по теореме о полноте) выводима, следовательно $(a \leftrightarrow b)$ может быть получена двукратным применением modus ponens. c,d) Вывод невозможен, доказывается аналогично пункту «a». e) Выводимость можно показать аналогично пункту «b».

3.1.14 a) Выводимо. b) Не выводимо. c) Выводимо. d) Выводимо. Решается аналогично задаче 3.1.13.

3.1.15 a) Рассмотрим набор $x_1, ..., x_n$, на котором F обращается в Л. Подставим $(f \to f)$ вместо тех переменных, которые принимают значение И и $\neg(f \to f)$ вместо остальных. Очевидно, полученная формула будет тождественно ложной $F(f_1, ..., f_n) =$ Л. Следовательно, $\neg F(f_1, ..., f_n) =$ И — общезначима. По теореме о полноте она выводима из A_1, A_2, A_3.

3.1.16 Заметим, что только схема A_3 содержит отрицания. Можно построить формальную модель, в которой импликация имеет свое обычное значение, а $\neg x = x$. Тогда в этой модели схемы A_1 и A_2 всегда будут выполнены, а A_3 — нет.

3.2. Исчисление высказываний

Обозначения и определения

В случае логики высказываний проверка общезначимости формулы, имеющей n переменных, может быть осуществлена путем перебора всех 2^n возможных наборов логических констант, сопоставляемых этим переменным при различных интерпретациях, и установлении того, что все значения формулы на этих наборах равны И. Таким образом здесь имеется не только алгоритм перечисления всех общезначимых формул, но и алгоритм проверки общезначимости. Однако даже для такого простейшего случая остается проблема уменьшения трудоемкости распознавания общезначимости по сравнению с практически малоприемлимым для сколь-нибудь значительных n перебором всех 2^n наборов значений переменных. К проблеме распознавания общезначимости формул логики высказываний тесно примыкает проблема решения систем логических уравнений сводящаяся к нахождению всех наборов значений переменных при которых заданная формула алгебры логики принимает значение И. Последняя проблема часто встречается в различных прикладных задачах дискретной оптимизации и диагностики, и поиску эффективных процедур решения ее, учитывающих в своих эвристических решающих правилах статистические особенности рассматриваемого класса задач, посвящено большое количество исследований. Мы ограничимся здесь лишь несколькими простейшими процедурами подобного рода (применительно к задаче распознавания общезначимости), поскольку они позволят нам развить технику доказательства теорем в логике высказываний, представляющую собой упрощенный аналог техники, развиваемой далее для автоматического доказательства теорем в логике предикатов.

Алгоритм Квайна

Прежде всего, опишем *алгоритм Квайна*, осуществляющий проверку общезначимости формулы f логики высказываний при помощи построения так называемого семантического дерева. Корню этого дерева — исходной вершине — приписывается формула f. Пусть уже построено некоторое подмножество вершин семантического дерева, каждой из которых сопоставлена некоторая формула. Если каждой концевой вершине данного дерева оказалась сопоставлена константа И, то процесс завершается, и формула f является общезначимой. Если некоторой концевой вершине дерева сопоставлена константа Л, то формула f не общезначима, и процесс также завершается. В противном случае найдется концевая вершина v, которой сопоставлена формула отличная от констант И, Л. Если эта формула g не содержит переменных, то ее можно преобразовать в логическую константу, используя тождества:

$$\neg\text{И} = \text{Л};\ \neg\text{Л} = \text{И};\ \text{И} \vee f = \text{И};\ \text{Л} \vee f = f;\ \text{И} \wedge f = f;\ \text{Л} \wedge f = \text{Л};\ \text{И} \to f = f;$$
$$\text{Л} \to f = \text{И};\ f \to \text{И} = \text{И};\ f \to \text{Л} = \neg f;\ \text{И} \leftrightarrow f = f;\ \text{Л} \leftrightarrow f = \neg f.$$

Если же она содержит хотя бы одну переменную x, и выполняются следующие действия:

а) Находятся результаты g_1 и g_2 подстановки в формулу g вместо переменной x, соответственно констант И и Л.

б) Находятся результаты g_1' и g_2' упрощения формул g_1 и g_2 при помощи перечисленных выше тождеств, применяемых до тех пор пока это возможно.

в) вводятся две новые вершины v_1 и v_2 семантического дерева, которым приписываются, соответственно, формулы g_1' и g_2'. К этим вершинам от вершины v проводятся ребра, отмеченные соответственно выражениями $x =$ $=$ И и $x =$ Л.

Далее повторяется описанный выше процесс рассмотрения концевых вершин семантического дерева.

Выбор конкретной переменной x формулы g, по которой проводится "разбор случаев осуществляется на основании различных эвристических решающих правил. Простейшим таким правилом является выбор переменной, имеющей наибольшее число вхождений в g, для получения наиболее сильного упрощения при переходе к g_1' и g_2', другим полезным соображением является такой выбор переменных x, при котором сопоставленные концевым вершинам семантического дерева формулы g оказывались бы представимы, например, как $h_1 \wedge h_2$ либо $h_1 \vee h_2$, где формулы h_1 и h_2 не имеют общих переменных. В этом случае вместо построения ветви дерева для $h_1 \wedge h_2$ ($h_1 \vee h_2$, $h_1 \to h_2$ и т.п.) можно было бы рассмотреть независимо формируемые деревья для h_1, h_2 и из анализа их концевых вершин сделать вывод относительно общезначимости g.

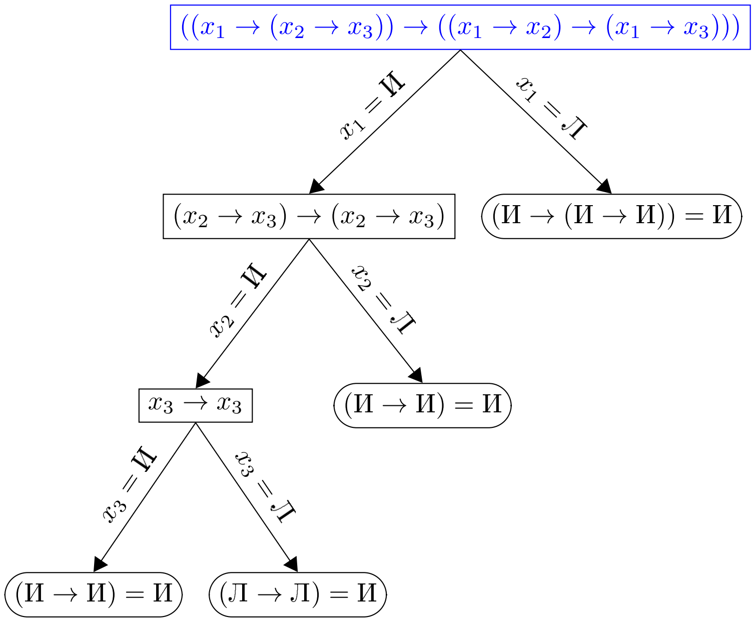


Рис. 3.1: Алгоритм Квайна (ф-ла общезначима).

Пример 1. В качестве примера применения алгоритма Квайна убедимся в общезначимости формулы, полученной из схемы аксиом A2:

$$((x_1 \to (x_2 \to x_3)) \to ((x_1 \to x_2) \to (x_1 \to x_3))) \tag{3.1}$$

Дерево разбора случаев для этой формулы приведено на рисунке 3.1.

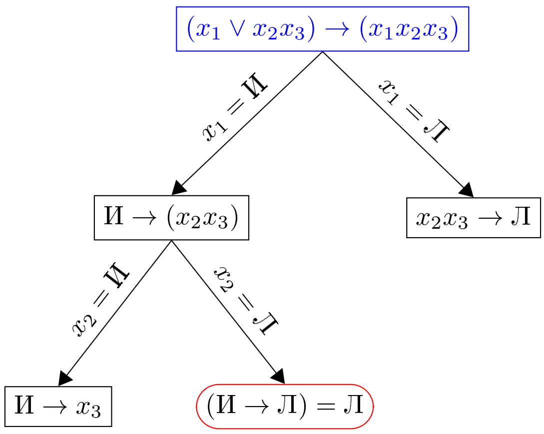


Рис. 3.2: Алгоритм Квайна (ф-ла не общезначима).

Пример 2. В качестве примера применения алгоритма Квайна убедимся в не общезначимости формулы

$$(x_1 \vee x_2x_3) \to (x_1x_2x_3) \tag{3.2}$$

Дерево разбора случаев для этой формулы приведено на рисунке 3.2. Заметим, что дерево построено только частично, т.к. получено выражение $x = \text{Л}$

в одной из концевых вершин.

Модифицированный алгоритм Квайна

Определение 1. *Если $x_1, \ldots, x_m$ — различные переменные, $m \geq 1$, то формулу вида $(x_1^{\sigma_1} \vee \ldots \vee x_m^{\sigma_m})$ назовем* дизъюнктом. *Логическую константу Л по определению так же считаем* дизъюнктом.

Определение 2. *Если $A_1, \ldots, A_n$ — различные дизъюнкты ($n \geq 1$), то формула $(A_1 \wedge \ldots \wedge A_n)$ называется* конъюнктивной нормальной формой(КНФ).

Произвольную формулу логики высказываний можно преобразовать к виду конъюнктивной нормальной формы, используя следующие эквивалентные преобразования:

а) логические связки $\rightarrow$, $\leftrightarrow$ устраняются при помощи тождеств

$$(a \rightarrow b) = (\neg a \vee b); \ (a \leftrightarrow b) = (a \wedge b \vee \neg a \wedge \neg b);$$

б) отрицания в формуле "опускаются до переменных"при помощи тождеств

$$\neg\neg a = a; \ \neg(a \vee b) = \neg a \wedge \neg b; \ \neg(a \wedge b) = \neg a \vee \neg b;$$

в) применяются преобразования дистрибутивности:

$$(a \vee (b \wedge c)) = (a \vee b) \wedge (a \vee c);$$

г) устраняются повторные вхождения переменных в дизъюнктивных подформулах, а также константы И, Л (если сама формула не есть И или Л):

$$a \vee a = a; \ a \vee \neg a = \text{И}; \ a \vee \text{И} = \text{И}; \ a \wedge \text{И} = a; \ a \wedge \text{Л} = \text{Л};$$

д) устраняются одинаковые дизъюнкты: $a \wedge a = a$.

Опишем модифицированный алгоритм Квайна, предназначенный для установления невыполнимости формулы логики высказываний g, преобразованной к виду конъюнктивной нормальной формы.

Для доказательства общезначимости некоторой формулы F ее отрицание $\neg F$ преобразуется к виду КНФ. Это удобно делать применяя формулы де Моргана к отрицанию ДНФ исходной формулы.

Вершинам семантического дерева в этом случае будут сопоставляться не формулы, а их множества дизъюнктов.

Первоначально вводим корень семантического дерева и сопоставляем ему множество дизъюнктов формулы g.

Пусть уже построена часть семантического дерева.

Если концевой вершине дерева сопоставлено пустое множество дизъюнктов (пустое множество соответствует логической константе И), то формула g — выполнима, и алгоритм прекращает работу.

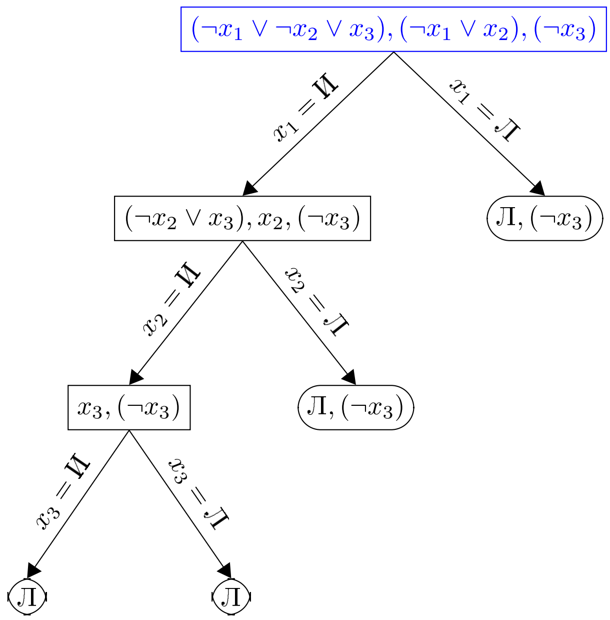


Рис. 3.3: Модифицированный алгоритм Квайна.

Если каждой концевой вершине семантического дерева соответствует множество дизъюнктов, содержащее дизъюнкт Л, то формула g — невыполнима, и алгоритм прекращает работу.

Если некоторой концевой вершине v сопоставлено непустое множество дизъюнктов S, не содержащее дизъюнкта Л, то в S входят дизъюнкты с переменными. Выбираем одну из таких переменных x и разбиваем S на три подкласса: S_1 — все дизъюнкты, в которые x входит без внешнего отрицания, S_2 — все дизъюнкты, в которые x входит с отрицанием, S_3 — все дизъюнкты в которых x не встречается. Далее рассматриваем множество дизъюнктов $S_x = \{d | d \vee \neg x \in S_2\}$ (если $\neg x \in S_2$, то здесь берется $d = \text{Л}$) и множество дизъюнктов и множество дизъюнктов $S_{\neg x} = \{d | d \vee x \in S_1\}$ (если $x \in S_1$, то здесь берется $d = \text{Л}$). Нетрудно видеть, что невыполнимость конъюнкции дизъюнктов списка S эквивалента одновременной невыполнимости конъюнкций дизъюнктов списков $S_x \cup S_3$ и $S_{\neg x} \cup S_3$. Поэтому для продолжения построения семантического дерева вводим две новых вершины v_1 и v_2, которым сопоставляем, соответственно, множества дизъюнктов $S_x \cup S_3$ и $S_{\neg x} \cup S_3$, причем проводим к v_1 и v_2 ребра от v, отмеченные соответственно выражениями $x = \text{И}$ и $x = \text{Л}$.

Далее повторяется описанный выше процесс рассмотрения концевых вершин семантического дерева.

Пример. В качестве примера применения модифицированного алгоритма Квайна обратимся опять к формуле f, заданной выражением (3.1). После устранения логических связок $\rightarrow$ мы получим формулу

$$g = (\neg(\neg x_1 \vee \neg x_2 \vee x_3)) \vee (\neg(\neg x_1 \vee x_2)) \vee \neg x_1 \vee x_3.$$

Далее переходим к рассмотрению формулы $\neg g$, которая имеет вид:

$$\neg g = (\neg x_1 \vee \neg x_2 \vee x_3) \wedge (\neg x_1 \vee x_2) \wedge x_1 \wedge (\neg x_3). \tag{3.3}$$

Семантическое дерево для формулы $\neg g$ приведено на рисунке 3.3.

Правило резолюции

Другой алгоритм распознавания невыполнимости конъюнктивной нормальной формы связан с рассмотрением специальной дедуктивной системы, использующей в качестве правила вывода так называемое *правило резолюции*. В этом случае конъюнктивная нормальная форма снова представляется как множество своих дизъюнктов, эти дизъюнкты и образуют перечень аксиом данной дедуктивной системы. *Правило резолюции*, будучи применено к двум дизъюнктам $A \vee B$ и $\neg A \vee C$ создает в качестве следствия новый дизъюнкт $B \vee C$ (здесь возможны вырожденные случаи отсутствия B либо C, если они оба отсутствуют, то результатом служит дизъюнкт Л). Достаточность применения данного правила вывода для распознавания невыполнимости гарантируется следующим утверждением ([12]).

Теорема 7. *Конъюнкция конечного семейства дизъюнктов S невыполнима тогда и только тогда, когда за конечное применений правила резолюции из S выводится дизъюнкт Л.*

Пример. В качестве примера применения правила резолюции обратимся опять к формуле $\neg g$, заданной выражением (3.3). Исходное множество дизъюнктов имеет вид $\{(\neg x_1 \vee \neg x_2 \vee x_3), (\neg x_1 \vee x_2), x_1, (\neg x_3)\}$. Тогда

- $\{(\neg x_1 \vee x_2),\, x_1\} \vdash x_2$;
- $\{(\neg x_1 \vee \neg x_2 \vee x_3),\, x_1\} \vdash (\neg x_2 \vee x_3)$;
- $\{(\neg x_2 \vee x_3),\, x_2\} \vdash x_3$
- $\{x_3,\, \neg x_3\} \vdash$ Л.

Следовательно, формула $\neg g$ невыполнима.

Упражнения.

Задача 3.2.1.

Проверить общезначимость с использованием алгоритма Квайна:

a) $(x_1 \to x_2) \to ((x_1 \wedge x_3) \to (x_2 \wedge x_3))$;
b) $(x_1 \to x_2) \to ((x_1 \vee x_3) \to (x_2 \vee x_3))$;
c) $(x_1 \to x_2) \to ((x_2 \to x_3) \to (x_1 \to x_3))$;
d) $(x_1x_2 \vee x_1x_3 \vee x_2x_3) \to (x_1x_2 \oplus x_1x_3 \oplus x_2x_3)$.

Задача 3.2.2. Доказать невыполнимость множества дизъюнктов с помощью модифицированного алгоритма Квайна:

a) $\{(\neg x_1), x_2, (x_1 \vee \neg x_2 \vee \neg x_3), (\neg x_2 \vee x_3)\}$;
b) $\{(x_1 \vee x_2 \vee x_3), (\neg x_1 \vee x_2 \vee x_3), (x_1 \vee \neg x_2 \vee x_3), (\neg x_1 \vee \neg x_2 \vee x_3), (\neg x_3)\}$;
c) $\{x_1 \vee x_3, x_1 \vee \neg x_2 \vee \neg x_3 \vee x_4, \neg x_1, x_1 \vee \neg x_3\}$;

Задача 3.2.3. Проверить общезначимость с использованием алгоритма Квайна:

a) $(((\neg x_2 \to \neg x_3) \oplus (\neg x_2 \vee x_2)) \oplus \neg x_3) \to x_3$;
b) $((((\neg x_2 \to \neg x_3) \oplus (\neg x_2 \vee x_2)) \oplus \neg x_3) \to x_3) \vee (\neg x_3)$;
c) $x_1 \vee ((\neg x_4 \vee \neg x_1) \vee (x_1 \oplus (x_1 \oplus \neg x_2)))$;
d) $\neg x_1 \oplus (((\neg x_1 \oplus (\neg x_2 \wedge x_3)) \wedge \neg x_4) \oplus x_4)$;
e) $(\neg x_1 \oplus (((\neg x_1 \oplus (\neg x_2 \wedge x_3)) \wedge \neg x_4) \oplus x_4)) \vee ((x_2 \wedge \neg x_4) \vee (x_4 \wedge \neg x_1) \vee$ $\vee (\neg x_3 \wedge \neg x_4))$;
f) $\neg x_3 \to (((\neg x_4 \oplus \neg x_2) \to x_1) \vee (\neg x_3 \wedge \neg x_1))$;
g) $((x_4 \wedge x_3) \oplus ((x_4 \to \neg x_3) \oplus \neg x_2)) \wedge x_1$;
h) $(((x_4 \wedge x_3) \oplus ((x_4 \to \neg x_3) \oplus \neg x_2)) \wedge x_1) \vee (\neg x_1 \vee \neg x_2)$;
i) $(x_1 \oplus (\neg x_2 \vee (((x_1 \oplus \neg x_3) \to x_1) \vee \neg x_1))) \oplus x_3$;
j) $((x_1 \oplus (\neg x_2 \vee (((x_1 \oplus \neg x_3) \to x_1) \vee \neg x_1))) \oplus x_3) \vee ((x_1 \wedge \neg x_3) \vee (x_3 \wedge \neg x_1))$;

Задача 3.2.4. Привести к КНФ отрицание формулы и применить модифицированный алгоритм Квайна:

a) $((x_1 \to x_2) \vee (x_2 \to \neg x_1))$;
b) $(x_1 \wedge x_3) \vee (x_1 \wedge \neg x_3) \vee (x_2 \wedge x_3) \vee (\neg x_1 \wedge x_2 \wedge x_3)$;
c) $((x_1 \to x_2) \to x_1) \to x_2$;
d) $(\neg x_1 \to x_2) \to (\neg x_2 \to x_1)$.
e) $(((x_3 \wedge (\neg x_2 \vee \neg x_1)) \vee (x_1 \to x_2)) \to (x_2 \wedge x_3)) \vee (\neg x_3)$;

Задача 3.2.5. Доказать невыполнимость семейства дизъюнктов, используя правило резолюции:

a) $\{x_2 \vee x_4, x_1 \vee x_4 \vee \neg x_2, x_2 \vee x_3 \vee \neg x_1, x_3 \vee \neg x_4, x_3 \vee \neg x_1 \vee \neg x_2 \vee \neg x_4, x_1 \vee$ $\vee x_2 \vee \neg x_4, x_3 \vee x_4, \neg x_3\}$;
b) $\{x_2 \vee x_3 \vee \neg x_1, x_1 \vee x_2 \vee x_3, x_3 \vee \neg x_2, x_1 \vee x_2 \vee \neg x_3, x_1 \vee x_3 \vee \neg x_2, \neg x_1 \vee$ $\vee \neg x_3, x_2 \vee \neg x_1 \vee \neg x_3, x_3 \vee \neg x_1 \vee \neg x_2, x_1 \vee x_3, \neg x_2\}$.
c) $\{x_2 \vee x_3 \vee \neg x_1, x_1 \vee x_2, x_1 \vee x_2 \vee x_3, x_2 \vee \neg x_1 \vee \neg x_3, x_1 \vee \neg x_2 \vee \neg x_3, x_1 \vee x_3 \vee$ $\vee \neg x_2, \neg x_1 \vee \neg x_2 \vee \neg x_3, x_2 \vee \neg x_3, x_1 \vee x_3, x_1 \vee x_2 \vee \neg x_3, x_2 \vee x_3, x_3 \vee \neg x_2\}$
d) $\{\neg x_1 \vee \neg x_3, x_1 \vee x_3, x_2 \vee x_3 \vee \neg x_1, x_2 \vee \neg x_1, x_1 \vee x_2 \vee \neg x_3, x_1 \vee x_3 \vee$ $\vee \neg x_2, \neg x_2 \vee \neg x_3, x_2 \vee \neg x_3, \neg x_1 \vee \neg x_2 \vee \neg x_3, x_1 \vee x_2 \vee x_3, x_2 \vee x_3, \neg x_1 \vee \neg x_2\}$.
e) $\{x_1 \vee x_2 \vee x_3 \vee x_4, x_1 \vee x_2 \vee \neg x_3 \vee \neg x_4, \neg x_3 \vee \neg x_4, x_1 \vee \neg x_2, \neg x_1 \vee \neg x_3 \vee$ $\vee \neg x_4, x_1 \vee x_2 \vee \neg x_4, x_2 \vee \neg x_1 \vee \neg x_3 \vee \neg x_4, x_2 \vee x_4 \vee \neg x_3, \neg x_2 \vee \neg x_4, x_3 \vee$ $\vee \neg x_2, x_2 \vee \neg x_3, x_1 \vee \neg x_2 \vee \neg x_3 \vee \neg x_4, x_1 \vee \neg x_2 \vee \neg x_3, x_2 \vee \neg x_4, x_3 \vee \neg x_1 \vee$ $\vee \neg x_2 \vee \neg x_4, \neg x_1 \vee \neg x_4, \neg x_2 \vee \neg x_3, x_3 \vee \neg x_1\}$.

Задача 3.2.6. Привести к КНФ отрицание формулы и применить правило резолюции:

a) $(x_1 \leftrightarrow x_2) \to (x_1 \to x_2)$;
b) $(x_1 \to x_2) \to (\neg x_2 \to \neg x_1)$;
c) $(((\neg x_2 \to x_3) \vee \neg x_1) \oplus ((\neg x_3 \wedge \neg x_3) \wedge (\neg x_3 \to (x_2 \vee x_1)))) \vee (x_2 \wedge \neg x_3)$;
d) $(\neg x_2 \vee ((\neg x_3 \vee \neg x_2) \to x_2)) \vee ((x_1 \vee (\neg x_3 \wedge x_3)) \wedge x_1)$;

Задача 3.2.7. Привести пример доказательства, основанного на общезначимости формулы a) $(\neg p_2 \to \neg p_1) \to (p_1 \to p_2)$; b) $((p_1 \to p_2) \wedge (p_2 \to p_3) \wedge$ $\wedge (p_3 \to p_1)) \to ((p_1 \leftrightarrow p_2) \wedge (p_2 \leftrightarrow p_3) \wedge (p_3 \leftrightarrow p_1))$.

Задача 3.2.8. а) Формула $\mathcal{F}$ исчисления высказываний не содержит операций кроме $\leftrightarrow$. Доказать, что $\mathcal{F}$ общезначима тогда и только тогда, когда каждая переменная входит в $\mathcal{F}$ четное число раз. б*) Формула исчисления высказываний $\mathcal{G}$ не содержит операций кроме $\leftrightarrow$ и $\neg$. Доказать, что $\mathcal{G}$ общезначима тогда и только тогда, когда каждая переменная и знак отрицания входит в $\mathcal{G}$ четное число раз.

Задача 3.2.9. Записать в виде формулы исчисления высказываний и проверить общезначимость.

a) *Если Джонс не встречал этой ночью Смита, то либо*[1] *Смит был убийцей, либо Джонс лжет. Если Смит не был убийцей, то Джонс не встречал Смита этой ночью и убийство произошло после* 23^{00}. *Если убийство было после* 23^{00}, *то либо Смит был убийцей, либо Джонс говорит правду. Следовательно, Смит — убийца.*

b) *Если капиталовложения останутся постоянными, то возрастут правительственные расходы или*[2] *возникнет безработица. Если правительственные расходы не возрастут, то налоги будут снижены. Если налоги будут снижены и капиталовложения останутся постоянными, то безработица не возникнет. Следовательно, правительственные расходы возрастут.*

Задача 3.2.10. Проверить совместность каждого из множеств утверждений (т.е. то, что их конъюнкция выполнима):

a) *Либо свидетель не был запуган, либо, если Генри покончил жизнь самоубийством, то записка была найдена. Если свидетель был запуган, то Генри не покончил жизнь самоубийством. Если записка была найдена, то Генри покончил жизнь самоубийством.*

b) *Если вечер скучен, то Алиса начинает плакать, или Анатоль рассказывает смешные истории. Если Сильвестр приходит на вечер, то вечер скучен или Алиса начинает плакать. Если Анатоль рассказывает смешные истории, то Алиса не начинает плакать. Сильвестр приходит на вечер тогда и только тогда, когда Анатоль не рассказывает смешные истории. Если Алиса начинает плакать, то Анатоль рассказывает смешные истории.*

c) *Если курс ценных бумаг растет или процентная ставка снижается, то либо падает курс акций, либо налоги не повышаются. Курс акций понижается тогда и только тогда, когда растет курс ценных бумаг и растут налоги. Если процентная ставка снижается, то либо курс акций не понижается, либо курс ценных бумаг не растет. Либо повышаются налоги, либо курс акций понижается и снижается процентная ставка.*

Задача 3.2.11. Формула исчисления высказываний не содержит значков логических операций кроме $\vee$ и $\wedge$. Докажите, что она выполнима, но не общезначима.

Задача 3.2.12. Пусть C — формула, в которой выделено некоторое вхождение подформулы A, а C' — формула, полученная из C заменой этого вхождения A на B. Докажите, что если $A \leftrightarrow B$ общезначима, то и $C \leftrightarrow C'$ — тоже общезначима.

---

[1] исключающее или

[2] дизъюнкция

Задача 3.2.13. Пусть формула A, содержит только знаки $\vee$, $\wedge$ и $\neg$ и переменные x_1, ... x_n, а формула A^* получена из A заменой x_i на $\neg x_i$, $\vee$ на $\wedge$ и $\wedge$ на $\vee$. Докажите, что $\neg A \leftrightarrow A^*$ общезначима.

Задача 3.2.14.a) Сколько различных выражений для множеств A и B можно составить с помощью операций $\cap$, $\cup$, $\setminus$? Два выражения считаются одинаковыми, если они тождественно равны при любых A и B.

b) Тот же вопрос для трех множеств A, B и C;

c) ... для n множеств $A_1, \dots, A_n$.

Задача 3.2.15.a) Сколько различных выражений для множеств A и B можно составить из переменных и с помощью операций $\cap$, $\cup$?

b) Тот же вопрос для трех множеств A, B и C.

Ответы и указания.

3.2.1 a) Формула общезначима (см. рис. 3.4). b) Формула общезначима (см.

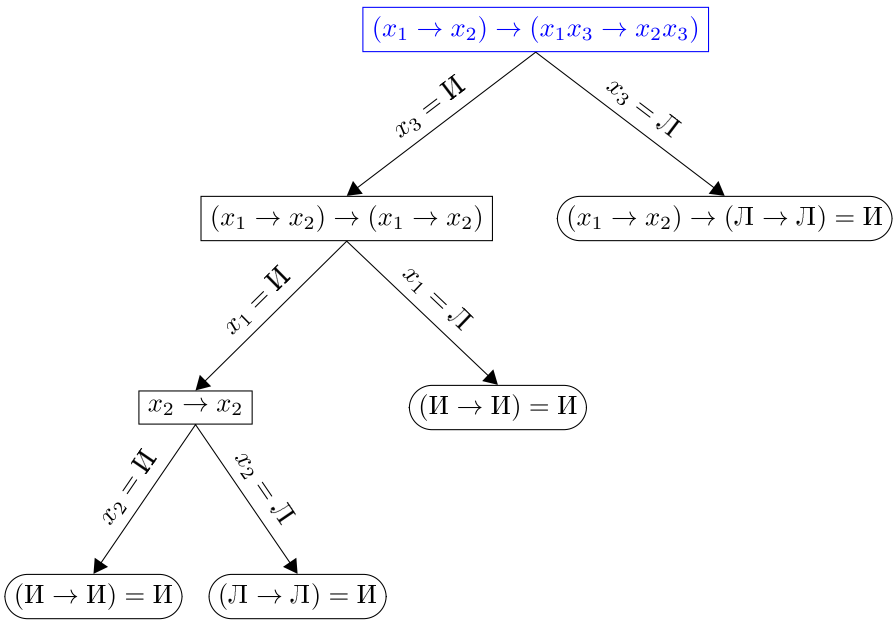


Рис. 3.4: К задаче 3.2.1.a.

рис. 3.5). c) Формула не общезначима (см. рис. 3.6). d) Формула общезначима. (см. рис. 3.7).

3.2.2 a) См. рис 3.8 b) См. рис 3.9 c) См. рис 3.10

3.2.3a) Не общезначима.

b) Общезначима.

c) Общезначима.

d) Не общезначима.

e) Общезначима.

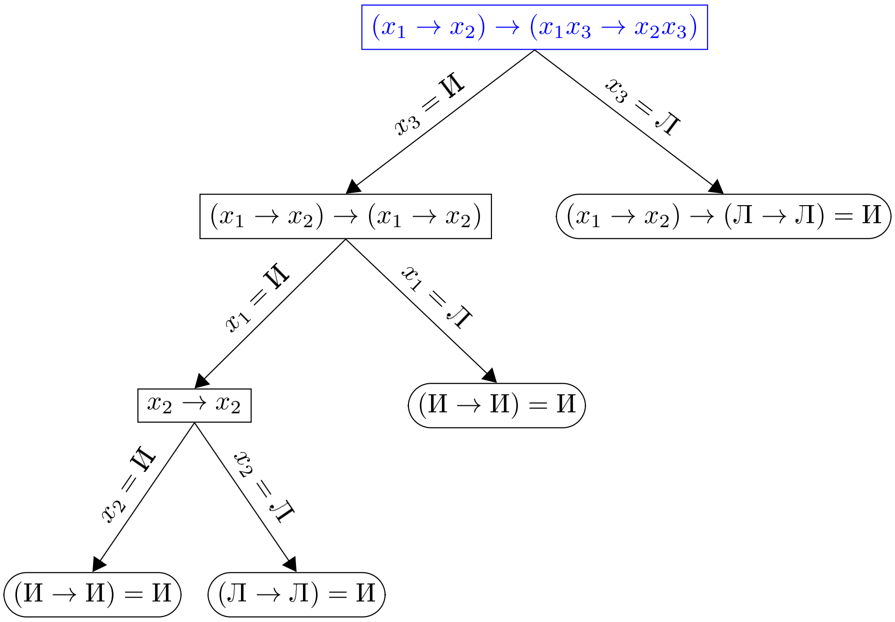


Рис. 3.5: К задаче 3.2.1.b.

f) Общезначима.
g) Не общезначима.
h) Общезначима.
i) Не общезначима.
j) Общезначима.

3.2.4 a) $\overline{((x_1 \to x_2) \vee (x_2 \to \neg x_1))} = \overline{(\neg x_1 \vee x_2)} \wedge \overline{(\neg x_1 \vee \neg x_2)} = x_1 \wedge \wedge (\neg x_2) \wedge x_2$; формула невыполнима, следовательно, исходная общезначима. b) $\overline{(x_1 \wedge x_3) \vee (x_1 \wedge \neg x_3) \vee (x_2 \wedge x_3) \vee (\neg x_1 \wedge x_2 \wedge x_3)} = (\neg x_1 \vee \neg x_3) \wedge (\neg x_1 \vee \vee x_3) \wedge (\neg x_2 \vee \neg x_3) \wedge (x_1 \vee \neg x_2 \vee \neg x_3)$; Далее, применяя модифицированный алгоритм Квайна (см. рис 3.11) видим, что исходная формула не общезначима. c) Приведем к ДНФ исходную формулу $\bar{x}_1 \vee x_2 \vee \bar{x}_1 x_2)$. Взяв отрицание, получим КНФ $x_1 \& \bar{x}_2 \& (x_1 \vee \bar{x}_2)$. Применяя модифицированный алгоритм Квайна, получаем, что отрицание выполнимо, следовательно, формула не общезначима. d) Общезначима, решается аналогично пункту «a». e) Не общезначима, решается аналогично пункту «b».

3.2.5 a) Обозначим $D_1 = x_2 \vee x_4$, $D_2 = x_1 \vee x_4 \vee \neg x_2$, $D_3 = x_2 \vee x_3 \vee \neg x_1$, $D_4 = x_3 \vee \neg x_4$, $D_5 = x_3 \vee \neg x_1 \vee \neg x_2 \vee \neg x_4$, $D_6 = x_1 \vee x_2 \vee \neg x_4$, $D_7 = x_3 \vee x_4$, $D_8 = \neg x_3$. Тогда $D_7, D_8 \vdash x_4$, $D_4, x_4 \vdash x_3$, $D_8, x_3 \vdash \text{Л}$. Остальные пункты делаются аналогично.

3.2.6 a) Заметим, что удобнее сначала привести формулу к ДНФ, а потом взять ее отрицание.

a) $F = (x_1 \leftrightarrow x_2) \to (x_1 \to x_2) = x_1\bar{x}_2 \vee \bar{x}_1 x_2 \vee x_1 \vee x_2$; КНФ отрицания: $\bar{F} = (\bar{x}_1 \vee x_2) \& (x_1 \vee \bar{x}_2) \& x_1 \& \bar{x}_2$. Применим к списку дизъюнктов правило резолюции:

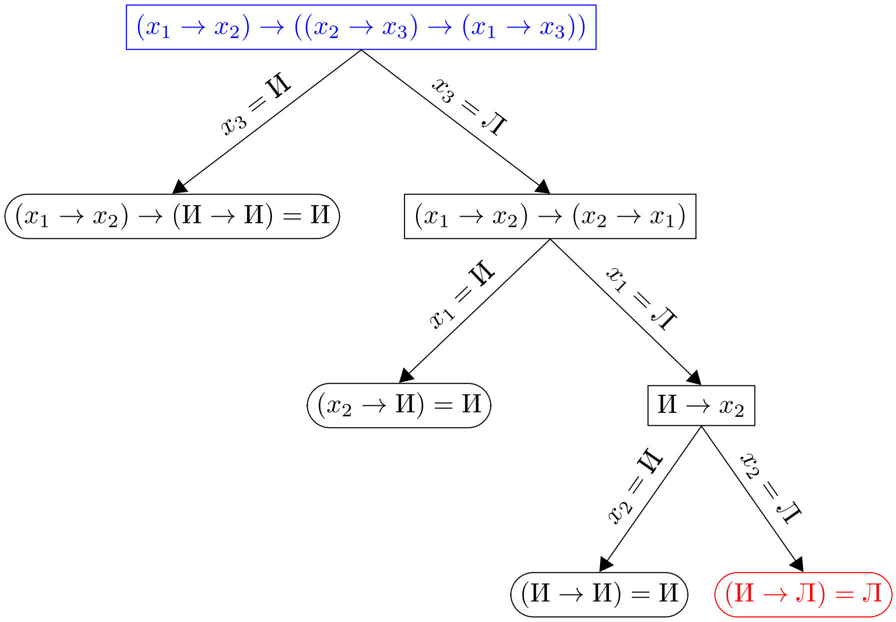


Рис. 3.6: К задаче 3.2.1.c.

- $(\bar{x}_1 \vee x_2), x_1 \vdash x_2$;
- $x_2, \bar{x}_2 \vdash Л$.

Получена константа Л, следовательно, исходная формула общезначима.

Остальные пункты решаются аналогично.

3.2.7 a) Любое доказательство от противного. b) Например (в курсе линейной алгебры) доказательство того, что равносильны утверждения: i) Матрица A обратима; ii) Ее строки (столбцы) линейно независимы; iii) A представима как произведение элементарных матриц.

3.2.8 a) Заменить $F \leftrightarrow G$ на $F \oplus G \oplus 1$. Получится линейная функция, которая тождественно истина в случае, если все переменные сокращаются. b) Заменить $F \leftrightarrow G$ на $F \oplus G \oplus 1$ и $\neg F$ на $F \oplus 1$.

3.2.9 a) Введем обозначения: $A \Leftrightarrow$ (Джонс встречал этой ночью Смита); $B \Leftrightarrow$ (Смит — убийца); $C \Leftrightarrow$ (Джонс лжет); $D \Leftrightarrow$ (убийство произошло после 23^{00}). Тогда утверждение запишется в виде: $\big(\neg A \to (B \oplus C)\big) \wedge \big(\neg B \to (\neg A \wedge D)\big) \wedge \big(D \to (B \oplus \neg C)\big) \to B$. Несложно проверить, что эта формула общезначима (например, с помощью алгоритма Квайна). b) Введем обозначения: $K \Leftrightarrow$ (капиталовложения постоянны); $R \Leftrightarrow$ (правительственные расходы возрастут); $B \Leftrightarrow$ (возникнет безработица); $N \Leftrightarrow$ (налоги будут снижены). Тогда утверждение запишется в виде $((K \to (R \vee B))\&(\neg R \to N)\&(N\&K \to \neg B)) \to R$. Несложно проверить, что она не общезначима.

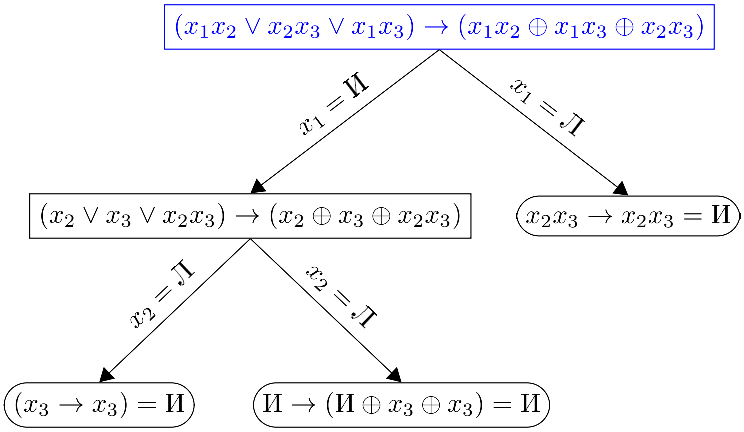


Рис. 3.7: К задаче 3.2.1.d.

3.2.10 a) Введем обозначения:

$$A \Leftrightarrow (\text{свидетель был запуган});$$

$$B \Leftrightarrow (\text{Генри покончил жизнь самоубийством});$$

$$C \Leftrightarrow (\text{записка была найдена}).$$

Рассмотрим конъюнкцию $(\neg A \oplus (B \to C))\&(A \to \neg B)\&(C \to B)$. Она выполнима, например при $A = \text{И}$, $B = C = \text{Л}$.
b) Введем обозначения:

$$A \Leftrightarrow (\text{вечер скучен});$$

$$B \Leftrightarrow (\text{Алиса начинает плакать});$$

$$C \Leftrightarrow (\text{Анатоль рассказывает смешные истории});$$

$$D \Leftrightarrow (\text{приходит Сильвестр}).$$

Рассмотрим конъюнкцию

$$\big(A \to (B \vee C)\big)\&\big(D \to (A \vee B)\big)\&\big(C \to \neg B\big)\&\big(D \leftrightarrow \neg C\big)\&\big(B \to C\big).$$

Она выполнима, например, при $A = C = \text{И}$, $B = D = \text{Л}$.
c) Введем обозначения:

$$A \Leftrightarrow (\text{курс ценных бумаг растет});$$

$$B \Leftrightarrow (\text{процентная ставка снижается});$$

$$C \Leftrightarrow (\text{падает курс акций});$$

$$D \Leftrightarrow (\text{растут налоги}).$$

Рассмотрим конъюнкцию:

$$\big((A \vee B) \to (C \oplus \neg D)\big)\&\big(C \leftrightarrow (A\&D)\big)\&\big(B \to (\neg C \oplus \neg A)\big)\&\big(D \oplus (C\&B)\big).$$

Она выполнима, например, при $A = C = D = \text{И}$, $B = \text{Л}$.

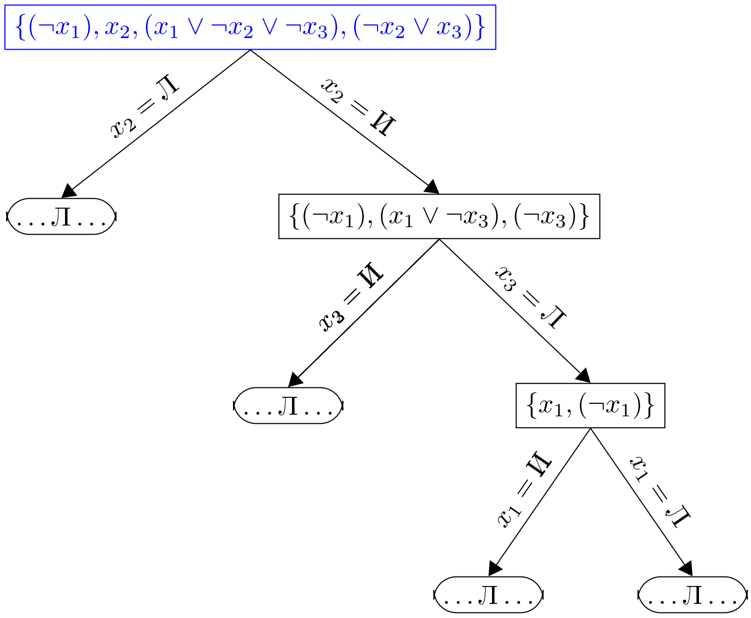


Рис. 3.8: К задаче 3.2.2.a.

3.2.11 При подстановке И вместо каждой переменной получаем И, а при подстановке Л — Л.

3.2.12 Зафиксировать предметные переменные и рассмотреть функцию $f(x)$, которая истина для тех и только тех x, для которых при подстановке $A = x$ в C получается истинная формула.

3.2.13 Доказывается индукцией по глубине формулы и применением правил де Моргана.

3.2.14 a) 15; b) 255; c) $2^{2^n} - 1$. Каждое такое выражение соответствует булевской функциии (кроме $f(x) \equiv 1$).

3.2.15 a) 4; b)18. Каждое такое выражение соответствует монотонной булевской функции (кроме констант 0 и 1).

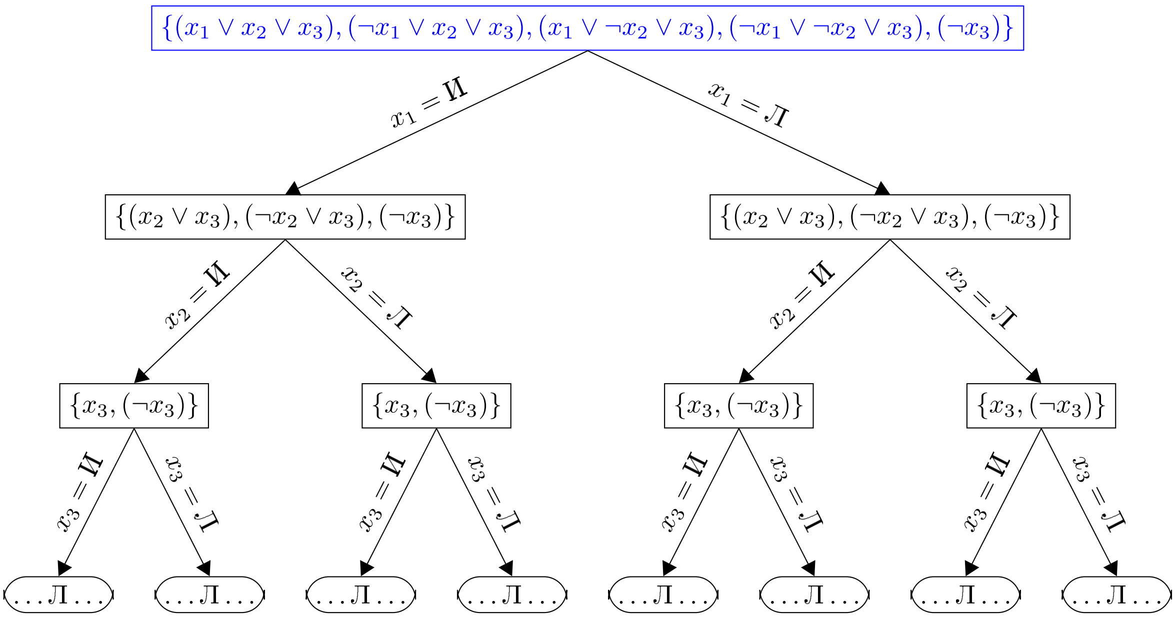


Рис. 3.9: К задаче 3.2.2.b.

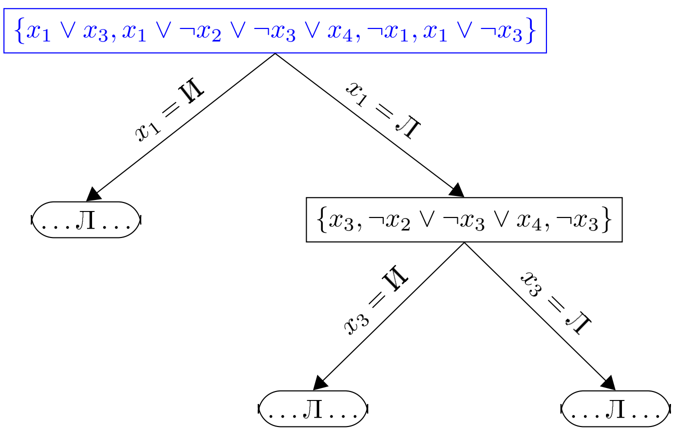


Рис. 3.10: К задаче 3.2.2.c.

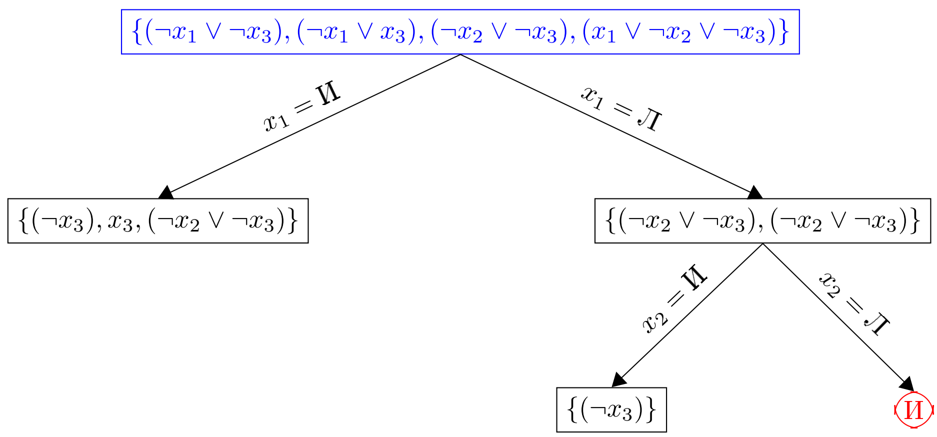


Рис. 3.11: К задаче 3.2.4.b.

3.3. Исчисление предикатов

Обозначения и определения

В случае языка логики предикатов наш алфавит будет состоять из следующих групп символов:

1) счетный список предметных переменных $x_1, x_2, \ldots$;
2) счетный список предметных констант $a_1, a_2, \ldots$;
3) счетный список функциональных символов $f_1, f_2, \ldots$;
4) счетный список предикатных символов $P_1, P_2, \ldots$;
5) логические константы И, Л;
6) логические связки $\neg, \vee, \wedge, \rightarrow, \leftrightarrow$;
7) кванторы $\forall$ и $\exists$;
8) скобки "(")"и запятая ".

Каждый функциональный либо предикатный символ характеризуется своей *арностью* — некоторым натуральным числом. При этом предполагается, что для каждого натурального n имеется бесконечно много как предикатных, так и функциональных символов арности n.

Правильные выражения языка логики предикатов разбиваются на два непересекающихся множества — множество термов и множество формул. Дадим сначала индуктивное определение *множества термов*.

Т1) Однобуквенное слово, состоящее из предметной переменной либо предметной константы, есть терм.

Т2) Если $t_1, \ldots, t_n$ — термы и g — функциональный символ арности n, то слово $g(t_1, \ldots, t_n)$ — есть терм.

Индуктивное определение *формул логики предикатов* опирается на уже введенное понятие терма.

Ф1) Однобуквенное слово состоящее из логической константы И либо Л, есть формула логики предикатов.

Ф2) Если $t_1, \ldots, t_n$ — термы и g — предикатный символ арности n, то слово $g(t_1, \ldots, t_n)$ есть формула логики предикатов.

Ф3) Если f и g — формулы логики предикатов, то слова $(\neg f)$, $(f \vee g)$, $(f \wedge g)$, $(f \rightarrow g)$, $(f \leftrightarrow g)$ суть формулы логики предикатов (такие формулы будем называть *атомарными* или *атомами*).

Ф4) Если f — формула логики предикатов и x — предметная переменная, то слова $(\forall x f)$ и $(\exists x f)$ суть формулы логики предикатов.

Интерпретацией языка логики предикатов называем четверку (M, F_1, F_2, F_3), такую что:

а) M — непустое множество, называемое областью интерпретации;

б) F_1 — функция, сопоставляющая каждой предметной константе некоторый элемент из M;

в) F_2 — функция, сопоставляющая каждому функциональному символу f арности n некоторую n-местную функцию, определенную на M и принимающую значения из M.

г) F_3 — функция, сопоставляющая каждому предикатному символу P арности n некоторой n-местный предикат определенный на M (функцию со значениями И, Л).

Если задана интерпретация $I = (M, F_1, F_2, F_3)$, то каждый терм t языка логики предикатов определяет в этой интерпретации некоторую функцию $(t)_I$, определенную на M и принимающую значения из M, а каждая формула f — предикат $(f)_I$, определенный на M (в вырожденных случаях $(t)_I$ может отождествляться с элементом M, а $(f)_I$ с логической константой).

Здесь используется следующее индуктивное определение:

1) Если x — предметная переменная, то $(x)_I$ есть тождественная функция от переменной x определенная на M.

2) Если a — предметная константа, то $(a)_I$ есть элемент $F_1(a)$ из M.

3) Если для термов $t_1, \ldots, t_n$ уже определены $(t_1)_I, \ldots, (t_n)_I$, f — функциональный символ арности n и $\phi = F_2(f)$ — функция от n переменных, отображающая M^n в M, то $(f(t_1, \ldots, t_n))_I = \phi((t_1)_I, \ldots, (t_n)_I)$

4) $(\text{И})_I = \text{И}$, $(\text{Л})_I = \text{Л}$.

5) Если для термов $t_1, \ldots, t_n$ уже определены $(t_1)_I, \ldots, (t_n)_I$, P — предикатный символ арности n и $\pi = F_3(P)$ — предикат от n переменных, отображающий M^n в $\{\text{И}, \text{Л}\}$, то $(P(t_1, \ldots, t_n))_I = \pi((t_1)_I, \ldots, (t_n)_I)$

6) Если для f и g уже определены $(f)_I$ и $(g)_I$, то $(\neg f)_I$, $(f \vee g)_I$, $(f \wedge g)_I$, $(f \to g)_I$, $(f \leftrightarrow g)_I$ получаются применением истинностных функций для $\neg, \vee, \wedge, \to, \leftrightarrow$ к $(f)_I$ и $(g)_I$.

7) Если уже определен предикат $(f)_I = \phi(x_1, \ldots, x_n)$, то $(\forall x_i f)_I = \psi(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$ — предикат, принимающий значение И на таких наборах $\alpha_1, \ldots, \alpha_{i-1}, \alpha_{i+1}, \ldots, \alpha_n$ элементов M, что для каждого α_i из M значение $\phi(\alpha_1, \ldots, \alpha_n)$ есть И. Аналогично, $(\exists x_i f)_I = \xi(x_1, \ldots, x_{i-1}, x_{i+1}, \ldots, x_n)$ — предикат,принимающий значение И на таких наборах $\alpha_1, \ldots, \alpha_{i-1}, \alpha_{i+1}, \ldots, \alpha_n$ элементов M, что существует α_i из M, что значение $\phi(\alpha_1, \ldots, \alpha_n)$ есть И.

Вхождение переменной x в формулу f логики предикатов, расположенное внутри подформул вида $(\forall x g)$ либо $(\exists x g)$, называется *связанным*, вхождения переменных не являющиеся связанными, называются *свободными*. Переменная имеющая хотя бы одно свободное вхождение в формулу либо терм f, называется *свободной переменной* f. Как нетрудно видеть, функция либо предикат $(f)_I$, определенная выше, зависит лишь от свободных переменных f. Формула без свободных переменных называется *замкнутой*.

В формуле $\exists x_1((\forall x_2 P_1(x_2, x_1)) \vee P_2(x_2, x_1))$ первое вхождение переменной x_2 связанное, а второе — свободное, оба вхождения переменной x_1 связанные. Тем самым, у этой формулы одна свободная переменная — x_2. Формула $((\exists x_1 P_1(x_1)) \to (\forall x_2 P_2(x_2)))$ — пример замкнутой формулы.

Терм t называется *свободным для переменной* x_i в формуле логики предикатов f, если никакое свободное вхождение x_i в f не лежит в области действия никакого квантора $Q x_j$, где $Q \in \{\forall, \exists\}$, x_j — переменная, входящая в терм t.

Например, терм $f_1(x_1, x_3)$ свободен для x_1 в формуле $((\forall x_2 P_1(x_1, x_2)) \to P_2(x_1))$, но не свободен для x_1 в формуле $(\exists x_3(\forall x_2(P_1(x_1, x_2) \to P_2(x_1))))$.

Если f — есть формула логики предикатов, x — свободная предметная переменная формулы f и t — терм, то через $S_t^x f$ обозначим результат подстановки в f терма t вместо всех свободных вхождений переменной x.

Лемма 5 (об интерпретациях)**.** *Если f — есть формула логики предикатов, x — свободная предметная переменная формулы f и t — терм, сво-*

бодный для переменной x в f, то для любой интерпретации I формула $(S_t^x f)_I$ определяет предикат, получающийся подстановкой в $(f)_I$ функции $(t)_I$ вместо переменной x.

Доказательство этой леммы может получено индукцией по построению формулы f и рекомендуется в качестве самостоятельного упражнения.

Если формула f определяет в интерпретации I предикат, тождественно равный И, то она называется *истинной в* I, а I в этом случае называется *моделью для* f.

Формулу логики предикатов f называем *общезначимой*, если она истинна во всех интерпретациях, и *выполнимой*, если хотя бы в одной интерпретации она определяет не тождественно ложный предикат.

Если формула $f_1 \wedge \ldots \wedge f_n \to f_0$ общезначима, то f_0 называется *логическим следствием* формул $f_1, \ldots, f_n$, если формула $f_1 \leftrightarrow f_2$ общезначима, то формулы f_1 и f_2 называются *эквивалентными*.

Формула $((\neg P_1(x_1)) \vee P_1(x_1))$ есть пример общезначимой формулы. Формула $((\neg P_1(x_1)) \vee P_1(f_1(x_1)))$ — выполнима, но не общезначима.

Далее, как и в случае логики высказываний, договоримся опускать скобки, подразумевая, что $\neg$ — самая приоритетная операция, и в случае, когда порядок расставления скобок не существенен (например, для ассоциативных операций). Кроме того, договоримся опускать скобки в формулах вида $(Q_1(Q_2 A))$, где Q_1, Q_2 — любые кванторы. Например, вместо $(\forall x_1(\exists x_2(\forall x_3 P_1(x_1, x_2, x_3))))$ будем писать $\forall x_1 \exists x_2 \forall x_3 P_1(x_1, x_2, x_3)$.

Полнота исчисления предикатов

Для указания дедуктивной системы, перечисляющей все общезначимые формулы логики предикатов, воспользуемся имеющимися у нас схемами аксиом A1)–A3), в которых теперь f, g, h — произвольные формулы логики предикатов, и добавим к ним следующие две схемы аксиом:

A4) Если f, g — формулы логики предикатов и x — предметная переменная, не являющаяся свободной переменной формулы f, то формула $(\forall x(f \to g)) \to (f \to (\forall x g))$ есть аксиома.

A5) Если f — есть формула логики предикатов, x — предметная переменная и t — терм, свободный для переменной x в f, то формула $(\forall x f) \to S_t^x f$ есть аксиома.

Правилами вывода в этой дедуктивной системе являются уже известное нам правило modus ponens, а также новое правило (известное как *правило обобщения*), позволяющее из формулы f выводить формулу $(\forall x f)$. Как и в случае логики высказываний, ограничиваемся только логическими связками $\neg$ и $\to$, причем из кванторов используем только квантор общности. Легко видеть, что квантор существования может быть выражен через квантор общности и отрицание: формулы $\exists_x A(x)$ и $\neg(\forall_x(\neg A(x)))$ эквивалентны.

Так как все аксиомы указанной дедуктивной системы общезначимы, а правила вывода сохраняют общезначимость, то все выводимые в ней формулы общезначимы. Обратно, имеет место следующее утверждение.

Теорема 8 (Геделя о полноте исчисления предикатов)**.** *Каждая общезначимая формула логики предикатов является выводимой в приведенной выше дедуктивной системе.*

Теорема 9 (Эрбрана)**.** *Замкнутая формула f вида $\forall x_1 \dots \forall x_n g$, где g не содержит кванторов, выполнима тогда и только тогда, когда каждое конечное подмножество множества $\{S^{x_1,\dots,x_n}_{t_1,\dots,t_n} g : t_1, \dots, t_n \in H_f\}$ выполнимо.*

Лемма 6 (об унифицирующей подстановке)**.** *Если $f_1, g_1, \dots, f_k, g_k$ — бескванторные формулы либо термы и существует такая подстановка σ, что $\sigma(f_1) = \sigma(g_1), \dots, \sigma(f_k) = \sigma(g_k)$, то существует минимальная подстановка σ_1, обладающая этим свойством, то есть $\sigma_1(f_1) = \sigma_1(g_1), \dots, \sigma_1(f_k) = \sigma_1(g_k)$ и для любой подстановки σ_2, удовлетворяющей условиям $\sigma_2(f_1) = \sigma_2(g_1), \dots, \sigma_2(f_k) = \sigma_2(g_k)$, выполнено: $\sigma_2 = \sigma_3\sigma_1$ при подходящей подстановке σ_3 (здесь умножение подстановок означает их последовательное применение).*

Предполагаемая здесь процедура автоматического доказательства теорем использует при поиске такого вывода константы Л следующие (как легко видеть, корректные) правила вывода:

R0 (*переобозначения переменных*). Из произвольного дизъюнкта D выводится дизъюнкт D', полученный из D переобозначением без отождествлений переменных, входящих в D.

R1 (*правило резолюции*). Если $f_1 \vee \dots \vee f_s$ и $g_1 \vee \dots \vee g_r$ — дизъюнкты, множества переменных которых не пересекаются, причем f_i имеет вид $P(t_1, \dots, t_k)$, а g_j — вид $\neg P(t'_1, \dots, t'_k)$ и σ — унифицирующая подстановка для пары $(P(t_1, \dots, t_k), P(t'_1, \dots, t'_k))$, $i \in \{1, \dots, s\}$, $j \in \{1, \dots, r\}$, то выводится дизъюнкт

$$\sigma(f_1) \vee \dots \vee \sigma(f_{i-1}) \vee \sigma(f_{i+1}) \vee \dots \vee \sigma(f_s) \vee \sigma(g_1) \vee \dots \vee \sigma(g_{j-1}) \vee \sigma(g_{j+1}) \vee \dots \vee \sigma(g_r)$$

(если $s = r = 1$, то выводится константа Л).

Дизъюнкт получаемый согласно правилу R1, называется *резольвентой* исходных дизъюнктов.

R2 (*правило склеивания*). Если $f_1 \vee \dots \vee f_s$ — дизъюнкт и σ — унифицирующая подстановка для (f_i, f_j), $i \neq j$, $i, j \in \{1, \dots, s\}$, то выводится дизъюнкт $\sigma(f_1) \vee \dots \vee \sigma(f_{i-1}) \vee \sigma(f_{i+1}) \vee \dots \vee \sigma(f_s)$.

Теорема 10. *Если формула F_2 невыполнима и ее конъюнктивная нормальная форма имеет вид $\&_{i=1}^m B_i$ то за конечное число шагов из дизъюнктов $B_1, \dots, B_m$ при помощи правил вывода $R0$, $R1$, $R2$ может быть выведена логическая константа Л.*

Приведем пример формализации задачи, описанной на «естественном» языке:

> *Все люди смертны. Сократ — человек. Следовательно, Сократ смертен.*

Обозначим принадлежность к роду людскому предикатом $A(x)$, свойство «быть смертным» — предикатом $B(x)$, а Сократа — константой S. Тогда утверждение запишется в виде: $(\forall x(A(x) \Rightarrow B(x))\&A(S)) \Rightarrow B(S)$. Для доказательства общезначимости докажем невыполнимость отрицания этой формулы: $\neg\,(\forall x(A(x) \Rightarrow B(x))\&A(S)) \Rightarrow B(S)$. Получаем после преобразований к бескванторной форме $(\neg A(x) \vee B(x))\&A(S)\&(\neg B(S))$. Применив правило резолюции с подстановкой $\sigma(x \to S)$ к дизъюнктам $(\neg A(x) \vee B(x))$ и $A(S)$, выводим дизъюнкт $B(S)$. применив правило резолюции повторно, к

$B(S)$ и $\neg B(S)$, выводим константу Л. Таким образом, отрицание исходной формулы невыполнимо, следовательно, она сама общезначима.

Еще один пример:

> *Некоторые кочаны капусты являются паровозами. Некоторые паровозы играют на рояле. Следовательно, некоторые кочаны капусты играют на рояле.*

Обозначим $A(x)$ предикат «x является кочаном капусты», $B(x)$ — «x является паровозом» и $C(x)$ — «x играет на рояле».

Тогда первое утверждение в задаче можно записать как[3] $\exists x(A(x)\&B(x))$, второе — $\exists x(B(x)\&C(x))$. полностью условие запишется в виде:

$$\big(\big(\exists x(A(x)\&B(x))\big)\&\big(\exists x(B(x)\&C(x))\big)\big) \Rightarrow \big(\exists x(A(x)\&C(x))\big).$$

Предваренная форма будет

$$\forall x_1\forall x_2\exists x_3\left((\neg A(x_1)) \vee (\neg B(x_1)) \vee (\neg B(x_2)) \vee (\neg C(x_2)) \vee (A(x_3)\&C(x_3))\right).$$

Сколемизируем (учитывая, что x_3 не зависит от x_1, $_2$)

$$\forall x_1\forall x_2\left((\neg A(x_1)) \vee (\neg B(x_1)) \vee (\neg B(x_2)) \vee (\neg C(x_2)) \vee (A(x_0)\&C(x_0))\right).$$

Несложно подобрать модель, в которой она не выполнена: $x \in E_2$, $A(x) = x$ $B(x) \equiv$ И, $C(x) = \neg x$, при $x_1 =$ Л$, x_2 =$ И (надо заметить, что $(A(x_0)\&C(x_0)) =$ Л при любом x_0).

Упражнения.

Задача 3.3.1. Доказать, что формулы не общезначимы, предъявив предикаты Φ и P для которых они ложны:

a) $\forall x\big(\Phi(x) \vee P(x)\big) \leftrightarrow \big(\forall x\Phi(x)\big) \vee \big(\forall xP(x)\big)$;
b) $\exists x\big(\Phi(x)\&P(x)\big) \leftrightarrow \big(\exists x\Phi(x)\big)\&\big(\exists xP(x)\big)$;
c) $\forall y\exists xP(x,y) \to \exists x\forall yP(x,y)$.

Задача 3.3.2. Какие из приведенных ниже формул общезначимы?

a) $\exists x\big(f(x) \to g(x)\big) \to \big(\forall xf(x) \to \forall xg(x)\big)$;
b) $\forall x\big(f(x) \to g(x)\big) \to \big(\forall xf(x) \to \forall xg(x)\big)$;
c) $\forall x\big(f(x) \to g(x)\big) \to \big(\exists xf(x) \to \exists xg(x)\big)$;
d) $\exists x\big(f(x) \to g(x)\big) \leftrightarrow \big(\forall xf(x) \to \exists xg(x)\big)$;
e) $\forall x\big(f(x) \to g(x)\big) \leftrightarrow \big(\exists xf(x) \to \forall xg(x)\big)$;

Задача 3.3.3. Задана область интерпретации — $\mathbb{N}$, на ней заданы предикаты $A_1^2(x,y) \Leftrightarrow (x \vdots y)$, $A_2^2(x,y) \Leftrightarrow (x < y)$ и константа «1». Записать следующие утверждения в виде формулы ИП:

a) p — простое число.
b) $(a,b) = 1$;
c) $(a,b) = c$;

[3] Можно было более формально понимать условие и записать $\exists x(A(x) \to B(x))$.

d) a является натуральной степенью числа 2 (в решении разрешается использовать константу «2»).

e) a является натуральной степенью некоторого простого числа (т.е. существуют простое p и натуральное k, такие, что $a = p^k$). Разрешается использовать предикат, полученный в пункте «а»: $A_1^1(p) \Leftrightarrow p$ — простое .

Задача 3.3.4. Задана область интерпретации — $\mathbb{R}$, на ней заданы предикаты $P_1^2(x,y) \Leftrightarrow (x = y)$, $P_2^2(x,y) \Leftrightarrow (x < y)$ и $P_1^3(x,y,\varepsilon) \Leftrightarrow (|x-y| < \varepsilon)$. Записать в виде формулы исчисления предикатов следующие утверждения:

a) Функция f стремится к данному A при $x \to 0$;
b) Не существует предела $f(x)$ при $x \to 0$;
c) Функция f непрерывна в точке x_0;
d) Функция f непрерывна на отрезке $[a,b]$.
e) * Если f непрерывна на отрезке $[a,b]$, то она равномерно непрерывна на нем.

Задача 3.3.5. Привести к предваренной нормальной форме:

a) $\exists x A_1^2(x,y) \Rightarrow \big(A_1^1(x) \Rightarrow \nexists\, u A_1^2(x,u)\big)$.
b) $\forall x(A_1^1(x) \Leftrightarrow \exists y \forall z A_1^3(x,y,z))$;
c) $\forall x\Bigg(A_1^1(x) \Rightarrow A_1^2(x,y)\Bigg) \Rightarrow \Bigg((\exists t A_1^1(t)) \Rightarrow (\exists z A_1^2(y,z))\Bigg)$;

Задача 3.3.6. Привести к Сколемовской форме:

a) $\forall x \exists y \forall u \exists v A_1^4(x,y,u,v)$;
b) $(\nexists\, x A_1^1(x)) \to \forall u \exists y \forall x A_1^3(u,x,y)$;

Задача 3.3.7. Записать в виде формулы ИП и привести к предваренной нормальной форме следующие утверждения:

a) «Ты можешь обманывать кое–кого все время, ты можешь обманывать всех некоторое время, но ты не можешь обманывать всех все время.»
b) «Если всякий разумный философ — циник и только женщины являются разумными философами, то тогда, если существуют разумные философы, то некоторые из женщин — циники.»

Задача 3.3.8. Пусть заданы предикаты $A_1^1(x) \Leftrightarrow (x \in \mathbb{N})$, $A_1^2(m,n) \Leftrightarrow (m < n)$, $A_1^3(x,y,\varepsilon) \Leftrightarrow (|x-y| < \varepsilon)$, константа 0.

a) Записать в виде формулы исчисления предикатов утверждение, что $\lim_{n\to\infty} a_n = Q$. (можно считать $a_n = a(n)$ — функция, определенная для натуральных аргументов).
b) Привести полученную формулу к предваренной нормальной и к сколемовской нормальной форме;
c) Записать утверждение о том, что не существует предела последовательности $\{b_n\}$ и привести к сколемовской форме.
d) * Записать теорему Вейерштрасса (про существование предела у неубывающей ограниченной последовательности) в виде формулы ИП. Разрешается использовать предикат $A_2^2(x,y) \Leftrightarrow x \leqslant y$.

Задача 3.3.9. Записать в виде формулы ИП и проверить выполнимость: *Во всех зоопарках, где есть слоны и носороги, нет жирафов. Во всех зоопарках, где есть носороги и нет жирафов, есть слоны. Наконец, во всех зоопарках, где есть слоны и жирафы, есть и носороги. Может ли существовать такой зоопарк, в котором есть слоны, но нет ни жирафов, ни носорогов?*

Задача 3.3.10. Сформулировать в виде формулы ИП. Привести к предваренной НФ и проверить общезначимость:

a) Некоторые улитки являются горами. Все горы любят кошек. Следовательно, все улитки любят кошек.
b) Все крокодилы могут летать. Все великаны являются крокодилами. Следовательно, все великаны могут летать.
c) Две рощи никогда не похожи друг на друга. Сосны и ели выглядят совершенно одинаково. Следовательно, сосны и ели не являются двумя рощами.
d) Никто не может стать президентом, если у него красный нос. У всех людей нос красный. Следовательно, никто не может быть президентом
e) Все вороны собирают картины. Некоторые собиратели картин сидят в птичьей клетке. Значит, некоторые вороны сидят в птичьей клетке.

Задача 3.3.11. Записать в виде формулы исчисления предикатов и доказать общезначимость, используя правила резолюции и склеивания:

a) Всякий парикмахер в деревне бреет тех и только тех, кто не бреется сам. Следовательно в деревне нет ни одного парикмахера.
b) Всякий, кто находится в здравом уме, может понимать математику. Ни один из сыновей Гегеля не может понимать математику. Сумасшедшие не допускаются к голосованию. Следовательно ни один из сыновей Гегеля не будет допущен к голосованию.
c) Если для любого множества x существует множество y, такое, что мощность y больше, чем мощность x. Если x включено в y, то мощность x не больше мощности y. Всякое множество включено в V. Следовательно V — не множество.

Задача 3.3.12. Предикат $P(x, y)$ удовлетворяет условиям:

- $(\forall x)P(x, x)$;
- Для всех предикатов $A(\cdot)$ формула $(P(x, y) \to (A(x) \to A(y))$ — общезначимая.

Доказать, что в любой области интерпретации P есть предикат тождественного равенства, т.е. $P(x, y) \leftrightarrow (x = y)$.

Задача 3.3.13. Запишите аксиомы группы, используя предикаты $E^2(x, y) \Leftrightarrow (x = y)$ и $G^3(x, y, z) \Leftrightarrow (x \circ y = z)$. Как записать условие того, что группа Абелева?

Задача 3.3.14. Пусть $\mathfrak{M} = \left\langle \mathbb{N}, S^3, P^3, 1 \right\rangle$, где S^3 и P^3 — предикаты, заданные формулами: $S^3(x, y, z) \Leftrightarrow (x + y = z)$ и $P^3(x, y, z) \Leftrightarrow (x \cdot y = z)$.

a) Записать формулу с одной свободной переменной p истинную в $\mathfrak{M}$ тогда и только тогда, когда p — простое число.
b) Привести эту формулу к предваренной форме.

Задача 3.3.15. Привести пример формулы ИП, общезначимой в любой предметной области мощности 2, но не общезначимой в некоторой предметной области большей мощности.

Задача 3.3.16. Предикаты $F(x)$ и $G(x)$ заданы алгебраическими уравнениями $F(x) \Leftrightarrow (f(x) = 0)$, $G(x) \Leftrightarrow (g(x) = 0)$. Выписать алгебраические уравнения, задающие предикаты:

a) $F(x) \vee G(x)$;
b) $F(x) \& G(y)$;
c) Выписать уравнение (не обязательно алгебраическое), задающее предикат $\neg F(x)$.

Задача 3.3.17. Пусть $P_i(x)$, $i = 1, 2, 3, 4$ — некоторые многочлены. Заданы предикаты:

$$A_1^1(x) \Leftrightarrow \left[\begin{array}{l} \begin{cases} P_1(x) < 0, \\ P_2(x) < 0 \end{cases} \\ \begin{cases} P_3(x) < 0, \\ P_4(x) < 0. \end{cases} \end{array}\right. \quad \text{и } A_2^1(x) \Leftrightarrow \begin{cases} \left[\begin{array}{l} P_1(x) < 0, \\ P_3(x) < 0 \end{array}\right. \\ \left[\begin{array}{l} P_2(x) < 0, \\ P_4(x) < 0. \end{array}\right. \end{cases}$$

Какие из приведенных формул являются общезначимыми:

a) $\exists x A_1^1(x) \leftrightarrow A_2^1(x)$;
b) $\forall x A_1^1(x) \leftrightarrow A_2^1(x)$;
c) $\exists x A_1^1(x) \to A_2^1(x)$;
d) $\forall x A_1^1(x) \to A_2^1(x)$;
e) $\exists x A_1^1(x) \leftarrow A_2^1(x)$;

Задача 3.3.18. Докажите, что если какое-то тождество, содержащее переменные для множеств и операции $\cap$, $\cup$, $\setminus$, неверно, то можно найти контрпример к нему, в котором множества пусты или содержат по одному элементу.

Ответы и указания.

3.3.1a) $x \in E_2$, $\Phi(x) = x$, $P(x) = \bar{x}$.
b) Те же, что и в предыдущем пункте.
c) $x, y \in E_2$, $P(x, y) = x \oplus y$.

3.3.2a) Да.
b) Да.
c) Нет.
d) Нет.
e) Нет.

3.3.3 a) $A_2^2(1,p)\&\left(\neg\big(\exists x A_2^2(1,x)\&A_2^2(x,p)\&A_1^2(p,x)\big)\right)$. Хотелось бы заметить, что 1 — **не** простое число, поэтому член $A_2^2(1,p)$ необходим.
b) $\neg\left(\exists x\big(A_1^2(a,x)\&A_1^2(b,x)\&A_2^2(1,x)\big)\right)$.
c) $A_1^2(a,c)\&A_1^2(b,c)\&\neg\left(\exists x\big(A_1^2(a,x)\&A_1^2(b,x)\&A_2^2(c,x)\big)\right)$.
d) $A_2^2(1,c)\&\left(\forall x\big(A_2^2(c,x)\&A_1^2(1,x)\big)\to A_1^2(x,2)\right)$. Пользуемся тем, что натуральная степень двойки не имеет нечетных делителей (кроме 1).
e) $A_2^2(1,c)\&\left(\exists p A_1^1(p)\&\left(\forall x\big(A_2^2(c,x)\&A_1^2(1,x)\big)\to A_1^2(x,p)\right)\right)$.

3.3.4 a) $\forall\varepsilon\left(P_2^2(0,\varepsilon)\to\left(\exists\delta P_2^2(0,\delta)\&\big(P_1^3(0,x,\delta)\to P_1^3(A,f(x),\varepsilon)\big)\right)\right)$.
b) $\forall A\exists\varepsilon\left(P_2^2(0,\varepsilon)\&\neg\left(\exists\delta P_2^2(0,\delta)\&\big(P_1^3(0,x,\delta)\to P_1^3(A,f(x),\varepsilon)\big)\right)\right)$.
c) $\forall\varepsilon\left(P_2^2(0,\varepsilon)\to\left(\exists\delta P_2^2(0,\delta)\&\big(P_1^3(x_0,x,\delta)\to P_1^3(f(x_0),f(x),\varepsilon)\big)\right)\right)$.
d) Тут необходимо учесть, что в концах отрезка непрерывность будет односторонней.

$$\forall x_0 P_2^2(x_0,a)\vee P_2^2(b,x_0)\vee\Bigg(\forall\varepsilon\exists\delta P_2^2(0,\varepsilon)\to$$
$$\to\left(P_2^2(0,\delta)\&\left(\forall x P_2^2(x,a)\vee P_2^2(b,x)\vee\big(P_1^3(x_0,x,\delta)\to P_1^3(f(x_0),f(x),\varepsilon)\big)\right)\right)\Bigg).$$

e) Обозначим результат предыдущего пункта $C_{[a,b]}(f)$ (это не предикат, а сокращение!).

$$C_{[a,b]}(f)\to\Bigg(\forall\varepsilon P_2^2(0,\varepsilon)\to\bigg(\exists\delta P_2^2(0,\delta)\&\Big(\forall x_1\forall x_2 P_2^2(x_1,a)\vee P_2^2(x_2,a)\vee$$
$$\vee P_2^2(b,x_1)\vee P_2^2(b,x_2)\vee\big(P_1^3(x_1,x_2,\delta)\to P_1^3(f(x_1),f(x_2),\varepsilon)\big)\Big)\bigg)\Bigg)$$

3.3.5 a) $\exists x\forall u(A_1^2(x,y)\vee\neg A_1^1(x)\vee\neg A_1^2(x,u)$;
b) $\forall x\exists y_1\forall z_1\forall y_2\exists z_2\left(A_1^1(x)\&A_1^3(x,y_1,z_1)\vee(\neg A_1^1(x))\&(\neg A_1^3(x,y_2,z_2)\right)$;
c) $\forall x\forall t\exists z\left(A_1^1(x)\&\big(\neg A_1^2(x,y)\big)\vee\big(\neg A_1^1(t)\big)\vee A_1^2(y,z)\right)$.

3.3.6 a) $\forall x\forall u A_1^4(x,f(x),u,g(x,u))$;
b) $\forall u\forall x\big(A_1^1(c_0)\vee A_1^3(u,x,f(u))\big)$.

3.3.7 a) Обозначим $P_1^2(x,t)\Leftrightarrow$ (ты можешь обманывать x в момент t). Тогда утверждение запишется в виде:

$$\big(\exists x_1\forall t_1 P_1^2(x_1,t_1)\big)\&\big(\exists t_2\forall x_2 P_1^2(x_2,t_2)\big)\&\neg\big(\forall x_3\forall t_3 P_1^2(x_3,t_3)\big).$$

Приводя к предваренной нормальной форме, получим:

$$\exists x_1\forall t_1\exists t_2\forall x_2\exists x_3\exists t_3\big(P_1^2(x_1,t_1)\&P_1^2(x_2,t_2)\&\neg P_1^2(x_3,t_3)\big).$$

b) Введем обозначения $P_1^1(x)\ \Leftrightarrow\ (x$ — разумный философ$)$, $P_2^1(x)\ \Leftrightarrow$ $\Leftrightarrow\ (x$ — циник$)$ и $P_1^3(x)\Leftrightarrow(x$ — женщина$)$. Тогда утверждение можно записать следующим образом:

$$\Big(\big(\forall x_1 P_1^1(x)\to P_2^1(x)\big)\&\big(\forall x_2 P_3^1(x)\leftarrow P_1^1(x)\big)\Big)\to\Big(\exists x_3 P_1^1(x_3)\to\exists x_4\big(P_3^1(x_4)\to P_2(x_4)\big)\Big).$$

Приведем к нормальной форме:

$$\exists x_1 \exists x_2 \forall x_3 \exists x_4 \left(P_1^1(x_1) \& (\neg P_2^1(x_1)) \vee (\neg P_3^1(x_2)) \& P_1^1(x_2) \vee (\neg P_1^1(x_3)) \vee (\neg P_3^1(x_4)) \vee P_2^1(x_4)\right).$$

3.3.8 a) Стандартное определение из учебника мат. анализа выглядит так:

$$\forall \varepsilon > 0 \exists N = N(\varepsilon) : \forall n > N(|a_n - Q| < \varepsilon).$$

Заметим, что от N можно не требовать натуральности, а для n она должна быть. Перепишем формулу в виде:

$$\forall \varepsilon((\varepsilon > 0) \Rightarrow (\exists N(\forall n(n \in \mathbb{N}) \& (n > N) \Rightarrow (|a_n - Q| < \varepsilon)).$$

Теперь, подставляя соответствующие предикаты, получим:

$$\forall \varepsilon(A_1^2(0, \varepsilon) \Rightarrow \left(\exists N \forall n(A_1^1(n) \& A_1^2(N, n) \Rightarrow A_1^3(a_n, Q, \varepsilon))\right).$$

b) Заменим импликации:

$$\forall \varepsilon \neg A_1^2(0, \varepsilon) \vee \left(\exists N \forall n \neg A_1^1(n) \vee \neg A_1^2(N, n) \vee A_1^3(a_n, Q, \varepsilon)\right).$$

Переносим кванторы в начало:

$$\forall \varepsilon\, \exists N\, \forall n \left(\neg A_1^2(0, \varepsilon) \vee \neg A_1^1(n) \vee \neg A_1^2(N, n) \vee A_1^3(a_n, Q, \varepsilon)\right).$$

Сколемизируем:

$$\forall \varepsilon\, \forall n \left(\neg A_1^2(0, \varepsilon) \vee \neg A_1^1(n) \vee \neg A_1^2(N(\varepsilon), n) \vee A_1^3(a_n, Q, \varepsilon)\right).$$

c) $\forall B \exists \varepsilon A_1^2(0, \varepsilon) \& \left(\forall N \exists n A_1^1(n) \& A_1^2(N, n) \& \neg A_1^3(b_n, B, \varepsilon)\right)$. Приведем к предваренной нормальной форме:

$$\forall B \exists \varepsilon \forall N \exists n A_1^2(0, \varepsilon) \& A_1^1(n) \& A_1^2(N, n) \& \neg A_1^3(b_n, B, \varepsilon).$$

Сколемизируем:

$$\forall B \forall N A_1^2(0, \varepsilon(B)) \& A_1^1(n(B, N)) \& A_1^2(N, n(b, N)) \& \neg A_1^3(b_n, B, \varepsilon(B)).$$

d) Запишем теорему Вейерштрасса:

$$\left(\forall m \forall n(A_1^1(m) \& A_1^1(n) \& A_2^2(m, n)) \to A_2^2(a_m, a_n)\right) \& \left(\exists M \forall n(A_1^1(n) \Rightarrow A_1^2(a_n, M))\right) \Rightarrow$$
$$\Rightarrow \left(\exists Q \forall \varepsilon(A_1^2(0, \varepsilon) \Rightarrow \left(\exists N \forall n(A_1^1(n) \& A_1^2(N, n) \Rightarrow A_1^3(a_n, Q, \varepsilon))\right)\right).$$

3.3.9 Пусть x — зоопарк, наличие слонов, носорогов и жирафов запишем в виде предикатов $A(x)$, $B(x)$ и $C(x)$, соответственно. Тогда условия задачи можно записать в виде: $\forall x((A(x) \& B(x)) \Rightarrow \neg C(x))$, $\forall x((B(x) \& \neg C(x)) \Rightarrow A(x))$ и $\forall x((A(x) \& C(x)) \Rightarrow B(x))$. а вопрос в виде $\exists x(A(x) \& \neg B(x) \& \neg C(x)$. Выполнимость очевидна при $A(x) =$ И, $B(x) = C(x) =$ Л

3.3.10 a) Обозначим $A_1(x) \Leftrightarrow (x$ — улитка); $A_2(x) \Leftrightarrow (x$ — гора); $A_3(x) \Leftrightarrow (x$ любит кошек). Тогда утвержение запишется в виде: $(\exists x(A_1(x) \& A_2(x))) \& (\forall x(A_2(x) \Rightarrow A_3(x))) \Rightarrow (\forall x(A_1(x) \Rightarrow A_3(x)))$. Предваренная форма: $\forall x \exists y \forall z\, ((\neg A_1(x)) \vee (\neg A_2(x)) \vee (A_2(y) \& \neg A_3(y)) \vee (\neg A_1(z)) \vee A_3(z))$.

Сколемизируем: $\forall x\forall z\,((\neg A_1(x))\vee(\neg A_2(x))\vee(A_2(y_0)\&\neg A_3(y_0))\vee(\neg A_1(z))\vee A_3(z))$. Формула не общезначима, т.к. легко построить модель, в которой она ложна: например $x\in E_2$, $A_1(x)=\text{И}$, $A_2(x)=A_3(x)=x$. Подставив, получим $\forall x\forall z(\neg\text{И}\vee\neg x\vee(y_0\&\neg y_0)\vee\neg\text{И}\vee z)$, после упрощения $\forall x\forall z(\neg x\vee z)$, что ложно при $x=\text{И}$, $z=\text{Л}$.
b) Общезначима. c) Общезначима. d) Не общезначима. e) Не общезначима.

3.3.11 a) Обозначим $A(x,y)\Leftrightarrow(x$ бреет $y)$, $B(x)\Leftrightarrow(x$ — парикмахер). Тогда утверждение запишется в виде:

$$(\forall x B(x)\Rightarrow(\forall y A(x,y)\Leftrightarrow\neg A(y,y)))\Rightarrow\neg\left(\exists x B(x)\right).$$

Приведем отрицание этого утверждения к предваренной форме, получим:

$$\exists z\forall x\forall y\big(\neg B(x)\vee(A(x,y)\&\neg A(y,y))\vee(\neg A(x,y)\&A(y,y))\big)\&B(z).$$

Сколемизируем и запишем в виде КНФ:

$$\forall x\forall y(\neg B(x)\vee A(x,y)\vee A(y,y))\&(\neg B(x)\vee\neg A(x,y)\vee\neg A(y,y))\&(B(z_0)).$$

Применим подстановку $y\to x$ и правило резолюции к первым двум дизъюнктам, получим дизъюнкт $\neg B(x)$. Применим к полученному и к третьему дизъюнкту подстановку $x\to z_0$ и правило резолюции, получим константу Л.

3.3.12 Если $x_0\neq y_0$, то можно построить предикат $A(x)\Leftrightarrow(x=x_0)$. Тогда $A(x_0)\to A(y_0)$ ложно, следовательно должен быть ложен предикат $P(x_0,y_0)$. А на одинаковых элементах этот предикат по первому свойству истинен. Значит это — предикат равенства.

3.3.13 Значала запишем то, что операция $\circ$ однозначно определена на элементах группы: $\forall x\forall y\big(\exists z G^3(x,y,z)\big)\&\big(\forall z_1\forall z_2 G^3(x,y,z_1)\&G^3(x,y,z_2)\to\to E^2(z_1,z_2)\big)$. Запишем определение единичного элемента e группы: $\forall x G^3(x,e,x)\&G^3(e,x,x)$. Существование обратного: $\forall x\exists x^{-1}G^3(x,x^{-1},e)\&G^3(x^{-1},x,e)$. Коммутативность: $\forall x\forall y\forall z G^3(x,y,z)\leftrightarrow G^3(y,x,z)$.

3.3.14 Запишем определение простого числа:

$$(p\text{ — простое})\Leftrightarrow(p>1)\&\forall x(1<x<p)\Rightarrow(p\not\vdots\, x).$$

Заметим, что отношение делимости $a\vdots b$ можно записывать как $\exists x P^3(x,b,a)$.
А условие $a<b$ можно записать как $\exists x S^3(x,a,b)$. В итоге приходим к формуле

$$\big(\forall x_1\left((\exists x_2 S^3(1,x_2,x_1)\&(\exists x_3 S^3(x_1,x_3,p))\Rightarrow\neg(\exists x_4 P^3(x_1,x_4,p))\right)\big)\&(\neg\exists x_5 S^3(1,x_5,p)).$$

Предваренная форма будет

$$\forall x_1,x_2,x_3,x_4,x_5\left((\neg S^3(1,x_2,x_1))\vee(\neg S^3(x_1,x_3,p))\vee(\neg P^3(x_1,x_4,p))\right)\&(\neg S^3(1,x_5,p)).$$

3.3.15 $\forall x, y, z(P_1(x) \leftrightarrow P_1(y)) \vee (P_2(y) \leftrightarrow P_1(z)) \vee (P_3(z) \leftrightarrow P_3(x))$. Если предметная область имеет мощность 2, то, очевидно, что из x, y, z какие-то два равны друг другу, откуда и вытекает общезначимость указанной формулы.

Для мощности большей 2 (например 3) можно предъявить пример. Пусть $x = 1$, $y = 2$, $z = 3$, $P_i(x) \leftrightarrow (x = i)$. Тогда в каждой скобке первое значение предиката истинно, а второе — ложно.

3.3.16a) $f(x)g(x) = 0$.
b) $f^2(x) + g^2(y) = 0$.
c) $\frac{f(x)}{f(x)} = 1$.

3.3.17 a) Не общезначима; b) Не общезначима; c) Общезначима; d) Общезначима; e) Не общезначима.

3.3.18 Пусть тождество имеет вид $F_1(A_1, \ldots, A_n) \equiv F_2(A_1, \ldots, A_n)$, где F_1, F_2 — формулы алгебры множеств. Сделаем замены: $A_i \to x_i$, $\cap \to \&$, $\cup \to \vee$, $A \setminus B \to a\&(\neg b)$. Получатся формулы Φ_1, Φ_2 исчисления высказываний. Если тождество неверно, то формула $\Phi_1(x_1, \ldots, x_n) \leftrightarrow \Phi_2(x_1, \ldots, x_n)$ не общезначима, следовательно, существует набор $a_1, ..., a_n$, на котором она принимает значение Л. Тогда рассмотрим $A_i^* = \begin{cases} \emptyset, & a_i = \text{Л} \\ \{a\}, & a_i = \text{И} \end{cases}$ — тождество будет опровергнуто наботом множеств $(A_1^*, \ldots, A_n^*)$.

Литература

[1] Александров П.С., Курс аналитической геометрии и линейной алгебры: Учебник. 2-е изд., стер., СПб.: изд-во «Лань», 2009, 512 стр.

[2] Алексеев Д.В., Использование метода В.Н. Козлова в образовательном процессе на кафедре МаТИС, Интеллектуальные системы, т. 17, № 1-4, с. 16-20

[3] *Ахо А., Хопкрофт Дж., Ульман Дж.* Построение и анализ вычислительных алгоритмов. — М.: Мир, 1979.

[4] Гаврилов Г.П., Сапоженко А.А. // Задачи и упражнения по дискретной математике: Учеб. пособие — 3-е изд., перераб. — М.:ФИЗМАТЛИТ, 2005, 416 с.

[5] *Дмитриев А.Н., Журавлев Ю.И., Кренделев Ф.П.* О математических принципах классификации предметов и явлений // Дискретный анализ. — 1966. — №7.

[6] *Кнут Д.* Искусство программирования для ЭВМ. Т. 3. Сортировка и поиск. — М.: Мир, 1978.

[7] Козлов В.Н., Математическое моделирование зрительного восприятия, Математические вопросы кибернетики, т. 6, 1996, с. 321-338.

[8] Козлов В.Н., Элементы математической теории зрительного восприятия, М.: Изд-во ЦПИ при мех.-мат. ф-те МГУ, 2001, 128 стр.

[9] Козлов В.Н., Доказательность и эвристика при распознавании визуальных образов, Интеллектуальные системы, т. 14, № 1-4, 2010 г., с. 35-52

[10] *Крушинский Л.В., Козлов В.Н., Кудрявцев В.Б.* О некоторых результатах применения математики к моделированию в биологии. // Математические вопросы кибернетики. — 1988. — Т. 1. — С. 52–88.

[11] *Королева З.Е.* О сравнении тестовых алгоритмов распознавания // Журнал выч. матем. и матем. физики. — 1975. — Т. 15, №3. — С. 749–756.

[12] Кудрявцев В.Б., Гасанов Э.Э., Подколзин А.С. Введение в теорию интеллектуальных систем. М.: Изд-во ф-та ВМиК МГУ, 2006, 208 с.

[13] *Мартин Дж.* Организация баз данных в вычислительных системах. — М.: Мир, 1980.

[14] *Переяславский В.И.* Об одном линейном методе распознавания образов // Комбинаторно-алгебраические методы в прикладной математике (Межвузовский сб.) — 1982.

[15] *Переяславский В.И.* О линейном тестовом алгоритме распознавания // ДАН СССР. — 1983. — Т. 271, №5.

[16] *Препарата Ф., Шеймос М.* Вычислительная геометрия: Введение. — М.: Мир, 1989.

[17] *Чегис И.А, Яблонский С.В.* Логические способы контроля электрических схем // Труды матем. ин-та им. В.А. Стеклова, АН СССР. — 1958. — Т. 51. — С. 270–360.

[18] *Яблонский С.В., Чегис И.А.* О тестах для электрических схем // Успехи матем. наук. — 1955. — Т. 10, вып.4(66). — С. 182–184.

[19] *Bottenbruch H.* Structure and use of ALGOL 60 // JACM (1962) **9**, 161-221.

[20] Kozlov V.N., Image Coding and Recognition and Some Problems of Stereovision, Pattern Recognition and Image Analysis, Vol.7, No. 4, 1997, pp. 448 - 466.