М. А. ТЫНКЕВИЧ
А. Г. ПИМОНОВ
Е. В. ПРОКОПЕНКО

ВВЕДЕНИЕ В ДИСКРЕТНУЮ МАТЕМАТИКУ (ОТ ТЕОРИИ К ПРАКТИКЕ)

Учебное пособие

МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ
РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Кузбасский государственный технический университет
имени Т. Ф. Горбачева»

М. А. ТЫНКЕВИЧ А. Г. ПИМОНОВ Е. В. ПРОКОПЕНКО

ВВЕДЕНИЕ В ДИСКРЕТНУЮ МАТЕМАТИКУ
(ОТ ТЕОРИИ К ПРАКТИКЕ)

Учебное пособие

Рекомендовано Сибирским региональным учебно-методическим центром высшего профессионального образования для межвузовского использования в качестве учебного пособия для студентов, обучающихся по направлению подготовки бакалавров 09.03.03 «Прикладная информатика»

Кемерово 2016

УДК 510.22:510.6:519.1

РЕЦЕНЗЕНТЫ

А. М. Гудов, доктор технических наук, доцент, федеральное государственное бюджетное образовательное учреждение высшего образования «Кемеровский государственный университет»

Кафедра вычислительной математики и компьютерного моделирования федерального государственного автономного образовательного учреждения высшего образования «Национальный исследовательский Томский государственный университет» (заведующий кафедрой *А. В. Старченко*, доктор физико-математических наук, профессор)

Тынкевич, М. А.

Введение в дискретную математику (от теории к практике) : учеб. пособие / М. А. Тынкевич, А. Г. Пимонов, Е. В. Прокопенко ; КузГТУ. – Кемерово, 2016. – 106 с.

ISBN 978-5-906805-97-3

В учебном пособии излагаются базовые понятия раздела математики, который принято называть дискретной математикой, основы теории множеств, символической логики и комбинаторного анализа. Значительное место уделено описанию теории графов и ее прикладным аспектам.

Предназначено для студентов, обучающихся по направлению подготовки бакалавров 09.03.03 «Прикладная информатика».

УДК 510.22:510.6: 519.1

ISBN 978-5-906805-97-3

ОГЛАВЛЕНИЕ

ПРЕДИСЛОВИЕ

У А. Н. Крылова в одной из басен, «петух, навозну кучу разрывая, нашел Жемчужное Зерно и говорит: «Куда оно? Какая вещь пустая!?». Аналогичный вопрос-утверждение в форме «Зачем она нужна эта математика?» регулярно звучит в устах современных школьников и студентов, стоящих в очереди за дипломом.

И, в самом деле, кому, зачем и на каком уровне?

Миллионы россиян до 1917-го года жили без математики и грамоты вообще, не помышляя о большем, чем церковно-приходская школа или четырехклассное начальное училище. Миллионы в мире и сейчас живут, ограничиваясь умением считать денежные купюры и барашков.

У создателя русской поэзии А. С. Пушкина в Царскосельском лицее на все вопросы по математике был однозначный ответ «ноль». И. Е. Репин и И. И. Левитан с их полуголодными и бесправными детством и юностью стали великими живописцами не в размышлениях на эту тему, а вопреки ожидавшему их будущему, благодаря незаурядному таланту и труду. Для великой балерины Г. С. Улановой выбор цели определился уже в детстве артистической семьей и природным талантом. Нелепо требовать любви к математике, физике и истории КПСС от неповторимой Е. С. Максимовой, в 10 лет выдержавшей конкурс в хореографическое училище при Большом театре (80 человек на место), обрекая при этом себя на жизнь в преодолении усталости и боли.

Увы, немногие могут выбрать предсказуемый жизненный путь в 7 и даже в 17 лет и предвидеть потребность в тех или иных знаниях. Равенство перед законом еще не определяет умственные способности, финансовые возможности и интеллектуальный уровень окружения. Подчас даже успешные выпускники элитных вузов (математики, физики) уходили из мира газовой динамики и теории взрыва в сферу искусства или становились незаурядными финансистами. Но иногда и известные специалисты в области горного дела и аналогичных сферах «садились за парту», чтобы расширить познания в математике и кибернетике.

В разные времена математическая наука и массовое математическое образование меняли значимость в глазах общества.

Математика родилась в связи с потребностями измерения площадей и налогообложения, строительства культовых сооружений, навигации по звездам, предсказания разливов рек и затмений светил. Еще до Пифагора человек строил прямые углы на местности с помощью веревки с узлами, отстоящими на 3, 4 и 5 единиц измерения. Трисекция угла, длина дуги, пропорции и прогрессии занимали умы Пифагора, Архимеда, Евклида, Эратосфена.

Говорят, что потомки быков, принесенных в жертву богам Пифагором после доказательства своей знаменитой теоремы, дрожат при открытии всего нового. Мудрость математики той поры была доступна немногим (и сегодня не всем дано внести новое в науку), и для фанатиков в 391 г. не было сомнений в том, что последнее прибежище «еретиков» – Александрийскую библиотеку – следует уничтожить.

Возрождение Европы после 800-летнего интеллектуального мрака Средневековья долго сохраняло дискретный характер объектов внимания математики, а занятия унаследованной от пифагорейцев магией чисел заставляли математиков балансировать между костром и благоволением властителя (даже Л. Эйлер вынужден был составлять гороскоп для Анны Иоановны).

В середине XVII века в споре о приоритете И. Ньютона и Г. Лейбница родилась новая **математика непрерывных величин** и стала именоваться «высшей» (объявив глупостью популярное заявление «на ноль делить нельзя» и бросив кость раздора философам в виде понятия «бесконечность»). Сегодня математика как наука достигла уровня, когда алгебраист не имеет понятия о математической физике и небесной механике. Новые элементарные частицы и планеты, лекарства и пластмассы возникают «на острие пера математика». За прошедшие столетия сложилась успешная европейская система образования в области точных и естественных наук.

Научно-технический рывок США в середине XX века во многом определился вовсе не «гениальностью» образовательной системы страны. От фашизма из Европы в 30-е годы в США бежала элита научного мира и, пусть не гениальные, но образованные в классических традициях ее помощники (не всем быть нобелевскими лауреатами, «и в искусстве на одного Аполлона приходится четверка лошадей»). Как говорил Остап Бендер, «идеи наши – бензин ваш», а предприимчивость деловых людей Америки известна всем.

В СССР в 50-е годы, с началом «оттепели», с завершением борьбы с кибернетикой, генетикой и неправильным языкознанием, с выходом из «шарашек» уцелевших интеллектуалов, молодежь с хорошей школьной математической подготовкой активно ринулась на не сулящие особых материальных благ физтехи и мехматы. Неплохой показатель IQ для поколения – в шахматы играл каждый десятый студент.

Имена И. В. Тамма, И. В. Ландау, М. В. Келдыша, А. П. Александрова и других советских физиков и математиков пользовались не меньшей популярностью, чем имена Н. Бора, Э. Ферми, М. Кюри или Н. Винера. Работы талантливых ученых из закрытых КБ и НИИ, подобных Ю. Б. Харитону, С. П. Королеву или Б. А. Бабаяну, и их учеников определили научно-технический прогресс страны на полвека вперед. Школьное и университетское образование в области точных и естественных наук получило признание в мире (увы, многое потеряно в застое конца XX и реформах XXI веков).

Прошло 250 лет, и сбылась мечта Митрофанушки об извозчике, который «свезет куда прикажут». Для перелета из Москвы в Рио-де-Жанейро не нужно знать географию и законы обтекания крыла. Для пользования телевизором не требуется знать его устройство. Владелец автомашины уже не лежит под ней с ключом в руках. Есть сферы, где нежелательный «человеческий фактор» заменен, по возможности, кнопкой на панели компьютера.

Но по-прежнему нужны знающие «извозчики», которые отремонтируют, попытаются объяснить или даже что-то усовершенствовать.

Сегодня уже не дискутируется вопрос о необходимости развития математической науки, поскольку современные физика, биология, ракетостроение, навигация, связь, лингвистика непознаваемы без знакомства с математикой. Школьников цивилизованного мира заставляют худо или бедно «учить математику», поскольку **познание основ математики – одно из немногих средств развития интеллекта, способности к логическому суждению и разумному восприятию мира**.

Подросток, по разным причинам не получивший нормального образования согласно потребностям века, не способный к логическому мышлению, не готовый к прозаическому труду, не видит перспективы в конкурентном мире, ищет виновных, идет в киллеры или усваивает примитивные лозунги спекулянтов от политики и фанатиков, обещающих райские кущи за убийство «неверных». Конечно, наличие какого-то образования не гарантирует защиты от идеологии нацизма, варварства и идиотизма, но снижает этот шанс.

Начало компьютерного века внесло свою специфику в традиционный образовательный процесс. Если ранее студент (школьник) за той или иной справкой обращался к учителю или к не всегда доступной книге, то сегодня ему достаточно войти в интернет. Но всякий ли пользователь может хоть что-то понять в этом кладезе мудрости?

Не имеющий и не жаждущий знаний студент неделями тупо смотрит на первую страницу электронного учебника, в итоге ассоциирует поиск корня уравнения со стоматологией и не отличает сплайн от спрайта. Не редкость, когда такой студент, получив как-то диплом о медицинском, юридическом или техническом образовании, при наличии гибкости скелета и умения повторять лозунги, делает успешную карьеру администратора (без всяких кодексов Юстиниана и Наполеона или частных производных…). Да и реформаторы образования, «в соответствии с требованиями отнюдь не наукоемкого производства», путают функции университетов и профессиональных училищ.

Дискретная математика с ее относительно простыми постановками задач, но трудоемкими комбинаторными методами, достаточно долго пребывала на втором плане, но с ростом возможностей средств вычислительной техники нашла реальный выход в теорию вероятностей, теорию графов, математическую лингвистику, робототехнику, теорию информации и другие сферы, где требуется интеллект человека. Знакомство с азбукой дискретной математики обязательно для будущего специалиста и не бесполезно для просто образованного человека, наряду с историей, театром, живописью или музыкой…

Глава 1. НАИВНАЯ ТЕОРИЯ МНОЖЕСТВ

1.1. Предыстория теории множеств

Создателем так называемой наивной (интуитивной) теории множеств является немецкий математик Георг Кантор (1845–1918), который ввел в математику учение о множествах (нем., *Mengenlehre*) и фундаментальные ныне ее понятия, такие как *взаимно-однозначное соответствие* между элементами множеств и *мощность* множества, расширив представление о *порядковых числах* от привычных для человека натуральных чисел до *трансфинитных* и допустив существование «бесконечного множества бесконечных множеств».

Теория Кантора о трансфинитных числах и идеи стандартизации математики на базе понятия «множества» были встречены резкой критикой части его современников.

Георг Кантор

Так, Леопольд Кронекер с его известной фразой «бог создал натуральные числа, а все прочее – дело рук человеческих» полагал, что математическими объектами могут считаться только натуральные числа и то, что к ним сводится, и яростно отвергал идеи Кантора.

С другой стороны, немецкий математик Эрнст Шрёдер еще в 1895 году обращает внимание на совпадение алгебры множеств и исчисления высказываний, тем самым устанавливая связь теории множеств с математической логикой.

Давид Гильберт, будущий лидер мировой математической науки, оказался прав, заявив: «Никто не изгонит нас из рая, который основал Кантор». Несмотря на обнаруженные парадоксы и возражения ряда философов и богословов, теоретико-множественный инструментарий успешно используется не только в современном математическом сообществе, но и в прикладной науке.

1.2. Символика. Операции над множествами

По определению Кантора, «множество есть многое, мыслимое нами как целое». В рамках наивной теории множеств (set theory) *множеством* считается любой *четко определенный набор объектов* (*элементов множества*).

В обиходе мы используем слова-синонимы «коллекция», «компания», «группа», «стадо» и пр., в математике – «семейство», «набор», «совокупность» и др.

Объекты, составляющие множество, различны, но единой природы – множество целых чисел, множество букв алфавита русского языка, множество депутатов Государственной думы и т. п. (естественно требование, что среди элементов одного множества не может быть совпадающих).

Обычно в математике множество обозначают прописными латинскими или иными буквами: N_0 – множество натуральных (неотрицательных целых) чисел; Q – рациональных, R – вещественных.

Пустое множество обозначают символом $\varnothing$.

При отсутствии стандартного обозначения используют перечисление элементов множества, например РАДУГА = («красный», «оранжевый», «желтый», «зеленый», «голубой», «синий», «фиолетовый»), указание признака в виде $\{x \,/\, \sin(x) = 0\}$ (множество всех x, синус которых равен нулю) или $\{x \in N_0 \,/\, x^2 < 1000\}$ (множество натуральных чисел x, квадраты которых не превышают 1000).

Достаточно часто в литературе используются обозначения типа $k = 1, 2, \ldots, n$; $k = 1 \div n$; $k = \overline{1, n}$, символизирующие то, что объект k последовательно принимает значения из упорядоченного множества натуральных чисел от 1 до n. Однако в компьютерных средах, использующих переменную типа `set` и логический оператор `if A in B`, не предполагают какой-то упорядоченности элементов массива.

Множества по *мощности* (количеству составляющих их элементов) разделяются на конечные и бесконечные. *Мощность конечного* множества A обозначают $|A|$. В случае бесконечных множеств мощность обозначается стандартными знаками – *кардинальными числами*. Так, мощность множества всех вещественных чисел $|R|$ обозначают символом c (от *continuum*), мощность счетного множества $\aleph_0$ (алеф-нуль) и др.

Для множеств определены *операции отношения*:

$a \in A$ – a есть элемент множества A;

$a \notin A$ – a не является элементом множества A;

$A \subset B$ (или $B \supset A$) – A есть подмножество множества B;

$A = B$ – A совпадает с B ($A \subset B$ и $B \subset A$);

$A \neq B$ – A не совпадает с B;

$A \subset\subset B$ (или $A \subseteq B$) – A строго содержится в B ($A \subset B$ и $A \neq B$).

Как и в общеизвестной алгебре действительных чисел, для создания новых множеств введены некоторые *операции над множествами*:

$A \cap B$ – *пересечение* множеств ($z \in A$ и $z \in B$) – совокупность всех элементов, принадлежащих обоим множествам;

$A \cup B$ – *объединение* множеств ($z \in A$ или $z \in B$) – совокупность всех элементов, принадлежащих хотя бы одному из множеств;

$A \setminus B$ – *разность* множеств ($z \in A$, $z \notin B$) – совокупность всех элементов, принадлежащих A, но не принадлежащих B.

Эти операции хорошо иллюстрируются нижеприведенными диаграммами Венна – Эйлера (рис. 1).

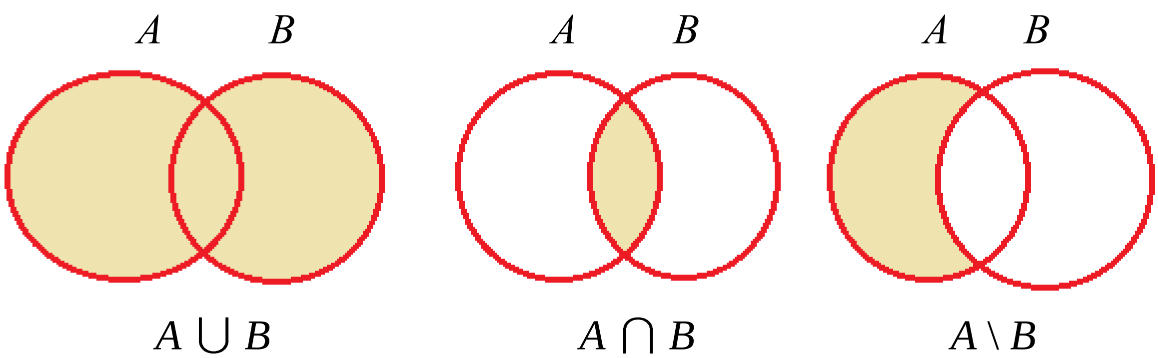


Рис. 1. Операции над множествами (диаграммы Венна – Эйлера)

Наряду с указанными операциями вводится операция *дополнения* (*отрицания*), представимая в виде $\bar{A}$ или $\neg A$ и соответствующая $\{x \mid x \notin A\}$ – все элементы за пределами множества A.

С помощью диаграммы Эйлера, например, легко показать что, если $A \bigcap B = \varnothing$, то $A \setminus B = A$ и $B \bigcap \bar{A} = B$, или дать представление о так называемой *симметрической разности* $(A \bigcup B) \setminus (A \bigcap B)$, если $A \bigcap B \neq \varnothing$.

Если создать несколько множеств с одинаковым именем и сопроводить их порядковым номером (индексом), например A_1, A_2, A_3, то вместо записи $A_1 \bigcup A_2 \bigcup A_3$ можно использовать эквивалентную конструкцию $\bigcup\limits_{i=1}^{3} A_i$.

Любопытный пример использования диаграммы Венна для демонстрации всех пересечений букв греческого, русского и латинского алфавитов читатель может обнаружить в Википедии (рис. 2).

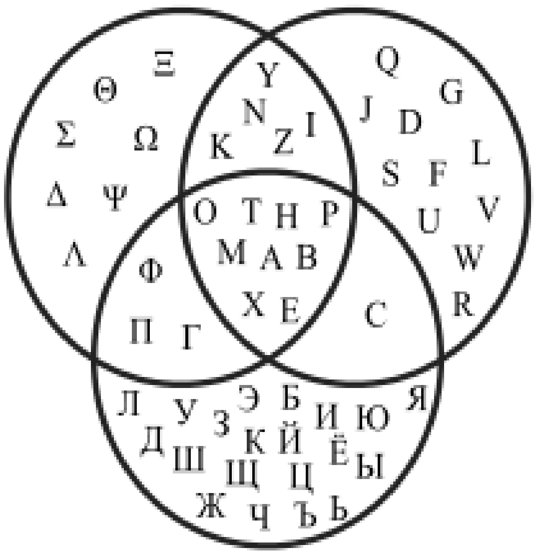


Рис. 2. Сборная алфавитов

Основные свойства объединения и пересечения определяются утверждением, что для любых подмножеств A, B и C некоторого универсального множества имеют место равенства:

$A \bigcup B = B \bigcup A$;

$A \bigcap B = B \bigcap A$;

$A \bigcup (B \bigcup C) = (A \bigcup B) \bigcup C$;

$A \bigcap (B \bigcap C) = (A \bigcap B) \bigcap C$;

$A \bigcup (B \bigcap C) = (A \bigcup B) \bigcap (A \bigcup C)$;

$A \bigcap (B \bigcup C) = (A \bigcap B) \bigcup (A \bigcap C)$.

Если первые четыре равенства, определяющие свойства коммутативности и ассоциативности, по аналогии с знакомой всем арифметикой очевидны, то на последние два читателю следует обратить более пристальное внимание.

1.3. Декартово произведение множеств

Особый интерес для многочисленных приложений в алгебре множеств, основы которой приведены выше, занимает операция $A \times B$ – прямое (декартово) произведение множеств: множество пар (a, b), где $a \in A$, $b \in B$.

Например, $\{a, b\} \times \{x, y, z\} = \{(a, x), (a, y), (a, z), (b, x), (b, y), (b, z)\}$ (рис. 3).

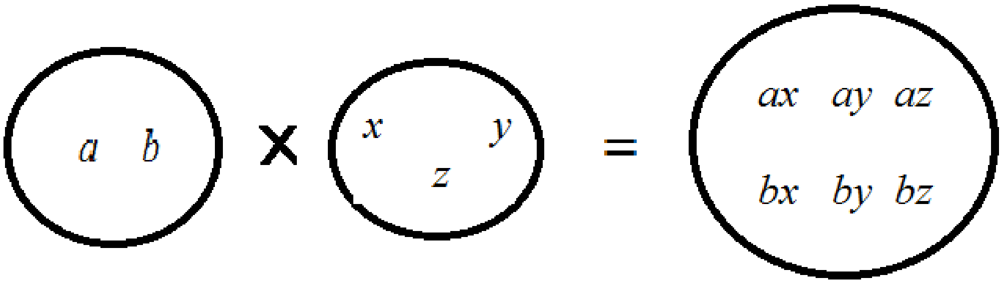


Рис. 3. Декартово произведение множеств

Следует помнить, что создаваемые пары являются упорядоченными, т. е. $(a, b) = (c, d)$ только при $a = c$ и $b = d$.

Если X и Y – множества действительных чисел, то их декартово произведение $X \times Y = \{(x, y), x \in X, y \in Y\}$ определяет известную всем декартову систему координат на плоскости. Метод привязки положения точки на плоскости с помощью прямоугольной системы координат, позже получившей название декартовой, был предложен Рене Декартом (1596–1650) и впоследствии распространен на пространства большей размерности.

Например, в шахматной нотации используется декартово произведение множества символов $\{a, b, c, d, e, f, g, h\}$ на множество целых чисел от 1 до 8, во избежание лишних слов «гроссмейстер О. Бендер сходил пешкой …» пишут $e2 – e4$.

Если $A = \{$символы прописных букв русского алфавита$\}$, $B = \{$символы прописных букв латинского алфавита$\}$, то можно видеть, что $A \cap B$ определяет множество букв, используемых в номерных знаках легковых автомашин как в России, так и в остальной Европе.

Обозначив через $A = \{$символы прописных букв русского алфавита}, B = {символы прописных букв латинского алфавита}, $N = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$ и выглянув в окно на улицу, можно убедиться в том, что множество номерных знаков легковых автомобилей (без учета номера региона) совпадает с $\{A \cap B\} \times N \times N \times N \times \{A \cap B\} \times \{A \cap B\}$.

Пусть A – множество выявленных в вашей фирме агентов конкурента AA и B – аналогичное множество конкурента BB. Попробуйте оценить реакцию конкурентов, если вы сообщите им информацию о $A \cap B$?

Наконец, если через A, B и C обозначены множества возможных показателей трех параметров состояния экономической системы, то $A \times B \times C$ задает множество гипотетически возможных состояний этой трехпараметрической системы, которое должно быть подвергнуто анализу специалистов на предмет их соответствия реальности.

Мы не пытались изложить здесь *теорию множеств* как математическую дисциплину со всеми ее сложностями и парадоксами, ограничившись лишь базовыми интуитивными понятиями, позволяющими наглядно и без излишнего многословия формулировать какие-то истины.

В дальнейшем читатель может убедиться, что алгебра множеств полезна при знакомстве с азами математической логики, теории графов, комбинаторики, теории вероятностей и многих других сфер применения идей и методов дискретной математики.

1.4. Упражнения

1. Найдите $A \cup B$, $A \cap B$, $A \setminus B$, $B \setminus A$, если:
   a. $A = \{a, b, c, d, e\}$, $B = \{a, c, e\}$;
   b. $A = \{a, b, c, d, e\}$, $B = \{k, l, m\}$;
   c. $A = \{a, b, c, d, e\}$, $B = \varnothing$.
2. Найдите $A \cup B$, $A \cap B$, $A \setminus B$, $B \setminus A$, если:
   a. $A = [2; +\infty)$, $B = (1; 7]$;
   b. $A = [-7; -4]$, $B = (0; 3)$;
   c. $A = (-\infty; 10)$, $B = [-1; 5)$;

d. $A = [0; 3]$, $B = [3; 6)$.

3. Даны множества $A = \{1, 2, 4, 6, 8\}$, $B = \{3, 4, 6\}$, $C = \{1, 5, 7\}$, $U = \{0, 1, 2, 3, 4, 5, 6, 7, 8, 9\}$. Найдите:

a. $(A \cap B) \setminus (A \cup C)$;

b. $(A \cup C) \setminus (A \cap B)$;

c. $B \setminus (A \cup C)$.

4. Докажите: $A \setminus (B \cup C) = (A \setminus B) \cap (A \setminus C)$, $A \setminus (B \cap C) = (A \setminus B) \cup (A \setminus C)$.

5. Докажите: $A \setminus (A \setminus B) = A \cap B$, $A \setminus B = A \setminus (A \cap B)$.

6. Докажите: $A \cap (B \setminus C) = (A \cap B) \setminus (A \cap C) = (A \cap B) \setminus C$, $(A \setminus B) \setminus C = (A \setminus C) \setminus (B \setminus C)$.

7. Докажите: $A \cup B = A \cup (B \setminus A)$, $(A \cap B) \cup (A \cap B) = A$.

8. Докажите: $(A \cup B) \setminus C = (A \setminus C) \cup (B \setminus C)$, $A \setminus (B \setminus C) = (A \setminus B) \cup (A \cap C)$.

9. Докажите: $A \setminus (B \cup C) = (A \setminus B) \setminus \mathrm{C}$, $A \cup B \subset C \Leftrightarrow A \subset C$ и $B \subset C$.

10. Докажите: $A \subset B \cap C \Leftrightarrow A \subset B$ и $A \subset C$, $A \subset B \cup C \Leftrightarrow A \cap B \subset C$.

11. Докажите: $A \subset B \Rightarrow C \setminus B \subset C \setminus A$, $A \subset B \Rightarrow B \subset A$.

12. Докажите: $A \cup B = A \cap B \Rightarrow A = B$, $A = B \Leftrightarrow A \cap B = \varnothing$ и $A \cup B = U$.

13. В группе 32 студента. Из них 18 посещают спортивную секцию, 12 – робототехнику, 8 – не посещают ни одну из них. Сколько студентов посещают и спортивную секцию, и робототехнику? Сколько студентов посещают только спортивную секцию?

14. В группе 30 студентов, из них 18 увлекаются плаванием, а 17 – волейболом.

а) Каким может быть минимальное число студентов, увлекающихся обоими видами спорта?

б) Каким может быть минимальное число студентов, увлекающихся хотя бы одним видом спорта?

15. Среди написанных на доске чисел, полученных случайным образом, 65 % делятся на 2, 70 % – на 3, 75 % – на 5. Каков наименьший процент чисел, кратных 30?

16. Все студенты первого курса КузГТУ направления подготовки «Прикладная информатика» изучают три системы программирования. В этом году 21 студент предпочли изучать Delphi, 10 выбрали VBA, 23 решили заниматься SharePoint. Кроме того, было 5 студентов, слушающих курс по Delphi и VBA, 2 изучают Delphi и SharePoint, 3 – SharePoint и VBA. Известно, что никто из студентов не посещает сразу три курса. Сколько студентов в группе? Сколько из них были увлечены только SharePoint?

17. Опрошено 110 любителей собаководства, 45 из них разводят дома пинчеров, 68 предпочитают овчарок, 87 – пуделей, 13 – пинчеров и пуделей, 11 – овчарок и пуделей, 10 – пинчеров и овчарок, 3 – содержат три вида собак.

а) Сколько человек держат овчарок, но не имеют пуделей?

б) Сколько человек разводят овчарок или пуделей, но не любят пинчеров?

в) У скольких нет ни пинчеров, ни пуделей?

г) Сколько человек разводят не только пуделей?

18. По итогам экзаменов из 30 студентов отличную оценку по математике имели 5 студентов, по физике – 7, по истории – 11, по математике и физике – 2, по математике и истории – 2, по физике и истории – 2, по всем трем предметам – 1. Сколько студентов получили хотя бы по одной отличной оценке?

19. Среди счастливчиков, кому повезло поймать золотую рыбку, пожелавших новую квартиру оказалось 15 человек, дорогую машину – 11, хорошую работу – 23, квартиру и машину – 7, квартиру и работу – 13, машину и работу – 10, все три желания загадало 4 человека. Сколько всего человек поймали золотую рыбку? Сколько среди них загадавших только одно желание?

20. Преподаватель решил узнать, кто из 25 студентов курса читал книги A, B и C. Результаты опроса оказались таковы: книгу A читали 15 студентов, книгу B – 17, книгу C – также 17. Книгу A или B читали 22 студента, A или C – 24, B или C – 21; все три книги прочли 9 студентов. Сколько студентов прочли только по одной книге? Сколько студентов не читали ни одной из этих трех книг?

Глава 2. МАТЕМАТИЧЕСКАЯ ЛОГИКА И АЛГЕБРА БУЛЯ

2.1. Логика в окружающем мире

«Латынь из моды вышла ныне. Так, если правду вам сказать, он знал довольно по-латыне, чтоб эпиграфы разбирать…», – писал А. С. Пушкин, но и ныне приятно щегольнуть знакомством с мудростью веков. Увы, не всякая броская фраза может служить руководством к действию. Так, лозунг «Pereat mundus et fiat justicia», обычно переводимый как «Пусть погибнет мир, но свершится правосудие» и понимаемый буквально служителем Фемиды, отнюдь не свидетельствует об интеллекте глашатая.

Знаменитый норвежский ученый и путешественник Тур Хейердал, в годы войны обучавшийся в школе радистов, так и не смог достичь «консенсуса» с сержантом, заявляя, что угол в 60º между носками сапог в строю не является значимым фактором в разгроме фашизма. Инженер, офицер или клерк, не осознавший последствий выполнения полученного приказа, но готовый ему добросовестно следовать, невзирая ни на что в окружающем его мире, подобен плохо запрограммированному роботу и потенциально опасен для общества, равно как и любая система безоговорочного подчинения.

Так называемая житейская логика, связанная с накопленным жизненным опытом и атрибутами мышления человека, плохо поддается формализации.

Тем не менее, едва ли найдется *человек разумный*, который в сомнениях о последствиях своего поступка (выбора, решения) не прибегнул к рассуждению «*если ... то ....*» или даже «*если ... то ...., иначе ...*», то есть проявил способность к так называемому логическому мышлению. Даже «Витязь на распутье» на картине В. В. Васнецова уже более 130 лет стоит перед альтернативой выбора из трех путей.

Тем более разработчик алгоритмов и программ едва ли не использует ключевых слов, присутствующих во всех алгоритмических языках, – `if`, `then`, `else`.

2.2. Символическая логика и алгебра Буля

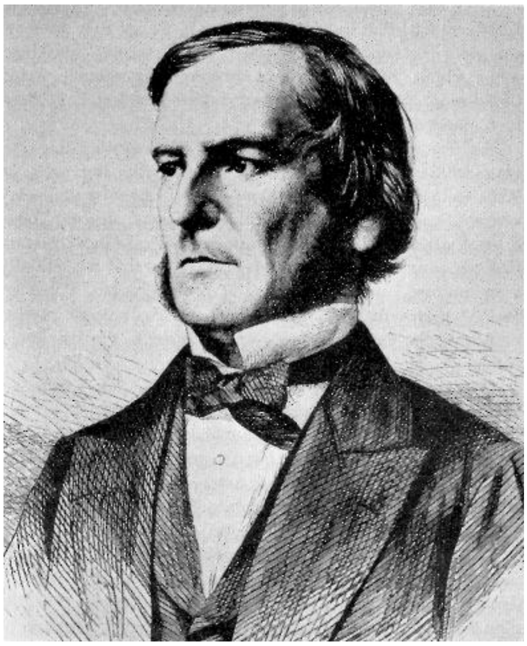
Создателем формальной логики как основы теории познания считается Аристотель, который 2400 лет назад дал изложение ее понятий и законов в трудах под названием «Органон». Его логика ставила целью «получение нового знания на основе ранее известного применением законов и правил мышления».

Не вдаваясь в многовековую и до сих пор продолжающуюся борьбу различных философских школ на поле формальной логики и вечный вопрос «Что есть истина?», обратимся к предмету нашего внимания.

Основу *математической (символической) логики* составляет *высказывание* – предложение, о котором можно сказать, *истинно* оно или *ложно*.

Так, заявление $2 \times 2 = 4$ истинно, заявление $2 \times 2 \geq 5$ ложно, но высказывание о «первичности яйца перед курицей» может быть принято за истину или отвергнуто лишь при определенных предпосылках.

Английский математик и логик **Джордж Буль** усмотрел аналогию между символическим аппаратом традиционной алгебры и символикой представления логических высказываний, показав, как из множества таких высказываний можно вывести любое заключение, следующее из этих высказываний, путем чисто символических манипуляций, и создав так называемую *булеву алгебру*, без которой немыслимо было начало компьютерной эры.



Джордж Буль
(1815–1864)

Наряду с логическими константами, обычно обозначаемыми символами `Истина` | `Ложь`, `True` | `False`, `1` | `0`, `И` | `Л` или `T` | `F`, в булевой алгебре простым высказываниям ставятся в соответствие логические переменные (A, B, C и т. п.), принимающие значения истинности или ложности.

Разумеется, существуют и *логические выражения (логические функции)* – сложные высказывания, получаемые в результате проведения логических операций над простыми высказываниями (логическими переменными и константами). Естественно, что при построении простых высказываний могут использоваться известные символы *операций отношения* $<$, $>$, $\leq$, $\geq$, $=$, $\neq$, $\in$, $\notin$ и другие.

При построении логических выражений используются *базовые логические операции*:

конъюнкция, логическое **И** – обозначение $\wedge$ или &;

дизъюнкция, логическое **ИЛИ** – обозначение $\vee$;

инверсия, отрицание, **НЕ** – обозначения последних лет $\neg$, $\ulcorner$, ~, но часто, как и в докомпьютерную эпоху, вместо $\neg a$ предпочитают писать $\overline{a}$.

Кстати, и в начале этой эры подготовка информации к вводу в память ЭВМ посредством пробивок на перфокартах или перфолентах велась на устройствах типа телеграфного аппарата с весьма ограниченным набором символов (возможности вывода графической информации практически не было; ЭВМ имели лишь АЦПУ – *алфавитно-цифровые* печатающие устройства). Поэтому в литературе по символической логике того времени вместо $A \wedge B$ писали AB и вместо $A \vee B$ – $A + B$.

Так, если обозначить через a и b соответственно истинность высказываний «Открыт Париж» и «Мне надо туда», то выражение $a \wedge \overline{b}$ определяет для поклонников творчества В. С. Высоцкого ситуацию «Открыт Париж, но мне туда не надо».

Наряду с базовыми используются производные от них логические операции: *исключающее ИЛИ* $\oplus$, *импликация* $\rightarrow$, *эквивалентность* $\leftrightarrow$, которые могут применяться при составлении логических выражений, но при дальнейшем анализе обычно выражаются за счет преобразований с помощью основных логических операций. Результаты выполнения всех логических операций вычисляются с помощью так называемых таблиц истинности (табл. 1).

Таблица 1. Таблицы истинности логических операций

$\neg$	$\wedge$	$\vee$	$\oplus$	$\rightarrow$	$\leftrightarrow$
	И $\wedge$ И = И	И $\vee$ И = И	И $\oplus$ И = Л	И $\rightarrow$ И = И	И $\leftrightarrow$ И = И
$\neg$И = Л	И $\wedge$ Л = Л	И $\vee$ Л = И	И $\oplus$ Л = И	И $\rightarrow$ Л = Л	И $\leftrightarrow$ Л = Л
$\neg$Л = И	Л $\wedge$ И = Л	Л $\vee$ И = И	Л $\oplus$ И = И	Л $\rightarrow$ И = И	Л $\leftrightarrow$ И = Л
	Л $\wedge$ Л = Л	Л $\vee$ Л = И	Л $\oplus$ Л = Л	Л $\rightarrow$ Л = И	Л $\leftrightarrow$ Л = И

Как и в элементарной алгебре, операции в логическом выражении имеют свои *приоритеты выполнения*:

1) инверсия (отрицание);

2) конъюнкция (логическое умножение);

3) дизъюнкция (логическое сложение), исключающее ИЛИ;

4) импликация, эквивалентность.

Так, выражение $a \vee b \wedge c \rightarrow d$ тождественно $(a \vee (b \wedge c)) \rightarrow d$.

Операцию $\oplus$ во многих приложениях (особенно компьютерных) называют *сравнением* или *поразрядным сложением по модулю 2*. Высказывание вида $A \oplus B$ в логическом выражении можно заменить на $A \wedge \overline{B} \vee B \wedge \overline{A}$.

В операции импликации $A \rightarrow B$ A называется *посылкой*, а B – *следствием* (если A, то B; из A следует B; A влечет B).

Как показано в табл. 1, выражение $A \rightarrow B$ утверждает, что истинная посылка порождает только верное, но никак не ложное следствие. При ложной посылке состояние следствия не предсказуемо (едва ли ложь влечет истину, но это лишь этическое соображение, с которым согласны не все прагматики).

Высказывание вида $A \rightarrow B$ можно заменить на $\overline{A} \vee B$.

Операция эквивалентности $A \leftrightarrow B$ выражается связками *тогда и только тогда*, *необходимо и достаточно*, *равносильно* и дает истину при совпадении операндов. Высказывание $A \leftrightarrow B$ можно заменить на $(A \wedge B) \vee (\overline{A} \wedge \overline{B})$.

Нетрудно видеть в изложенном выше существенную близость с аппаратом теории множеств, что приводит к определению одного из популярных в приложениях раздела прикладной математики – булевой алгебры.

Булева алгебра – это множество с элементами 0 (Ложь) и 1 (Истина), для которого определены операции ∧ (конъюнкция), ∨ (дизъюнкция), ¬ (отрицание) таким образом, что для всех a, b и c из этого множества верны законы (аксиомы), приведенные в табл. 2

Таблица 2. Аксиомы булевой алгебры

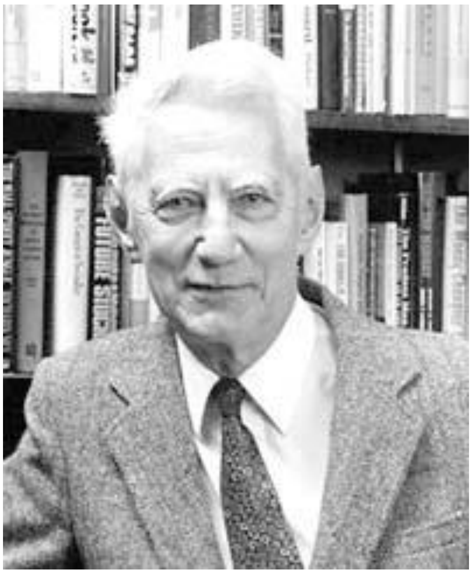
Аксиома	**Выражение**
Ассоциативность	$a \vee (b \vee c) = (a \vee b) \vee c$
	$a \wedge (b \wedge c) = (a \wedge b) \wedge c$
Коммутативность	$a \vee b = b \vee a$
	$a \wedge b = b \wedge a$
Дистрибутивность	$a \vee (b \wedge c) = (a \vee b) \wedge (a \vee c)$
	$a \wedge (b \vee c) = (a \wedge b) \vee (a \wedge c)$
Законы поглощения	$a \vee (a \wedge b) = a$
	$a \wedge (a \vee b) = a$
Дополнительность	$a \vee \overline{a} = 1$
	$a \wedge \overline{a} = 0$

Для оценки условий истинности той или иной формулы достаточно построить таблицу истинности. Так, для выражения $\overline{x \vee \overline{y}} \vee \overline{x} \wedge z$ вычисления приведены в табл. 3.

Таблица 3. Таблица истинности логического выражения

Переменные			**Промежуточные выражения**					**Формула**
x	y	z	$\overline{y}$	$x \vee \overline{y}$	$\overline{x \vee \overline{y}}$	$\overline{x}$	$\overline{x} \wedge z$	$\overline{x \vee \overline{y}} \vee \overline{x} \wedge z$
0	0	0	1	1	0	1	0	0
0	0	1	1	1	0	1	1	1
0	1	0	0	0	1	1	0	1
0	1	1	0	0	1	1	1	1
1	0	0	1	1	0	0	0	0
1	0	1	1	1	0	0	0	0
1	1	0	0	1	0	0	0	0
1	1	1	0	1	0	0	0	0

Булева алгебра прекрасно описывает работу арифметико-логических устройств ЭВМ – многообразие электрических схем (переключатели, триггеры, сумматоры и др.), компоненты которых находятся в одном из двух состояний: да – нет, включено – выключено и т. п. Так, если приведенная формула описывает функционирование некоторой системы с элементами такого рода, то легко видеть их состояние, необходимое для достижения нужного эффекта.



Клод Шеннон
(1916–2001)

Это увидел еще в 1930-е годы Клод Шеннон – американский инженер и математик, основатель *теории информации*, составляющей основу современных систем связи и *кибернетики* как синтеза теории автоматов и теории систем управления.

Именно он в 1948 году использовал слово «**бит**» для обозначения наименьшей единицы информации при изучении переключательных схем и ввел ее как способ анализа и проектирования схем.

Знаменитая формула Шеннона оценки информационной энтропии (меры неопределенности) в битах

$$H = \sum_{i=1}^{n} p_i \log_2 p_i$$

выступает как визитная карточка теории информации.

Использование вышеприведенных законов позволяет преобразовать некоторое исходное логическое выражение к тождественному, более простому, что важно как при конструировании автоматов, так и при разработке программных модулей.

Очевидно, что для достижения этой цели первоначально избавляются от всех производных операций в логическом выражении путем замены их на базовые эквиваленты.

Следствием ассоциативности являются формулы

$$a \vee (b \vee c) = (a \vee b) \vee c = a \vee b \vee c,$$

$$a \wedge (b \wedge c) = (a \wedge b) \wedge c = a \wedge b \wedge c,$$

позволяющие опускать скобки.

Свойство коммутативности операций позволяет менять порядок следования операндов в выражениях.

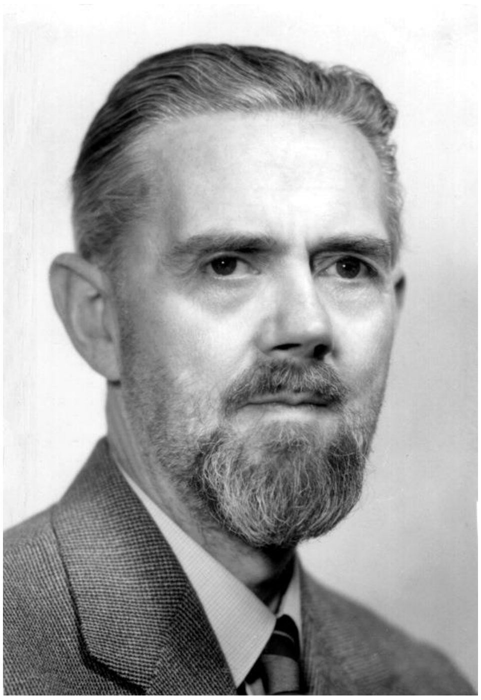
Так называемые законы Моргана позволяют исключать отрицания перед скобками: $\overline{a \wedge b} = \overline{a} \vee \overline{b}$ и $\overline{a \vee b} = \overline{a} \wedge \overline{b}$.

Закон *отрицания отрицания* (закон *двойного отрицания*) $\overline{\overline{a}} = a$ также способствует существенным упрощениям.

Например, взяв логическое выражение $\overline{C} \wedge D \vee \overline{C \vee D} \vee C$ и попытавшись его упростить, получаем:

$$\overline{C} \wedge D \vee \overline{C \vee D} \vee C = \overline{C} \wedge D \vee \left(\overline{C} \wedge \overline{D}\right) \vee C = \overline{C} \wedge \left(D \vee \overline{D}\right) \vee C = \overline{C} \wedge 1 \vee C =$$
$$= \overline{C} \vee C = 1.$$

Очевидно, что в случае многокомпонентных схем обработка соответствующего их описания вручную едва ли возможна и потому уже в конце 60-х годов началась разработка специализированных языков анализа и синтеза логических схем. Первой ласточкой в этой области был разработанный и реализованный на ЭВМ в 1962 г. в Томском университете А. Д. Закревским (1928–2014) *логический **я**зык **п**редставления **а**лгоритмов **с**интеза дискретных автоматов* – ЛЯПАС, известный за рубежом под названием Russian Programming Language – первый язык программирования логических задач [Закревский А. Д. Алгоритмический язык ЛЯПАС и автоматизация синтеза дискретных автоматов. Томск, 1966].



Уильям Росс Эшби
(1903–1972)

2.3. Серая логика и нечеткие множества

Еще в 60-е годы один из столпов кибернетики, английский психиатр, пионер в исследовании самоорганизации сложных систем **Уильям Росс Эшби** (1903–1972) сформулировал *принцип необхо-*

димого разнообразия: «разнообразие средств системы управления должно быть хотя бы не меньше, чем разнообразие управляемой ею ситуации». Соответственно, если с позиций классической черно-белой логики мы однозначно заявляем, что $a \in A$ или $a \notin A$, то в реальной жизни работает *серая логика* (это понятие впервые появилось в 1956 г. в работе У. Р. Эшби «Введение в кибернетику»).

В 1965 году Л. Заде ввел понятие **нечеткого множества** (*fuzzy set* – *размытого, туманного, пушистого*), в котором расширил классическое понятие множества, допустив, что функция принадлежности элемента множеству может принимать не только значения 0 или 1, но и любые значения на отрезке [0, 1]. Появившаяся на этой основе **нечеткая логика** (*fuzzy logic*) с ее отсутствием строгих стандартов применяется в экспертных системах, искусственных нейронных сетях, системах искусственного интеллекта. В нечеткой логике наряду с традиционными *истина* или *ложь* используются значения *возможно, иногда, почему бы и нет, не помню*.

Так, даже в сфере реализации арифметики в компьютерных расчетах не все однозначно. Например, оператор `if 2 * 2 = 4 then ...` несомненно даст вывод об истинности проверяемого условия, но в операторе `if 3 * (1 / 3) = 1 then ...` *скорее всего, наверное, с высокой степенью вероятности*, но не всегда проверяемое и подобное ему условие истинно – не следует забывать о двоичной системе представления данных, об ограниченности места под это представление в памяти компьютера и, соответственно, лишь о приближенных представлениях как обыкновенных и десятичных дробей, так и чисел с порядком, превышающим длину разрядной сетки мантиссы. При этом многое зависит от объявления типа переменной (`integer`, `real`, `double precision` и др.), ее значений и качества проработки тех или иных блоков транслятора.

2.4. Упражнения

1. Даны два высказывания *x*: «Петров учится в торговом институте» и *y*: $7 \leq 5$. Оцените истинность приведенных выражений, если *x* – истинное высказывание:

а) $\bar{x} \vee \bar{y}$; б) $\bar{x} \wedge \bar{y}$; в) $y \to x$; г) $\bar{y} \to \bar{x}$.

2. Даны высказывания: A = {принтер – устройство ввода информации}, B = {процессор – устройство обработки информации}, C = {монитор – устройство хранения информации}, D = {клавиатура – устройство ввода информации}.

Определите истинность высказываний:

а) $(A \wedge B) \wedge (C \vee D)$; б) $(A \vee B) \Leftrightarrow (C \,\&\, D)$; в) $A \leftrightarrow B$.

3. В предположении, что высказывания x и y – ложные, а z – истинное, оцените значение выражения $(\overline{\overline{x \to z}}) \wedge (y \vee (\bar{z} \to x))$.

4. Какие из следующих высказываний равносильны:

а) $\overline{x \to y}$; б) $x \wedge \bar{y}$; в) $x \to \bar{y}$; г) $y \to x$; д) $\bar{y} \to \bar{x}$?

5. Составьте таблицу истинности для формулы $\bar{x} \wedge y \vee \overline{x \vee y} \vee x$ и выясните, почему ее называют *тождественно истинной*.

6. Составьте таблицу истинности для формулы $\overline{x \vee y} \wedge (x \wedge \bar{y})$ и выясните, почему ее называют *тождественно ложной*.

7. Постройте таблицу истинности для формул $x \wedge y = \overline{\bar{x} \vee \bar{y}}$, $x \vee y = \overline{\bar{x} \wedge \bar{y}}$.

8. Постройте таблицу истинности для формулы $((C \vee B) \to B) \wedge (A \wedge B) \to B$.

9. Постройте таблицу истинности для формулы $A = (S \vee F) \to (\overline{S \vee F})$.

10. Докажите равносильность $x \sim y = (\bar{x} \wedge \bar{y}) \vee (x \wedge y)$.

11. Упростите выражение $(A \wedge B) \vee (A \wedge B) \vee (\bar{A} \wedge B) \vee (A \wedge B)$.

12. Упростите выражение $(\overline{x \vee y} \to (x \vee y)) \wedge y$.

13. Упростите выражение $\bar{x} \vee \overline{x \vee y} \vee \overline{y \wedge \overline{\overline{x \wedge y}}}$.

14. Упростите выражение $(x \wedge z) \vee (x \wedge ((\bar{z} \wedge \bar{y}) \vee z))$.

15. Упростите выражение $\overline{x \vee y} \wedge (x \wedge \bar{y})$.

16. Упростите выражение $\bar{x} \wedge y \vee \overline{x \vee y} \vee x$.

17. Упростите выражение $(x \vee y) \wedge (\overline{x} \vee y) \wedge (\overline{x} \vee \overline{y})$.

18. Упростите выражение $x \wedge \overline{y} \vee \overline{x} \wedge y \wedge z \vee x \wedge z$.

19. Упростите выражение $x \wedge y \vee x \wedge y \wedge z \vee x \wedge z \wedge p$.

20. Упростите выражение

$x \wedge \overline{y} \vee x \wedge y \wedge z \vee x \wedge \overline{y} \wedge z \vee x \wedge \overline{y \wedge z}$.

21. Упростите выражение: $x \wedge y \wedge (\overline{x} \wedge z \vee \overline{\overline{x \wedge y} \wedge z} \vee z \wedge t)$.

22*. Студентам объявили: «В понедельник будет одна пара занятий по бухучету и одна по математике, причем, если на первой паре математики не будет, то бухучет будет на второй паре; если третья пара не математика, то бухучет – на четвертой паре, а если математика будет на первой паре, то бухучет – на пятой». Определите, на каких парах будут бухучет и математика?

Введя обозначения x_k (математика, k-я пара), y_k (бухучет, k-я пара), строим немногословное высказывание, соответствующее условиям задачи, и выясняем условия его истинности

$S = (\overline{x_1} \rightarrow y_2) \wedge (\overline{x_3} \rightarrow y_4) \wedge (\overline{x_1} \rightarrow y_5) = ((x_1 \vee y_2) \wedge (x_3 \vee y_4)) \wedge$ $\wedge (\overline{x_1} \vee y_5) = ((x_1 \wedge x_3) \vee (x_1 \wedge y_4) \vee (y_2 \wedge x_3) \vee (y_2 \wedge y_4)) \wedge (\overline{x_1} \vee$ $\vee y_5) = ((x_1 \wedge y_4) \vee (y_2 \wedge x_3)) \wedge (\overline{x_1} \vee y_5) = (x_1 \wedge y_4 \wedge \overline{x_1}) \vee (x_1 \wedge$ $\wedge y_4 \wedge y_5) \vee (y_2 \wedge x_3 \wedge \overline{x_1}) \vee (y_2 \wedge x_3 \wedge y_5) = y_2 \wedge x_3 \wedge \overline{x_1}$.

Можно ли сделать вывод?

Глава 3. ВВЕДЕНИЕ В КОМБИНАТОРИКУ

Комбинаторный анализ – раздел математики, посвященный решению задач выбора и расположения элементов некоторого множества в соответствии с заданными правилами.

3.1. Диофантовы уравнения

Человек Древнего мира имел дело лишь с натуральными числами, еще не усвоив библейское «надо делиться», и привычные для нас операции с обыкновенными дробями (*рациональными* числами) были привилегией избранников – мыслителей, способных постичь пределы разума. Мы и сейчас число π относим к категории *иррациональных* (непостижимых для разума).

Хотя, как утверждают историки науки, уравнения первой степени умели решать древние египтяне за двадцать веков до нашей эры, современная теория чисел и алгебраическая геометрия многим обязаны греческому математику Диофанту Александрийскому* (середина III века, начало мрачной эпохи Средневековья) и так называемым *диофантовым уравнениям*, связанным с поиском их целочисленных решений.

Одной из таких задач была задача: найти четыре числа такие, чтобы квадрат суммы их, если к нему прибавить одно из них, или отнять, оставался бы квадратом.

Решение уравнений путем перебора вариантов требует существенных затрат энергии и знаний, накопленных более чем за двадцать веков. Например, столкнувшись с уравнением для алгебраи-

---

* Эпитафия (загадка)

Прах Диофанта гробница покоит; дивись ей – и камень
Мудрым искусством его скажет усопшего век.
Волей богов шестую часть жизни он прожил ребенком
И половину шестой встретил с пушком на щеках.
Только минула седьмая, с подругою он обручился.
С нею пять лет проведя, сына дождался мудрец;
Только полжизни отцовской возлюбленный сын его прожил.
Отнят он был у отца ранней могилой своей.
Дважды два года родитель оплакивал тяжкое горе,
Тут и увидел предел жизни печальной своей.

ческого многочлена $x^3 - 7x^2 - 16x + 112 = 0$, мы знаем, что оно имеет три корня (один вещественный и два комплексно сопряженных или три вещественных, различных или совпадающих). Поскольку этот многочлен представим произведением $(x - x_1)(x - x_2)(x - x_3)$, где x_1, x_2, x_3 – корни уравнения, то их произведение $x_1 x_2 x_3 = 112$. Обнаружив для 112 простые множители 7 и 2, подозреваем, что искомые корни равны ± 7, ± 2, ± 4, ± 8 или ± 16 . Подставив эти десять версий в уравнение, выбираем приемлемые (здесь 7, 4 и –4; но в общем случае удача не гарантирована).

Вроде бы все просто. Но, как писал прапорщик Л. Н. Толстой в Севастополе во время Крымской войны: «Чисто писано в бумаге, да забыли про овраги, как по ним ходить». Каковы затраты хотя бы на разложение числа на множители, если это число не 112, а некое шестнадцатизначное?

Попробуем найти неотрицательные целочисленные решения системы двух линейных уравнений с тремя неизвестными

$$4x_1 + 5x_2 + 3x_3 = 121,$$
$$3x_1 + 2x_2 + 4x_3 = 121.$$

Очевидно, что здесь $x_1 \leq 121 / 4$, $x_2 \leq 121 / 5$, $x_3 \leq 121 / 3$ и $x_1 \leq 121 / 3$, $x_2 \leq 121 / 2$, $x_3 \leq 121 / 4$, откуда $0 \leq x_1 \leq 30$, $0 \leq x_2 \leq 24$, $0 \leq x_3 \leq 30$. Переберите $31 \cdot 25 \cdot 31 = 24025$ вариантов и выберите приемлемый…

При решении задач оптимального планирования приходится иметь дело с ответами типа – произвести 16,35 самолетов; 63,7 БМП и 1276,2 велосипедов с привлечением 962,3 работников. В этом случае не всегда приемлемо округление (может оказаться, что не хватает ресурсов или имеется их существенное недоиспользование).

Кстати, есть много оптимизационных задач с решениями типа «да – нет» (0 или 1).

Многие из подобных задач решаются методами целочисленного линейного программирования [9], но еще больше существует задач с фантастически большим объемом перебора. Даже когда быстродействие компьютера исчисляется триллионами операций в секунду (флопсов), решение таких задач требует огромных временных затрат.

3.2. Возрождение математики. Простые числа и теоремы Ферма

В 1621 г. «Арифметика» Диофанта была издана во Франции и на полях этой книги оставил свои гениальные заметки и решения великий математик Пьер Ферма (1601–1665). Именно здесь зафиксировано утверждение, что каждое *простое* число вида $4n+1$ представимо суммой двух квадратов и притом только одним способом ($5 = 2^2 + 1^2$, $17 = 4^2 + 1$, $41 = 5^2 + 4^2$ и т. д.)*. Ферма доказал, что *если a не делится на простое число p, то число $a^{p-1} - 1$ делится на p* (**малая теорема Ферма**). Он предложил математическому миру интереснейшие задачи, связанные с решением уравнений в целых числах (многие из них спустя столетие решил Леонард Эйлер) и, в частности, **великую теорему Ферма** с многовековой историей: *Для любого натурального числа $n > 2$ уравнение $a^n + b^n = c^n$ не имеет натуральных решений a, b, c.*

Заметим, что интерес к простым числам возник, по крайней мере, еще во времена Евклида. Эратосфен Киренский (276–194 гг. до н. э.) предложил знаменитое *решето Эратосфена* как способ определения последовательности простых чисел.

Спустя тысячелетие европейские математики, движимые более любопытством, нежели поиском хлеба насущного, вернулись к попыткам простого поиска простых чисел. Так были выделены числа Мерсенна вида $M_p = 2^p - 1$, где p – простое число; числа Ферма вида $F_n = 2^{2^n} + 1$, где n – неотрицательное целое число (найдены для $n = 0, 1, 2, 3, 4$) и др.

Ниже мы вернемся к простым числам как к интересным помощникам в сфере криптографии.

---

* Натуральное число, не имеющее делителей кроме 1 и самого себя, называют простым. Например, 2, 3, 5, 7, 107, …, $2^{31} - 1 = 2\,147\,483\,647$ (найдено Л. Эйлером) и др.

3.3. Числа Фибоначчи и поиск экстремума функции

Последовательность натуральных чисел, определяемая в форме

$$F_0 = F_1 = 1,\ F_k = F_{k-1} + F_{k-2},\ k \geq 2,$$

называется числами Фибоначчи[**] (1, 1, 2, 3, 5, 8, 13, 21, 34, 55, 89, 144, 233, 377, 610, 987, 1597, 2584, 4181, 6765, 10946, ...). Обратите внимание на то, что двадцатое число Фибоначчи превышает 10^4, двадцать пятое превышает 10^5 и т. д.

Одной из проблематичных задач прикладной математики является задача поиска экстремумов функции $F(x)$.

Вроде бы, нет проблем. Школьнику наших дней предлагают продифференцировать эту функцию, найти корни уравнения $F'(x) = 0$, которые и определяют точки экстремума. Увы, он умеет с легкостью решать лишь линейные, квадратные и самые наипростейшие нелинейные уравнения с помощью таблиц значений элементарных функций.

С началом компьютерной эры поиск точки экстремума (например, максимума – наибольшего значения) функции $F(x)$ на некотором отрезке $[a, b]$ значительно упростился. Так, при желании найти ее с точностью порядка тысячной доли интервала можно разбить этот интервал на 1000 частей и вычислить $F(x)$ в 1001 точке.

Можно доказать, что для функции $y = f(x)$, унимодальной на отрезке $[0, F_n]$ (обладающей единственным экстремумом искомого типа), точка максимума может быть локализована в интервале единичной длины путем вычисления и взаимного сравнения не более n значений $f(x)$[*].

Отсюда следует, что для уменьшения исходного интервала в 1000 раз достаточно лишь шестнадцати вычислений значения функции ($F_{16} = 1597$).

Так, для достижения заданной точности ε находим номер n числа Фибоначчи такой, что $F_n > (b - a) / \varepsilon$.

[**] Фибоначчи (Леонардо Пизанский) (1170–1250) – первый выдающийся математик средневековой Европы, принесший в Европу с Востока десятичную систему счисления.

[*] Р. Беллман. Динамическое программирование. – М. : ИИЛ, 1960.

Возьмем в интервале точку, отстоящую от его начала на F_{n-1} / F_n доли его длины, т. е.

$$x^* = a + (b - a)\frac{F_{n-1}}{F_n},$$

и вычислим значение $f^* = f(x^*)$.

Возьмем точку $x = a + (b - x^*)$ (в сущности, точку с отступом от a на долю F_{n-2} / F_n длины интервала).

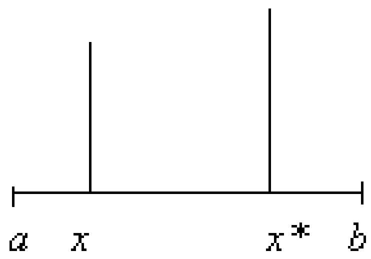


Рис. 4. Сужение интервала методом Фибоначчи

Если $f(x) < f^*$, то точка максимума при $x < x^*$ входит в интервал от x до b (заменяем a на x) и при $x > x^*$ – в интервал от a до x (заменяем b на x).

В противном случае, если $x < x^*$, берем интервал от a до x^* (заменяем b на x^* и x^* на x с запоминанием $f(x)$ в качестве f^*) и, если $x > x^*$, берем интервал от a до x (заменяем a на x^* с запоминанием $f(x)$ в качестве f^*). Случай равенства $f(x) = f^*$ может быть отнесен к любому из рассмотренных вариантов.

Эту процедуру повторяем n раз, что и гарантирует достижение заданной точности.

Так, для функции $f(x) = x\,(1 - x)$, заведомо обладающей единственным максимумом, на отрезке [0, 1] поиск точки максимума с точностью 0,0001 требует двадцати вычислений значения функции, т. к. $F_{20} = 10946$ (процесс вычислений приведен в табл. 4).

При больших значениях k отношение $F_k \big/ F_{k-1} \sim \frac{1+\sqrt{5}}{2} \approx 1{,}62$.

Это число (1,62), называемое «золотым сечением» (древние греки считали, что прямоугольник с отношением сторон $1 / 1{,}62 \sim 0{,}62$ имеет самые приятные пропорции, и использовали его в своей архитектуре), позволяет построить видоизменение рассмотренного выше метода.

Вместо метода Фибоначчи можно выбирать начальные точки на удалении $0{,}62\,(b - a)$ от концов промежутка и ранее описанный процесс продолжать до тех пор, пока интервал поиска не окажется меньше величины допустимой погрешности.

Таблица 4. Поиск экстремума методом Фибоначчи

k	***a***	***b***	***x****	***f*(*x**)**	***x***	***f*(*x*)**
1	0	1	0,61803	0,23607	0,38197	0,23607
2	0	0,61803	0,38197	0,23607	0,23067	0,18034
3	0,23607	0,61803			0,47214	0,24922
4	0,38197	0,61803	0,47214	0,24922	0,52786	0,24922
5	0,47214	0,61803	0,52786	0,24922	0,56231	0,24612
6	0,47214	0,56231			0,50658	0,24996
7	0,47214	0,52768	0,50658	0,24996	0,49342	0,24996
8	0,47214	0,50658	0,49342	0,24996	0,48529	0,24978
9	0,48529	0,50658			0,49845	0,25000
10	0,49342	0,50658	0,49845	0,25000	0,50155	0,25000
...	....	....	....	....	....	....
20	0,49991	0,50018	**0,50009**	**0,25000**	0,50000	0,25000

3.4. Магические квадраты

Предметом неиссякающего интереса математиков (и не только математиков) всего мира от Европы до Китая остаются задачи про магические квадраты и кубы.

Магический квадрат – квадратная матрица (таблица) размерности n на n, заполненная натуральными числами так, что сумма чисел в каждой строке, каждом столбце и на главных диагоналях одинакова (магическая константа).

Так, если используются целые числа от 1 до n (*нормальный магический квадрат*), его магическая константа равна $M(n) = n\,(n^2 + 1) / 2$. Например, $M(3) = 15$, $M(4) = 34$, $M(5) = 65$ и т. д.

Примерами нормального магического квадрата могут служить известный квадрат, изображенный в 1514 г. на гравюре Альбрехта Дюрера (рис. 5), и так называемый *дьявольский* квадрат (рис. 6), где магической константе равны и суммы по ломаным диагоналям (образуются при сворачивании квадрата в трубку; например см. элементы, выделенные на рис. 6).

Легко убедиться в том, что задача построения магического квадрата имеет множество решений. Существуют и многочисленные вариации требований к создаваемым квадратам и подходов к их построению.

16	3	2	13
3	10	11	8
9	6	7	12
4	15	14	1

Рис. 5. Нормальный магический квадрат А. Дюрера

7	12	1	14
2	13	8	11
16	3	10	5
9	6	15	4

Рис. 6. Дьявольский квадрат (XI век, Индия)

Н. Я. Виленкин [8], ссылаясь на знаменитого американского популяризатора математики Мартина Гарднера, упоминает некого К. У. Адамса, который занимался построением магического шестиугольника (рис. 7) с 1910 до 1962-го года (найденное им решение оказалось единственным).

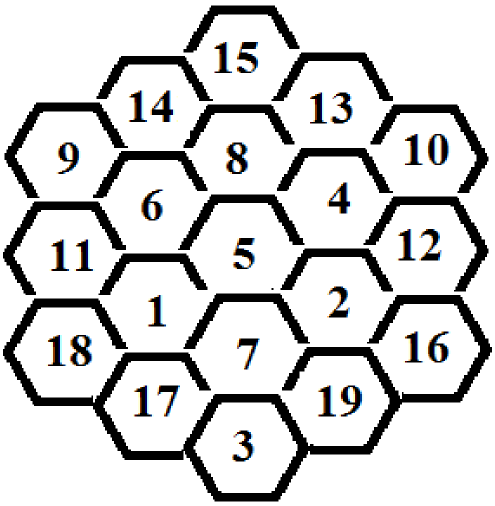


Рис. 7. Магический шестиугольник

Решение подобных задач, разумеется, едва ли напрямую способствует повышению урожайности кукурузы, но посредством математических олимпиад повышает интеллект тех, кто в будущем будет искать пути этого повышения (кому полтора столетия назад могло почудиться, что алгебра Буля станет одним из столпов современной цивилизации).

Как это не парадоксально, но становлению комбинаторики способствовала страсть человека к игре. Шахматы были забавой королей и высшей аристократии, а «третье сословие» играло в трик-трак и кости и жаждало хотя бы оценить шансы на успех (увы, когда сегодня такие оценки стали доступны, тысячи любителей «халявы» соблазняются финансовыми пирамидами и подобными им играми).

Рождение комбинаторики как значимого раздела математики в трудах Б. Паскаля и П. Ферма по теории азартных игр создало фундамент для теории вероятностей, колоссальный вклад в развитие которой внесли Я. Бернулли, Л. Эйлер, Г. В. Лейбниц (сам термин «комбинаторика» введен в математику Лейбницем в 1666 г. в «Рассуждениях о комбинаторном искусстве») [15]. Из последующих работ Э. Галуа, Ж. Лагранжа и др. в сфере комбинаторного анализа возникли фундаментальные понятия современной алгебры, такие как группа, поле, кольцо и др.

Благодаря одному из знаменитых математиков XX столетия Палу Эрдёши* (1913–1996), который «несмотря на всю широту математического творчества, питал особую слабость к дискретной математике», комбинаторика стала считаться отдельной дисциплиной на математическом Олимпе [14].

3.5. Перестановки, размещения, сочетания

Возьмем элементарнейшую задачу. В базе данных бюро обмена, содержащей n записей (что предлагают и что хотят), хотелось бы сделать выборку вариантов прямого обмена. Задача практически эквивалентна поиску в квадратной n-мерной матрице симметричных элементов $a_{ik} = a_{ki}$, $i \neq k = 1, \ldots, n$.

Очевидно, что при программировании «в лоб» (цикл в цикле) количество вызовов записей для просмотра равно $N = n\,(n - 1) / 2$. Если на один вызов ваш суперкомпьютер затрачивает 10 мкс, то при $n \sim 1000$ решение будет найдено за 50 с (остается лишь радоваться компьютерному прогрессу). Но если $n \sim 100000$, то время вашего

* Человек нелегкой судьбы, Пал Эрдёш, наряду с Альбертом Эйнштейном, Робертом Оппенгеймером, Джоном фон Нейманом, Эмми Нётер, К. Гёделем и другими выдающимися учеными XX века, в середине 1930-х из-за разгула нацизма в Европе нашел пристанище в Институте прикладных исследований в Принстоне, но в 50-е годы в период «охоты за ведьмами» в США стал персоной non grata вместе с А. Эйнштейном, Р. Оппенгеймером, известным драматургом Артуром Миллером, великим актером Чарли Чаплиным, знаменитым кинорежиссером Стенли Крамером и другими деятелями науки и культуры.

ожидания определяется как $N \sim 5 \cdot 10^9$ и составляет более 6 сут.

Тем более не приходится говорить о быстром поиске опосредованных обменов. Поэтому и уделяется особое внимание разумной организации больших хранилищ данных с точки зрения минимизации времени доступа.

Рассмотрим непересекающиеся множества A и B, содержащие соответственно m и n различных элементов. Не возникнет сомнения в том, что выбор элемента из A **или** B возможен $m + n$ способами, тогда как выбор сначала из A и затем из B (A **и** B) возможен mn способами.

Если отказаться от требования $|A \cap B| = \varnothing$, то число возможных выборов в первом случае равно $|A \cup B| = |A| + |B| - |A \cap B|$.

Аналогичная ситуация выбора из A, B или C дает $|A \cup B \cup C| =$ $= |A| + |B| + |C| - |A \cap B| - |A \cap C| - |B \cap C| + |A \cap B \cap C|$.

Что касается второго случая (сначала из A и затем из B), то он соответствует выборам из декартова произведения $A \times B$ (множество пар (a, b), где $a \in A$, $b \in B$) с мощностью $|A \times B| = mn$.

Остановимся на некоторых понятиях комбинаторного анализа.

Пусть некоторое множество состоит из n различных элементов. Комбинации этих элементов, отличающиеся лишь порядком их следования, называют перестановками без повторений.

Очевидно, что перебор вариантов сводится к выбору некоторого элемента из n, затем очередного элемента из $n - 1$ оставшихся и так далее и, соответственно, число возможных перестановок равно

$$P(n) = n \cdot (n - 1) \cdot (n - 2) \cdot \ldots \cdot 2 \cdot 1 = n!.$$

Например, число перестановок цифр 0 и 1 равно двум (01, 10), тогда как разместить 10 лидеров стран за столом переговоров можно 10! = 3 628 800 способами.

Если же во множестве из n элементов присутствуют k разновидностей (групп одинаковых элементов) численностью соответственно $n_1, n_2, \ldots, n_k$ ($n_1 + n_2 + \ldots + n_k = n$), то число различных перестановок (говорят о *перестановках с повторениями*) равно

$$P(n_1, n_2, \ldots, n_k) = \frac{n!}{n_1! n_2! \ldots n_k!}.$$

Например, при выборке из урны с 6 разноцветными шарами существует $6! = 720$ различимых последовательностей. Если же из них 3 белых и 3 черных, то число таких последовательностей равно $6! / (3! \cdot 3!) = 20$. Из букв слова АБРАКАДАБРА ($n = 11$, $n_А = 5$, $n_Б = 2$, $n_Р = 2$, $n_К = n_Д = 1$) можно построить $11! / (5!\ 2!\ 2!\ 1!\ 1!)$ различных 11-буквенных слов.

Следующим из базисных понятий комбинаторного анализа является *размещение*.

Размещениями называют комбинации k элементов, выбираемых последовательно из n-элементного множества, отличающиеся составом или порядком расположения элементов.

Если в исходном множестве все элементы различные, то говорят о *размещениях без повторений*. Количество возможных таких размещений равно

$$A(n, k) = \frac{n!}{(n-k)!}.$$

В игре «Выбери последовательность счастливой комбинации 5 из 10» число получаемых размещений

$$A(10, 5) = 10 \cdot 9 \cdot 8 \cdot 7 \cdot 6 > 30000.$$

Шанс угадать выигрывающую последовательность весьма скромный.

Изменим условия выборки и будем выбирать «образ» символа, не изымая сам символ из исходного множества. Количество возможных таких *размещений с повторениями* равно n^k.

Например, для множества из трех элементов $\{a, b, c\}$ число возможных размещений без повторений по 2 равно $A(3, 2) = 3 \cdot 2 = 6$: ab, ac, ba, bc, ca, cb. При *выборке с возвратами* число возможных размещений с повторениями по 2 составит $3^2 = 9$: aa, ab, ac, ba, bb, bc, ca, cb, cc.

Третье из базисных понятий комбинаторного анализа связано с *сочетаниями*.

Сочетаниями называют комбинации k элементов, выбираемых последовательно из n-элементного множества, *отличающиеся хотя бы одним элементом*.

При обычной выборке, когда все выбранные k элементов как бы временно последовательно изымаются из множества, число возможных таких *сочетаний без* повторений равно

$$C(n,k)=\frac{n!}{k!\,(n-k)!}.$$

Так, при организации соревнования, где каждый из 9 спортсменов должен встретиться со всеми соперниками, число встреч равно

$$C(9,2)=\frac{9!}{2!\,7!}=\frac{9\cdot 8}{2}=36.$$

При заполнении билета лотереи «5 из 10», при выборе 5 попутчиков при наличии 10 желающих и т. п. для нас не важен порядок их следования и, соответственно, число приемлемых вариантов $C(10, 5) = (10 \cdot 9 \cdot 8 \cdot 7 \cdot 6) / 5! = 252$ (можно рискнуть и сыграть!).

Величины $C(n, k)$ в математике называют *биномиальными коэффициентами*. Для них используют и другие стандартные обозначения $C_n^k=\begin{pmatrix} n \\ k \end{pmatrix}$.

Многочлен

$$(a+b)^n=C_n^0\,a^n+C_n^1\,a^{n-1}b+C_n^2\,a^{n-2}b^2+\ldots+C_n^{n-1}\,a\,b^{n-1}+C_n^n\,b^n.$$

называют **биномом Ньютона**

Кстати, легко показать, что

$$C_n^0+C_n^1+C_n^2+\ldots+C_n^{n-1}+C_n^n=2^n.$$

Это число всех возможных сочетаний из множества элементов произвольной длины.

Решая задачу оценки результата займа некоторой единичной денежной суммы на n лет под p % годовых, обнаруживаем, что по истечении срока долг составит

$$(1+x)^n=C_n^0+C_n^1\,x+C_n^2\,x^2+\ldots+C_n^{n-1}\,x^{n-1}+C_n^n\,x^n=$$

$$=1+n\,x+\frac{n\,(n-1)}{2}x^2+\frac{n\,(n-1)\,(n-2)}{6}x^3+\ldots+n\,x^{n-1}+x^n,\text{ где } x=p/100.$$

При рассрочке платежа на 5 лет при 15%-й ставке покупатель должен выплатить $1 + 5 \cdot 0{,}15 + 10 \cdot 0{,}0225 + 10 \cdot 0{,}003375 + \ldots \approx 2$ – вдвое, а при ставке 25 % – втрое большую сумму.

Общеизвестен треугольник Паскаля, позволяющий находить биномиальные коэффициенты при минимуме вычислений согласно приведенным на рис. 8 рекуррентным соотношениям.

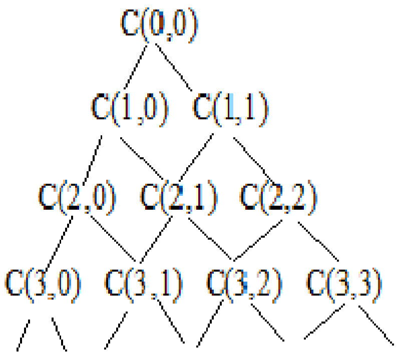
	1 1 1 1 2 1 1 3 3 1 1 4 6 4 1 1 5 10 10 5 1 …	$C(0, 0) = 1$, $C(n, r) =$ $= C(n-1, r-1) +$ $+ C(n-1, r)$, где $n > 0$, $r \le n$.

Рис. 8. Треугольник Паскаля

Следует заметить, что компьютерное вычисление $C(n, r)$ даже при не слишком больших n через отношение факториалов чревато опасностями (возможно переполнение – overflow). Так значение 22! ($21! \approx 5{,}1091 \cdot 10^{19}$) выходит за допустимый диапазон представления чисел в компьютере для определенного типа данных (например, целого типа – integer). Поэтому в этой ситуации предпочтительнее использовать последовательную вычислительную конструкцию

$$C_n^r = \left(\frac{n}{1}\right)\left(\frac{n-1}{2}\right)\left(\frac{n-2}{3}\right)\ldots\left(\frac{n-r+1}{r}\right).$$

В случае больших n в вычислениях может помочь знакомство с формулой Стирлинга (Муавра – Стирлинга)

$$n! \approx \sqrt{2\pi n}\left(\frac{n}{e}\right)^n.$$

Аппарат перестановок и сочетаний, исторически возникший благодаря азартным играм, стал фундаментом теории вероятностей, большинство из дискретных распределений которой отличаются лишь условиями выбора и состава множества и позволяют контролировать качество продукции, оценивать возможность распространения эпидемий и др.

В комбинаторике решаются не только задачи *подсчета количества* различных конфигураций из элементов конечных множеств, но и создаются методы решения такого рода задач на бесконечных множествах (особое внимание уделяется *простым* числам и их распределению; до сих пор не решена проблема Гольдбаха (1742 г.) о возможности разложения четного числа на сумму двух простых).

Практически неразделима комбинаторика с многообразием задач теории графов (раскраска графа, гамильтоновы контуры, оценка связности, бинарные деревья и др.).

3.6. Комбинаторика в криптографии

Передача сообщений всегда сопровождалась помехами, и до сих пор задача помехозащищенности информации не теряет своей актуальности. Столь же древней является проблема защиты информации от «чужого глаза».

В истории Древней Греции был случай, когда текст сообщения написали на голове обритого гонца и ждали отправки до момента отрастания волос. В «Гиперболоиде инженера Гарина» А. Н. Толстого текст был написан на спине ребенка, которому было запрещено мыться на многомесячном пути с Дальнего Востока до Москвы. Купцы, революционеры и шпионы пользовались симпатическими чернилами.

Герой рассказа «Золотой жук» Эдгара По, основателя жанра детектива, столкнулся с криптограммой пирата Кидда, где зашифровано местоположение клада. В рассказе Артура Конан-Дойла некий «мерзавец» заменял буквы на фигурки «пляшущих человечков». Здесь в ряде других детективов и в практике разведки для расшифровки использовался частотный метод, основанный на том, что в достаточно больших текстах, написанных на конкретном языке, разные буквы встречаются с разной частотой (в русском языке наиболее часто используются буквы О, Е, А, И, Т, С, Р, В, Л и с наименьшей частотой – Щ, Э, Ф, а в английском языке наиболее часто используются E (12,7 %), A (8,2 %), T (9,1 %), O (7,5 %), I (7,0 %), N (6,7 %), S (6,3 %) и т. д.).

В одной из древнеиндийских книг (34 века назад) попросту предлагалось алфавит языка разделить пополам и при шифровании производить взаимную замену символов. Так для русского языка

А	Б	В	Г	Д	Е	Ё	Ж	З	И	Й	К	Л	М	Н	О	П
Р	С	Т	У	Ф	Х	Ц	Ч	Ш	Щ	Ъ	Ы	Ь	Э	Ю	Я	

текст «ЭТО БОЛЬШОЙ СЕКРЕТ» превращался в «МВЯПСЯЬЛЗЯЪПБХЫАХВ».

Не слишком осложнило доступ «вражеской агентуры» появление телеграфного аппарата и азбуки Морзе (1844 г.).

Криптография как искусство разработки и взлома шифров неотделима от комбинаторики. Большинство ее методов базируется на операции приведения по модулю, появившейся в математике еще в трудах Евклида.

Запись означает остаток при целочисленном делении A на B. Так 17 mod 3 = 2, 25 mod 5 = 0, –7 mod 12 = 5. Для вычисления A mod B из A вычитается B до тех пор, пока не получится неотрицательное число меньшее B. Это число и будет искомым результатом.

Кстати, человек давно использует эту операцию, отсчитывая дни недели приведением порядкового номера дня по модулю 7. В странах, живущих по лунному календарю, используется приведение дней по модулю 28. Отсчет времени суток по часам связан с приведением по модулю 24 или 12.

При всех ухищрениях многовековой истории в основе большинства шифров лежит идея *аффинного шифра*

$$C(x) = (a\,x + b) \bmod n,$$

где a, b – целые числа, меньшие n – количества букв в алфавите.

Известен **шифр Цезаря** с ключом $a = 1$, $b = 3$, ориентированный на латинский алфавит с $n = 26$ буквам (циклическое смещение на 3 позиции влево).

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25
A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z
D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	A	B	C

Так букве X (позиция $x = 23$ в алфавите) соответствует буква A, поскольку ее место в шифроалфавите $C(23) = (1 \cdot 23 + 3) \bmod 26 = 0$.

Знаменитое послание Цезаря из Галлии VENI VIDI VICI (пришел, увидел, победил) превратилось в YHQL YLGL YLFL.

Соображения однозначности при кодировании и расшифровке требуют, чтобы *наибольший общий делитель* (НОД) значений a и n *был равен* 1.

Например, при кодировании (шифровании), взяв $a = 14$, $b = 0$ и $n = 21$ (общий делитель 7), мы получим одно и то же значение $C(x) = 0$ при $x = 3$, 6 и других значениях.

Процедура расшифровки (декодирования) несколько сложнее. Так из $C(x) = (a\,x + b) \bmod n = y$ следует $(a\,x) \bmod n = (y - b) \bmod n$. Возникает необходимость найти для a *обратное преобразование* a^* такое, что $a^*\,a\,x \bmod n = a^*\,(y - b) \bmod n$, откуда

$$x = a^*\,(y - b) \bmod n.$$

Это эквивалентно поиску $a^* < n$ такого, что $a^*\,a \bmod n = 1$.

Если взять $C(x) = (2\,x + 3) \bmod 26$, то числа $a^* < 26$ такого, что $2\,a^* \bmod 26 = 1$, найти невозможно. В самом деле, $2 \cdot 1$, $2 \cdot 2$, $2 \cdot 3$, …., $2 \cdot 25$, приведенные по модулю 26, не равны 1.

Если взять $C(x) = (3\,x + 2) \bmod 26$, где НОД(3, 26) = 1, то $3\,a^* \bmod 26 = 1$ достигается при $a^* = 9$.

Следовательно, для декодирования $x = a^*\,(y - 2) \bmod 26$ будет однозначное соответствие:

$y = 0$ $\quad x = 9\,(0 - 2) \bmod 26 = -18 \bmod 26 = 8;$
$y = 1$ $\quad x = 9\,(1 - 2) \bmod 26 = -9 \bmod 26 = 17;$
$y = 2$ $\quad x = 9\,(2 - 2) \bmod 26 = 0 \bmod 26 = 0;$
$y = 3$ $\quad x = 9\,(3 - 2) \bmod 26 = 9 \bmod 26 = 9;$
$y = 4$ $\quad x = 9\,(4 - 2) \bmod 26 = 18 \bmod 26 = 18;$
$y = 5$ $\quad x = 9\,(5 - 2) \bmod 26 = 27 \bmod 26 = 1;$
…
$y = 24$ $\quad x = 9\,(24 - 2) \bmod 26 = 198 \bmod 26 =22;$
$y = 25$ $\quad x = 9\,(25 - 2) \bmod 26 = 207 \bmod 26 = 25.$

Если шифр Цезаря (циклический сдвиг на фиксированное число позиций) доступен криптоаналитику из-за его популярности, то аффинный шифр посложнее расшифровать, хотя множество ключей a, b, n ограничено требованиями a, $b < n$ и НОД(a, n) = 1.

Конечно, если снять ограничение на порядок букв исходного латинского алфавита, то даже при известном ключе число вариантов кодирования возрастет до 26! = 403 сентиллионов ($403 \cdot 10^{30}$).

С возникновением компьютера вынужденно усложнился аппарат шифрования. Так, вместо $f(x) = (a\,x + b) \bmod n$ в алгоритме Диффи – Хеллмана (1976 г.) использована функция $f(x) = a^x \bmod p$, где p – простое число.

Тем не менее, при потрясающих возможностях современных компьютерных технологий всегда возникает опасность хищения ключа. Соответственно, возникает идея алгоритма RSA (аббревиатура от фамилий Rivest, Shamir и Adleman), в основу которого положена **малая теорема Ферма**: если a не делится на простое число p, то число $a^{p-1} - 1$ делится на p.

Берем $n = p\,q$, где p и q – простые числа. Найдем значение функции Эйлера $\varphi(n) = (p - 1)\,(q - 1)$. Примем за a любое число, для которого наибольший общий делитель НОД$(a, \varphi(n)) = 1$.

Пара (a, n) выступает как *открытый ключ* системы (значения p и q должны храниться в секрете).

Для возможности обратного преобразования находим величину d такую, что $(a\,d) \bmod \varphi(n) = 1$, которая выступает как закрытый ключ системы.

Взяв числовой код m символа исходного сообщения, шифруем его, получая $M = m^a \bmod n$. Для расшифровки достаточно выполнить обратное преобразование

$$m = M^d \bmod n \equiv (m^a)^d \bmod n.$$

Например, для $p = 3$ и $q = 11$ находим $n = 33$ и $\varphi(33) = (3 - 1)\,(11 - 1) = 20$. Выбираем $a = 7$, не имеющее общих делителей с 20. Получен открытый ключ (33, 7).

Ищем d такое, что $(7\,d) \bmod 20 = 1$ и обнаруживаем $d = 3$.

Теперь для передачи сообщения «9» достаточно найти значение $9^7 = 4782969$, привести его по модулю 33 и, получив код 15, отправить его адресату. Получатель находит $15^3 = 3375 \bmod 33 = 9$.

Итак, что же здесь сложного?

Зная n, но не зная p и q, невозможно найти значение функции Эйлера, и при этом надо суметь разложить n на простые множители, что для больших n может потребовать сверхбольших затрат.

Полагаю, что читатель имел дело с возмущенным заявлением компьютера типа overflow (переполнение – выход результата за пределы возможности представления чисел данного типа в компьютере). Более того, многозначное целое число невозможно отобразить в разрядной сетке «ячейки памяти» без потери значащих цифр.

Работа с RSA требует не только высокой квалификации криптоаналитиков, но и существенной технической поддержки затрат. Попытки взлома системы чреваты еще большими сложностями вплоть до создания специальных суперкомпьютеров.

Блестящим криптоаналитиком был знаменитый математик, отец теории алгоритмов (машина Тьюринга, 1936 г.), основатель теории искусственного интеллекта Алан Тьюринг (1912–1954), в годы войны занимавшийся разоблачением тайн германской шифровальной машины ЭНИГМА. Необходимость разоблачения ЭНИГМЫ ради спасения жизни моряков северных и трансатлантических конвоев от подводных лодок, защиты тысяч мирных граждан от немецких бомбардировщиков и ФАУ послужила значимым толчком к созданию универсальных ЭВМ.

И сегодня тысячи специалистов занимаются вопросами помехозащищенности при передаче информации, ее секретности и, увы, взлома ее хранилищ.

3.7. Упражнения

1. Сколькими способами можно рассадить класс, если присутствует 24 человека, а мест 26?
2. Сколькими способами можно вытянуть 5 карт бубновой масти из колоды, содержащей 36 карт?
3. Сколько трехзначных чисел можно составить из цифр 1, 2, 3 так, чтобы цифры в записи числа не повторялись?
4. В кухне 5 лампочек с отдельными выключателями. Сколько существует способов освещения?
5. Сколькими способами можно расставить на полке 12 книг, включающих 4 одинаковых учебника по математике, 6 одинаковых учебников по информатике, 2 одинаковых учебника по химии?

6. В булочной продается 10 различных видов пончиков. Сколькими способами можно выбрать 12 пончиков?

7. Сколько прямых линий можно провести через 7 точек, из которых лишь 3 лежат на одной прямой?

8. На выпускном вечере 20 студентов группы попарно обменялись своими фотографиями. Сколько всего потребовалось сделать фотографий?

9. Сколькими способами в пассажирский поезд, состоящий из 9 вагонов, можно продать четырем пассажирам билеты в разные вагоны и сколько билетов можно продать без этого ограничения?

10. Сколькими способами можно обить 6 различных стульев, если имеется 12 сортов обивочного материала?

11. Сколько слов (включая лишенных смысла) можно составить из всех букв слова «Миссисипи»?

12. Найдите число возможных вариантов выхода в полуфинал первенства по шахматам трех из 20 участников.

13. Сколько существует способов вытащить из колоды, содержащей 52 карты, 13 карт, из которых 9 карт будут одной масти?

14. Из колоды, содержащей 52 карты, вынули 10 карт. В скольких случаях среди этих карт окажется хотя бы один туз? В скольких случаях ровно один туз? Ровно два туза?

15. Автомобильные номера состоят из трех букв, за которыми идут 4 цифры, например «МКМ-07-37». Сколько машин можно снабдить различными номерами, если для этой цели используются 25 букв?

16. Сколько чисел больших 100 можно записать с помощью цифр 1, 2, 3, 4, если цифры в числе не повторяются?

17. Из 20 сотрудников лаборатории 5 человек должны выехать в командировку. Сколько может быть различных составов отъезжающей группы, если заведующий лабораторией и два ведущих инженера одновременно уезжать не должны?

18. Сколькими способами можно рассадить по жребию восемь рыцарей за круглым столом, чтобы первый и второй рыцари сидели рядом?

19. Двое друзей, A и B, стоят в очереди из 8 человек. Сколько существует вариантов очередей, в которых между A и B стоят два человека?

20. Сколькими способами можно сформировать железнодорожный состав из 9 вагонов так, чтобы второй и четвертый вагоны шли через один?

21. Сколькими способами можно рассадить вокруг круглого стола 6 мальчиков и 6 девочек, если каждая девочка должна сидеть между двумя мальчиками?

22. Сколькими способами можно рассадить случайным образом 12 студентов на 12 первых местах одного партера, чтобы студенты A и B сидели рядом?

23. Сколькими способами 7 человек могут встать в очередь так, чтобы два определенных лица не стояли рядом?

24. Две команды, в каждой из которых по 5 спортсменов, строятся в одну шеренгу. Сколькими способами можно построить шеренгу, чтобы игроки одной команды не стояли рядом?

25. Сколькими способами могут быть размещены дни рождения 12 человек в году, при условии, что в нем 365 дней? Во скольких случаях все дни рождения попадут на разные дни года, а во скольких на разные месяцы?

26. Решите уравнение $\dfrac{(n+2)!}{n!} = 72$.

27. Решите уравнение $\dfrac{A_{x+1}^{4}\, P_{x-4}}{P_{x-1}} = 15$.

28. Решите уравнение $C_{x+1}^{x-1} = 21,\ x \in N$.

29. Решите уравнение $C_{2n}^{n+1} : C_{2n+1}^{n-1} = \dfrac{7}{13}$.

30. Решите уравнение $A_{x}^{5} = 18\, A_{x-2}^{4}$.

Глава 4. ТЕОРИЯ ГРАФОВ

Как наглядно показать с позиций генетики вред браков между родственниками, что было характерно для многих европейских монархий?

Как наилучшим образом дать наглядное представление о системе валентных связей в молекуле некоторого химического соединения?

Как помочь Янки из Коннектикута, известному благодаря популярному среди читающих молодых людей произведению Марка Твена, если он надумает баллотироваться в президенты США и решит проехать кратчайшим путем через все столицы штатов?

Сколько ферзей можно расставить на шахматной доске, чтобы они не находились под ударом? А скольких ферзей достаточно, чтобы они не били друг друга, но держали под ударом все остальные поля?

Сколько камер наблюдения требуется установить на перекрестках городских улиц, чтобы все улицы оказались под присмотром?

Можно ли начертить заданную фигуру, не отрывая пера от бумаги и не проводя линий дважды? Может ли «мышь в лабиринте» добраться до кусочка сала?

Можно ли смонтировать электрическую схему так, чтобы провода на ней не пересекались за пределами клемм?

Сколько красок достаточно для раскраски карты Российской Федерации при условии, что соседствующие с ней субъекты нельзя окрашивать одинаково?

Сколько железнодорожных узлов требуется поразить, чтобы исключить возможность железнодорожного сообщения между двумя данными пунктами?

На каких из 2516 работ строительного проекта, в первую очередь, должен сосредоточить свое внимание его руководитель?

Как построить кратчайшую транспортную сеть, связывающую между собой все населенные пункты Н-ского района?

Множество подобных задач и составляет предмет внимания так называемой теории графов (**graph theory**), которая зародилась в ходе решения разных головоломок в XVIII веке и долго, как и вся дискретная математика, была «Золушкой серьезного математического мира» [8], хотя у ее истоков стоял великий Леонард Эйлер (1707–1783) и решением вышеназванных задач занимались многие незаурядные математики.

Формально такое название самостоятельной математической дисциплины было дано в 1936 г. в работе венгерского математика Денеша Кёнига (1884–1944).

Хотя *конструктивные* алгоритмы теории графов (построение эйлеровых циклов, путешествия по лабиринтам и пр.) не слишком многочисленны, она позволяет дать наглядное, удобное для человеческого восприятия представление о задаче, дать четкую постановку задачи, выяснить условия существования ее решения.

В последующие годы, особенно в связи с колоссальным увеличением вычислительных возможностей компьютерной техники, теория графов стала рабочим инструментом [1] теории конечных автоматов, теории информации, разработки систем связи, теории кооперативных игр, математической лингвистики, генной инженерии, военного планирования, аппарата сетевого планирования и управления сложными техническими проектами и т. д.

Большинство серьезных публикаций с изложением теории графов, среди которых заслуживают особого внимания книги [4 – 5], предъявляет достаточно высокие требования к уровню математической подготовки читателя.

Особого же упоминания заслуживает первая на русском языке книга французского математика К. Бержа [1], настольная книга для нескольких поколений советских математиков-прикладников, ставшая за более полувека после публикации библиографической редкостью. Этот труд представляет собой изумительное сочетание содержательности излагаемой тематики, изящества изложения и математической культуры с уважительным отношением к заинтересованному читателю с весьма скромными математическими познаниями.

Ниже мы остановимся лишь на азбучных понятиях теории графов, не задаваясь доказательством большинства положений и более ориентируясь на интерес и природные способности читателя.

4.1. Ориентированные графы. Основные понятия

В популярной литературе графом часто называют любой рисунок на плоскости, состоящий из точек (кружочков, прямоугольников), соединенных линиями или стрелками, забывая о том, что теория графов является частью математики и использует ее терминологию.

4.1.1. Граф как способ представления множества

В основе классического определения графа [1] лежит следующее заявление: говорят, что *дан граф* $G = (X, \Gamma)$, *если заданы непустое множество* X *и отображение* Γ *множества* X *в себя*.

Как известно, любое множество может быть задано перечислением его элементов подобно $X = \{a, b, x, c\}$ (множество, изображенное на рис. 9 *а*, состоит из четырех указанных элементов) или определением свойств элементов $X = \{x \,/\, f(x) = 0\}$ (множество X состоит из корней уравнения $f(x) = 0$).

Всякое отображение Γ множества X во множество Y есть *закон*, по которому каждому элементу $x \in X$ ставится в соответствие некоторое подмножество $\Gamma x \subset Y$, причем $\Gamma x \neq \varnothing$. Для указанного выше отображения характерно лишь то, что оно не выводит за пределы исходного множества, то есть для всех $x \in X$ выполняется $\Gamma x \in X$.

В отображении $\Gamma x = y$ элемент y называют *образом* x, а x – *прообразом* y.

Множество $\Gamma A = \bigcup\limits_{x \in A} \Gamma x$ выступает *образом* множества $A \subset X$, а A – *прообразом* для ΓA.

Например, если X – множество людей и Γx – множество детей конкретного человека $x \in X$, то $\Gamma(\Gamma x) \equiv \Gamma^2 x$ – множество внуков x, $\Gamma^{-1} x$ – множество его родителей, $\Gamma^{-2} x$ – множество бабушек и дедушек x. Обратите внимание, что здесь имеет место отображение X в себя, не выводящее за пределы человеческого общества.

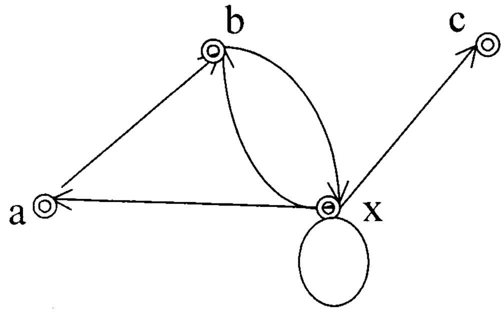
Граф (X, Г) можно задавать перечислением его компонент, например

$$X = \{a, b, x, c\}, \Gamma a = b, \Gamma b = x, \Gamma x = \{a, b, x, c\}, \Gamma c = \varnothing;$$

рисунком (рис. 9, *а*), где элементы множества изображены точками (*вершинами*) и отображения – стрелками (*дугами*), или совокупностью (X, U) множества вершин и множества дуг

$$X = \{a, b, c, x\}, U = \{(a, b); (b, x); (x, a); (x, b); (x, x); (x, c)\}.$$

Для компьютерной обработки графы можно задавать в матричном представлении, например матрицей смежности (рис. 9, *b*), списками дуг или другими компактными таблицами (рис. 9, *c*).



a

	a	*b*	*x*	*c*
a	0	1	0	0
b	0	0	1	0
x	1	1	1	1
c	0	0	0	0

b

a	*b*
b	*x*
x	*a*
x	*b*
x	*x*
x	*c*

c

Рис. 9. Способы задания графа

Можно привести множество примеров использования графов.

Одним из образцов графов служат генеалогические деревья, в вершинах которых размещены данные о человеке, а стрелки определяют родственные отношения.

На циферблате наручных механических часов изображены 12 вершин (состояний времени суток), а смещение часовых стрелок задает порядок смены состояний.

Человеческий мозг представляет собой *нейронную сеть*, состоящую из «простейших» связанных между собой элементов (нейронов), передающих сигнальную информацию.

Карта города с ее улицами (не всегда прямыми) и перекрестками выступает типичным образцом графа.

Заметим, что в графе могут присутствовать так называемые *изолированные вершины*, не имеющие ни образов, ни прообразов. Для функционирования системы, описываемой таким графом, они не представляют интереса и могут возникать в результате какой-то трансформации первоначального графа (своеобразная заброшенная деревня на карте, к которой ныне нет дорог).

Две *вершины* называют *смежными*, если они различны и существует дуга, идущая из одной вершины к другой.

Две *дуги* называют *смежными*, если они различны и имеют общую граничную точку.

Множество дуг, входящих в вершину или исходящих их нее, называют *инцидентным* этой вершине, а их количество – *степенью вершины*.

Путем в графе называют последовательность дуг (u_1, u_2, ...), в которой начало каждой очередной дуги совпадает с концом предыдущей.

Следует иметь в виду, что графы являются предметом внимания *топологии*, где нет понятия метрики и допустима любая деформация (растяжение или сжатие без разрывов). Так, с точки зрения топологии, Венера Милосская, бейсбольный мяч и теннисный мячик имеют одну и ту же «сферическую» поверхность, равно как бублик и чайная чашка с ручкой тождественны по форме тору (тороиду).

Соответственно, под *длиной пути* понимается число составляющих его дуг.

Если в пути *никакая дуга не встречается дважды*, его называют *простым*, а в противном случае – *составным*.

Путь можно задавать и перечислением его вершин

$$\mu\ [x_1\ x_n] = [x_1\ x_2\ ...\ x_{n-1}\ x_n].$$

Путь, в котором *никакая вершина не встречается дважды*, называется *элементарным*. Кстати, такой путь, проходящий через *все* вершины графа, называется *гамильтоновым* (рис. 10, *а*).

Путь может быть *конечным* или *бесконечным*. Конечный путь, в котором начальная и конечная вершины совпадают, называют *контуром*. Контур единичной длины называют *петлей*.

Граф (X, U) называют *симметрическим* (рис. 10, *b*), если $(x, y) \in U \Rightarrow (y, x) \in U$ – две любые смежные вершины соединены в обоих направлениях, и *антисимметрическим*, если $(x, y) \in U \Rightarrow (y, x) \notin U$ – две смежные вершины соединены только в одном направлении.

Граф (X, U) называют *полным* (рис. 10, *c*), если $(x, y) \notin U \Rightarrow (y, x) \in U$ – любые две вершины соединены хотя бы в одном направлении.

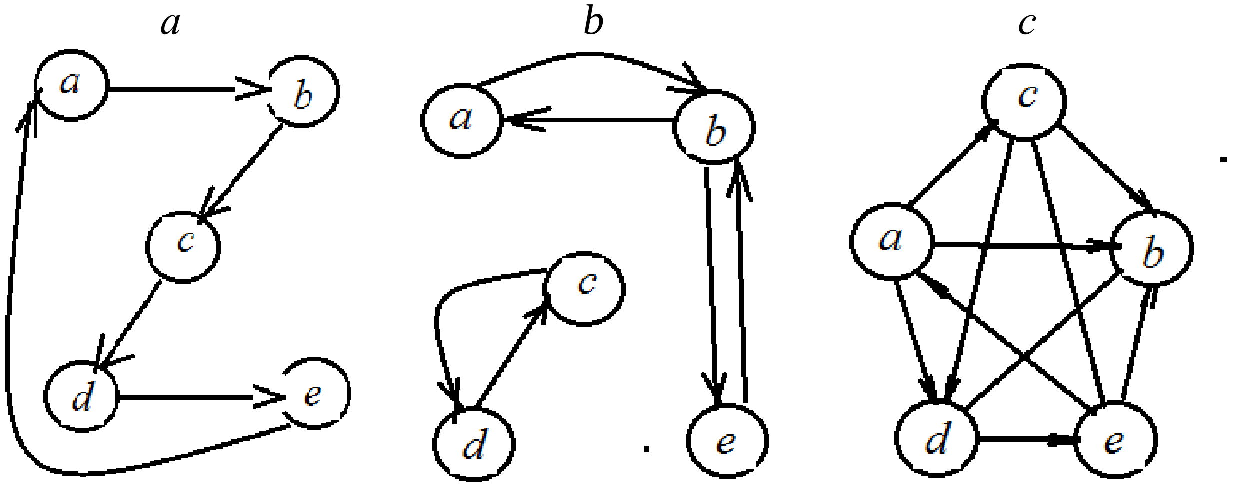


Рис. 10. Графы
(*a – гамильтонов контур*; *b – симметрический граф*; *c – полный граф*)

Граф (X, U) называют *сильно связным*, если для любых двух различных его вершин x и y существует путь из x в y.

4.1.2. Представление графа для компьютерной обработки

Едва ли представление графа для последующего компьютерного анализа в виде сканированного рисунка рационально не только по технологическим причинам, но и с точки зрения простоты программной обработки. К тому же не следует забывать, что не всякий граф удается разместить на листе бумаги.

Как уже указывалось выше, для компьютерной обработки графы можно задавать в матричном представлении, например, матрицей смежности (рис. 9, *b*), списками дуг (рис. 9, *c*) или другими компактными таблицами.

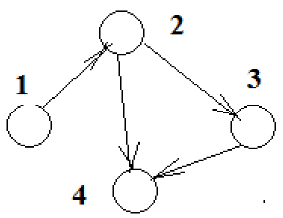
Как правило, объем списка дуг много меньше числа ненулевых элементов матрицы смежности.

Однако если нет существенных лимитов на емкость памяти или время работы компьютера, то программная реализация многих задач при представлении графа матрицей смежности не требует искусства.

Так, если матрицу смежности обозначить как $S = \{s_{ij},\ i, j = \overline{1, n}\}$ и элемент $s_{ij} = 1$ понимать как возможность перехода из вершины i в вершину j одним шагом, то ненулевые элементы матрицы $S^2 = S \cdot S$ определяют возможность перехода за 2 шага, S^3 – за 3 шага и т. д.*

Если в графе отсутствуют контуры, то матрица S^{k+1} обращается в нулевую (k – длина самого «длинного» пути).

Например, в нижеприведенной ситуации (рис. 11) мы видим, что из вершины 1 можно добраться до вершины 4 за 2 или 3 шага и путей длиной 4 и более не существует.



$$S = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad S^2 = \begin{pmatrix} 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix} \quad S^3 = \begin{pmatrix} 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$$

Рис. 11. Граф с возможностью перехода из 1 в 4 за 2 или 3 шага

Если же $S^{n+1} \neq 0$ (n – число вершин), то в графе присутствует хотя бы один контур. Так, в примере на рис. 12 существует контур, проходящий через вершину 2.

4.1.3. Усечения графа

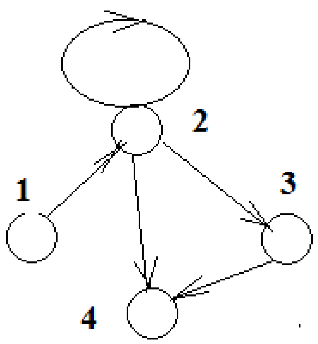
Остановимся на понятиях, связанных с удалением некоторых

---

* Мы полагаем, что читателю знакомы правила умножения матриц и он в состоянии запрограммировать эту процедуру или воспользоваться ее библиотечной реализацией в любой среде программирования.

компонент графа.

Подграфом графа (X, Г) называется граф (A, ГA), где $A \subset X$ и Г$A\,x$ = Г$x \bigcap A$ – получается выбором части вершин исходного графа с сохранением всех присутствующих связей между этими вершинами.



$$S = \begin{array}{|c|c|c|c|}\hline 0&1&0&1\\\hline 0&1&0&1\\\hline 0&1&0&0\\\hline 0&0&1&0\\\hline\end{array}\quad S^2 = \begin{array}{|c|c|c|c|}\hline 0&1&1&1\\\hline 0&1&1&1\\\hline 0&1&0&1\\\hline 0&1&0&0\\\hline\end{array}\quad S^3 = \begin{array}{|c|c|c|c|}\hline 0&2&1&1\\\hline 0&2&1&1\\\hline 0&1&1&1\\\hline 0&1&0&1\\\hline\end{array}$$

$$S^4 = \begin{array}{|c|c|c|c|}\hline 0&3&1&2\\\hline 0&3&1&2\\\hline 0&2&1&1\\\hline 0&1&1&1\\\hline\end{array}\quad S^5 = \begin{array}{|c|c|c|c|}\hline 0&4&2&3\\\hline 0&4&2&3\\\hline 0&3&1&2\\\hline 0&2&1&1\\\hline\end{array}\quad \ldots$$

Рис. 12. Граф с контуром, проходящим через вершину 2

Частичным графом (X, Г) называется граф (A, Δ), где $\Delta x \subset \Gamma x$ при всех $x \in X$, который получается выбором всех вершин исходного графа с потерей некоторых связей между ними.

Частичным подграфом (X, Г) называется граф (A, ΔA), где $A \subset X$ и $\Delta A\ x \subset \Gamma x \bigcap A$, который получается выбором части вершин исходного графа с сохранением лишь части присутствующих связей между этими вершинами.

Например, для графа – карты железных дорог России – карта железных дорог внутри Кемеровской области выступает в роли подграфа. Если же на некоторый период закрыть движение по отдельным маршрутам, то от исходной карты (графа) остается частичный граф (если на ней сохранены все железнодорожные станции) или частичный подграф (если часть станций ликвидирована).

Рассмотренный выше граф, предполагающий полную ориентацию связей между некоторыми объектами (состояниями) и тем самым отражающий причинно-следственные связи в изучаемой системе, в современной литературе предпочитают называть *орграфом* или *диграфом* [4].

4.2. Граф при отказе от ориентированности

Существует и другая система понятий о графе, в которой отказываются от учета ориентации. В отличие от *орграфов*, здесь базовым понятием является не *дуга*, а *ребро* – пара элементов множества, связанных хотя бы в одном направлении. Соответственно, понятие *пути* заменяется понятием *цепи* как последовательности ребер, а *контур* заменяется *циклом*. Теряет смысл и понятие петли.

Следует заметить, что в литературе (особенно последних лет) без надлежащих оговорок рассматривают неориентированные графы, смешивая понятия дуги и ребра, пути и цепи, контура и цикла. При таком подходе в транспортных сетях исчезает одностороннее движение, в сетевых графиках (есть прикладная математическая дисциплина, называемая *сетевым планированием*) не всегда можно понять порядок следования работ, при описании функционирования систем из-за отсутствия понятия петли невозможно описать факт сохранения предыдущего состояния системы.

4.3. Компоненты связности и множества сочленения

При отказе от требования ориентированности приходится несколько расширить понятие связности и объявлять *граф связным* при возможности *перехода* между любой парой вершин *по цепи* (очевидно, что сильно связный граф является связным, но не наоборот).

Очевидно, что граф может состоять из нескольких *компонент связности*. Примером многокомпонентности может служить железнодорожная сеть мира (хотя есть кинорежиссеры, в фильмах которых железнодорожные эшелоны идут из Москвы до Магадана).

Для характеристики связного графа по устойчивости к разрушению, наряду с упомянутыми выше подграфами и частичными графами, используются некоторые специальные термины.

Вершина связного графа выступает как *точка сочленения (шарнир)*, если подграф, получаемый при ее удалении, несвязен. Если к тому же эта точка связана с каждой из компонент связности подграфа только одним ребром, ее называют *точкой простого сочленения*.

В тайной организации связные – точки сочленения, в армейской части таковыми являются командиры подразделений. В сети железных дорог страны Хабаровск – точка сочленения.

Кстати, граф, обладающий гамильтоновым циклом, не имеет точек сочленения.

Непустое множество вершин, после удаления которого связный граф перестает быть таковым, называется *множеством сочленения*. Размерность наименьшего такого множества называется *числом связности*.

По отношению к точке (множеству) сочленения среди остальных вершин выделяют наименее и наиболее удаленные (*центроиды* и *периферoccиды*).

Существует много примеров необходимости построения графов повышенной связности или выявления минимального множества сочленения. Например, поиск минимального числа железнодорожных узлов, после разрушения которых в результате какой-то катастрофы железнодорожная сеть распадется на несколько несвязных образований.

4.4. Деревья

Связный граф, не содержащий циклов, называют *деревом*, а множество деревьев – *лесом* (дерево с n вершинами имеет $n-1$ ребро, и, как утверждает формула Кэли, для каждого n можно построить n^{n-2} деревьев). Часть вариантов при $n = 6$ приведена на рис. 13, *а*.

В приложениях, связанных с выбором из двух альтернатив (*да* или *нет*), проявляется интерес к так называемым *бинарным деревьям*. Подобные *деревья решений* широко используются в интеллектуальном анализе данных в попытках создать модель предсказания значения на выходе на основе нескольких данных на входе.

В бинарном дереве выделена *корневая вершина* и каждая вершина инцидентна либо одному, либо трем ребрам (степень вершины равна 1 или 3). Вершины степени 3 и вершина, следующая за корнем, называются *внутренними вершинами*. Вершины степени 1 называют *листьями* дерева.

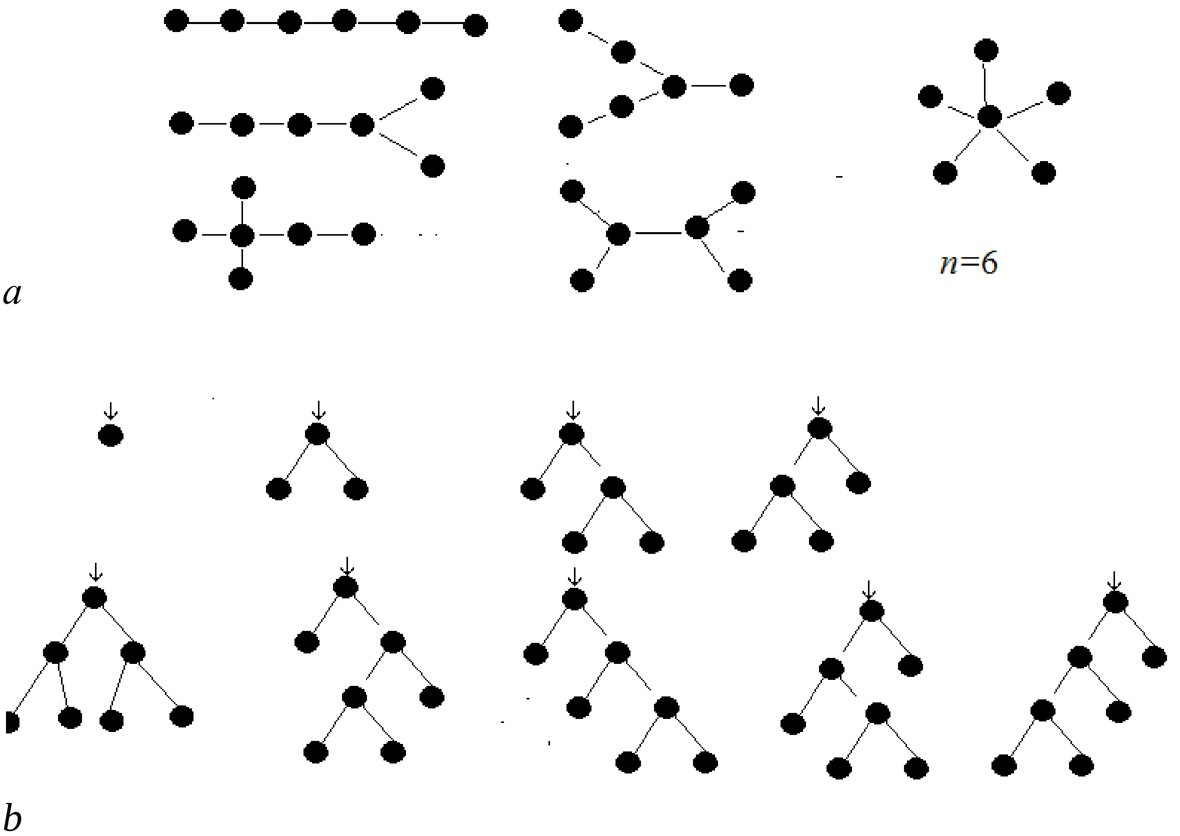


Рис. 13. Примеры деревьев

Сколько же существует бинарных деревьев с одинаковым числом внутренних вершин m? На рис. 13, b изображены бинарные деревья для $m = 0$, 1, 2 и 3 (соответственно 1, 1, 2 и 5).

Поскольку такое дерево с m внутренними вершинами можно разложить на два дерева с k_1 и k_2 внутренними вершинами ($k_1 + k_2 = m – 1$), то можно доказать, что искомое число

$$F_m = F_0\, F_{m\text{-}1} + F_1\, F_{m\text{-}2} + F_1\, F_{m\text{-}2} + \ldots + F_{m\text{-}2}\, F_1 + F_{m\text{-}1}\, F_0,$$

$$F_0 = F_1 = 1.$$

Это рекуррентное соотношение дает внушительные оценки численности бинарных деревьев: 14 ($m = 4$), 42 ($m = 5$), 132 ($m = 6$), 429 ($m = 7$), …, 2674440 ($m = 14$) [14].

4.5. Сумма и произведение графов

Выше мы определяли $A \times B$ как декартово произведение множеств: множество пар (a, b), где $a \in A$, $b \in B$.

Что касается вышеуказанных операций (сумма и произведение) над графами, то они введены для изучения эффекта *совокупного действия нескольких систем*, каждая из которых представима в форме графа.

Произведением (*пересечением*) графов $G_i = (X_i, \Gamma_i)$ называется граф $G = (X, \Gamma)$, в котором

$$X = X_1 \times X_2 \times ... \times X_n = \{(x_1\, x_2 \,...\, x_n) \,/\, x_i \in X_i\},$$

$$\Gamma(x_1\, x_2 \,...\, x_n) = \Gamma_1 x_1 \times \Gamma_2 x_2 \times ... \times \Gamma_n x_n$$

(все подсистемы одновременно изменяют свое состояние).

Суммой (*объединением*) графов $G_i = (X_i, \Gamma_i)$ называется граф $G = (X, \Gamma)$, в котором

$$X = X_1 \times X_2 \times ... \times X_n = \{(x_1\, x_2 \,...\, x_n) \,/\, x_i \in X_i\}$$

и

$\Gamma(x_1\, x_2 \,...\, x_n) = [\Gamma_1 x_1 \times \{x_2\} \times ... \times \{x_n\}] \cup [\{x_1\} \times \Gamma_2 x_2 \times ... \times \{x_n\}] \cup \cup ... \cup [\{x_1\} \times \{x_2\} \times ... \times \Gamma_n x_n]$ (только одна из подсистем изменяет свое состояние).

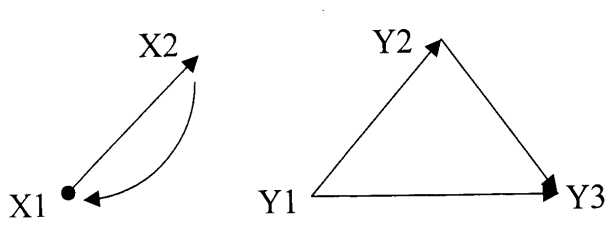


Рис. 14. Примеры графов

Так, для графов, представленных на рис. 14, их произведение и сумма имеют вид, показанный на рис. 15, 16.

Понятия суммы и произведения графов используются как при описании и анализе технических систем, так и при изучении ситуаций, исследуемых теорией игр и статистических решений.

4.6. Порядок на графах

Еще Пифагор заявлял, что «порядок – прекраснейшее украшение жилища». Но, даже отвлекаясь от соображений эстетики, мы знаем, что заливка фундамента предшествует установке оборудования, а в песне «Мы с чудесным конем все поля обойдем, соберем, и посеем, и вспашем» несколько нарушен порядок сельхозработ.

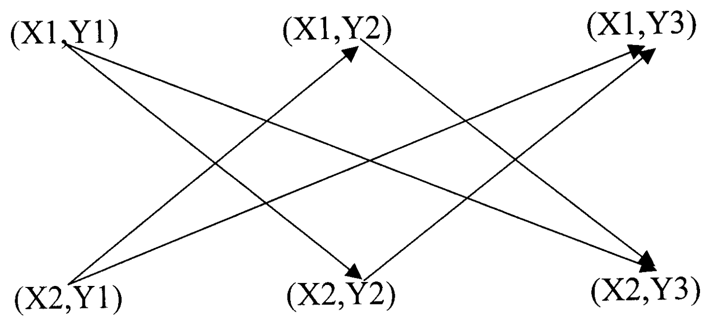


Рис. 15. Произведение графов для рис. 14

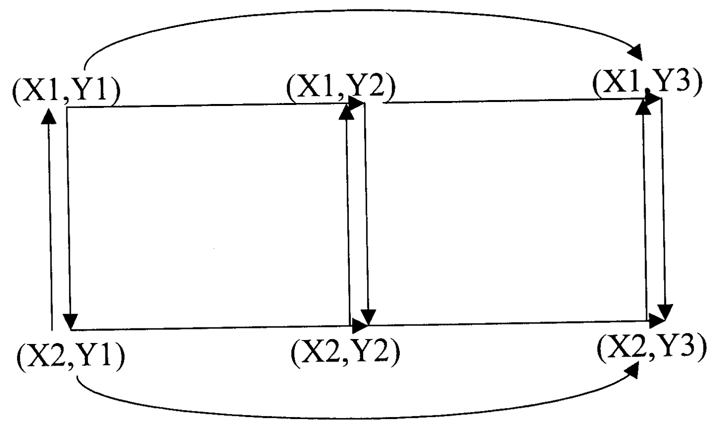


Рис. 16. Сумма графов для рис. 14

При решении задач сетевого планирования, создании иерархических, реляционных или сетевых баз данных, проектировании систем передачи сигналов и т. д. приходится устанавливать тот или иной порядок на соответствующих графах. К тому же многие положения различных экономических теорий имеют интерпретацию в виде графов: так отношения предпочтения $x \prec y$ между потребительскими наборами в теории спроса могут интерпретироваться как наличие пути из x в y, что в теории графов называется *отношением квазипорядка*.

4.6.1. Порядковая функция

Остановимся на понятии *порядковой функции* и условиях ее существования, обозначая через $|A|$ число элементов множества A.

Как уже указывалось, граф (X, Γ) называется *конечным*, если $|X| < \infty$.

Для конечных графов условия существования порядковой функции достаточно просты (недопустимо наличие контуров), для бесконечных нужен учет количества образов, прообразов и др.

Граф (X, Γ) называется Γ-*конечным*, если $|\Gamma x| < \infty$ для всех $x \in X$, и Γ^{-1}-*конечным*, если $|\Gamma^{-1} x| < \infty$ для всех $x \in X$. Другими словами, для Γ и Γ^{-1}-конечности необходимо, чтобы каждая из его вершин имела конечное число образов и прообразов. Граф, обладающий Γ и Γ^{-1}-конечностью, называется *локально конечным*.

Например, приведенный на рис. 17 бесконечный граф Γ конечен, но не Γ^{-1}-конечен.

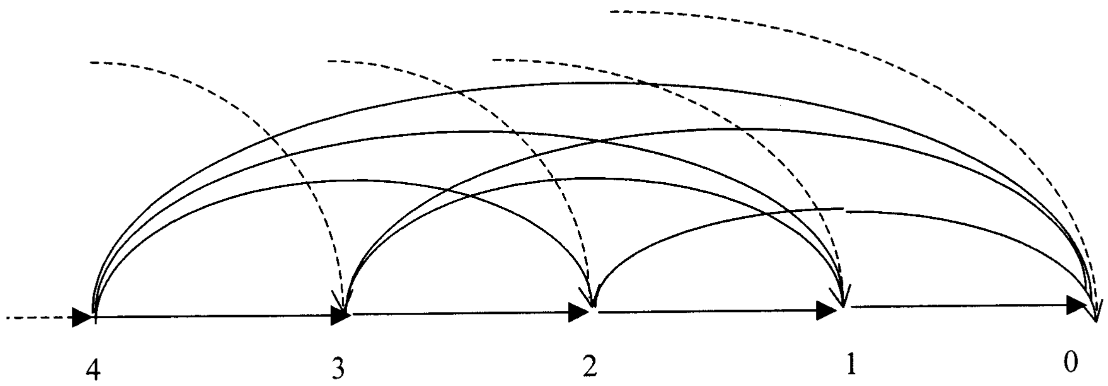


Рис. 17. Бесконечный граф

Доказано (**теорема Радо**), что в случае локально конечного графа существует целозначная порядковая функция $\varphi(x)$ такая, что $\varphi(x) < \varphi(y)$ для всех $y \in \Gamma x$ ($\varphi(x) = 0$ для вершин, не имеющих образов).

Построение порядковой функции базируется на выделении множеств

$$X(0) = \{x \,/\, \Gamma x = \varnothing\},$$
$$X(1) = \{x \,/\, \Gamma x \subset X(0)\},$$
$$X(2) = \{x \,/\, \Gamma x \subset X(1)\}, \ldots$$
$$X(k) = \{x \,/\, \Gamma x \subset X(k-1)\}, \ldots$$
$$X(\omega) = \bigcup_{k<\omega} X(k),$$
$$X(\omega + 1) = \{x \,/\, \Gamma x \subset X(\omega)\}, \ldots$$

(символ ω использован как обобщение обычного целого числа – аналог бесконечности – для бесконечных последовательностей) и сводится к достаточно простой процедуре. Находим вершины, не имеющие образов, присваиваем им ранг 0. Затем берем вершины, все образы которых имеют ранг 0, и присваиваем ранг 1, далее берем вершины, для всех образов которых уже установлен ранг и т. д.

Индекс соответствующего множества сопоставляется всем его вершинам в качестве порядка (во многих приложениях этот индекс называют *рангом*, а нумерация ведется от нуля по возрастанию, причем в пределах одного ранга порядок нумерации произволен).

Примеры порядковой функции приведены на рис. 18.

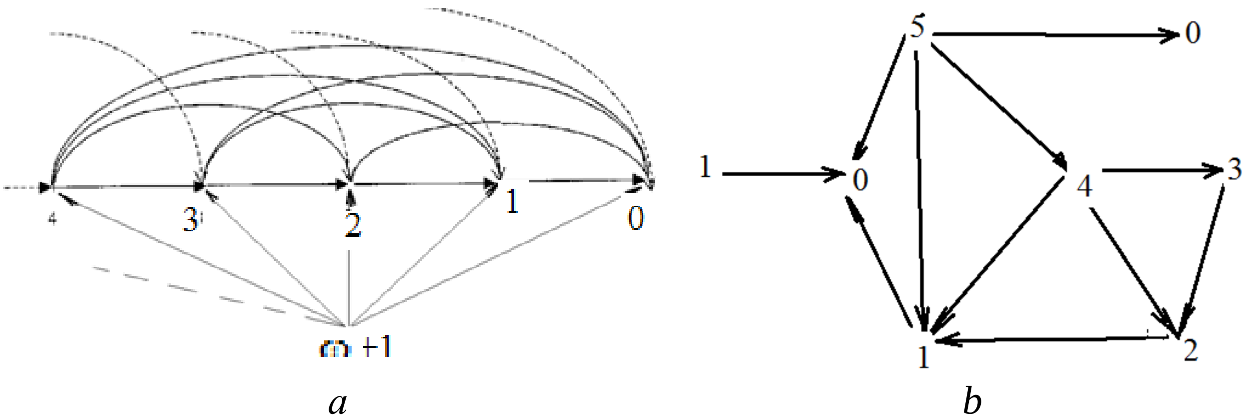


Рис. 18. Порядковая функция

Установление порядка минимизирует продолжительность численных процедур при поиске самых длинных или коротких путей в транспортных сетях и пр.

Естественно, что порядковая функция существует не для всех графов (например, для содержащих контуры).

4.6.2. Функция Гранди

Особое место в установлении порядка на графе занимает **функция Гранди**.

В случае конечного графа функция Гранди $g(x)$ сопоставляет вершине x *наименьшее из целых неотрицательных чисел, отличное от значений функции Гранди на образах этой вершины*.

Для бесконечного графа функция Гранди $g(x)$ определяется как наименьшее порядковое число, отличное от значений $g(\Gamma x)$. Если $\Gamma x = \varnothing$, то $g(x) = 0$.

Примеры функции Гранди приведены на рис. 19 (для графа на рис. 18, *а* она совпадает с порядковой). Граф может не иметь функции Гранди (например, при наличии контура нечетной длины) или иметь несколько функций Гранди (рис. 20).

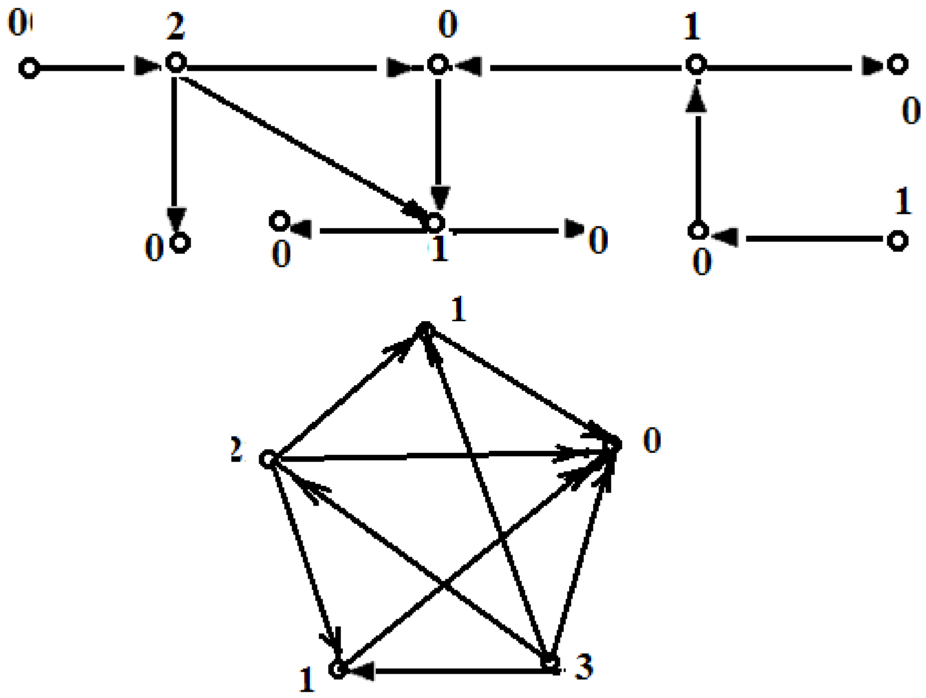


Рис. 19. Функция Гранди

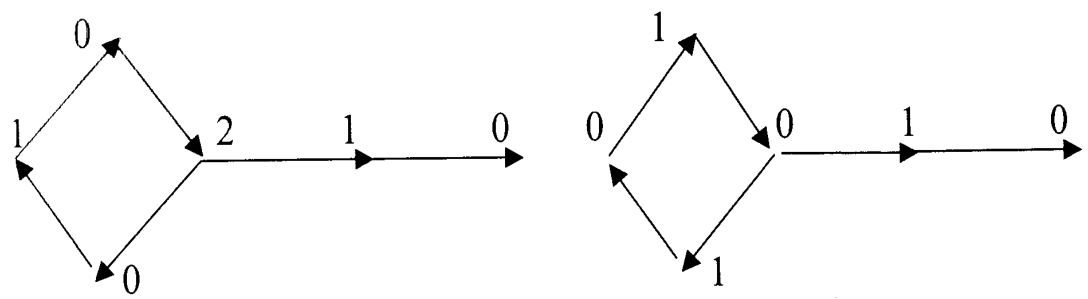


Рис. 20. Неоднозначность определения функции Гранди

Если определены функции Гранди $g_k(x)$ на N графах, то достаточно просто определяется функция Гранди на сумме этих графов:

$$g(x_1 x_2 \ldots x_N) = g_1(x_1) \oplus g_2(x_2) \oplus \ldots \oplus g_N(x_N),$$

где операция $\oplus$ (d-суммирование [1]) соответствует поразрядному сложению двоичных представлений чисел с приведением суммы по модулю 2 ($1 \oplus 1 = 0$).

Если читатель знаком с преобразованием целых чисел из десятичной системы в двоичную и обратно, то ему понятны приведенные ниже примеры:

$$
\begin{array}{rr}
 & 5 = 101_2 \\
\oplus & 7 = 111_2 \\
 & 25 = 11001_2 \\
\hline
 & 11011_2
\end{array} = 27
\qquad
\begin{array}{rr}
 & 47 = 101111_2 \\
\oplus & 63 = 111111_2 \\
 & 16 = 10000_2 \\
\hline
 & 000000_2
\end{array} = 0
\qquad
\begin{array}{rr}
 & 15 = 1111_2 \\
\oplus & 62 = 111110_2 \\
 & 28 = 11100_2 \\
\hline
 & 101101_2
\end{array} = 45
$$

В дальнейшем мы увидим эффективные применения функции Гранди на графе и на сумме графов для решения любопытных задач. Увы, что касается функции Гранди на произведении графов, то столь простого метода ее поиска нет.

4.7. Основные числа теории графов

4.7.1. Цикломатическое число мультиграфа

Рассмотрим мультиграф $G = (X, U)$, где X – множество вершин, U – множество *ребер* (в мультиграфе, в отличие от обычного графа, пара вершин может быть связана *более чем одним ребром*). Примером мультиграфа могут служить формулы органической химии.

Если мультиграф обладает n вершинами, m ребрами и p компонентами связности (подграфами, не имеющими общих вершин и ребер), то число $\nu(G) = m - n + p$ называется *цикломатическим числом мультиграфа*. Это число определяет количество независимых циклов, что делает его интересным в логистике для решения многих задач транспортного типа.

Если вспомнить понятие дерева как связного графа, не содержащего циклов, и учесть, что дерево с n вершинами имеет $n - 1$ ребро, то его цикломатическое число $\nu(G) = (n - 1) - n + 1 = 0$. Соответственно, цикломатическое число можно понимать как число ребер, которое нужно удалить, чтобы граф стал деревом.

4.7.2. Хроматическое число и хроматический класс графа

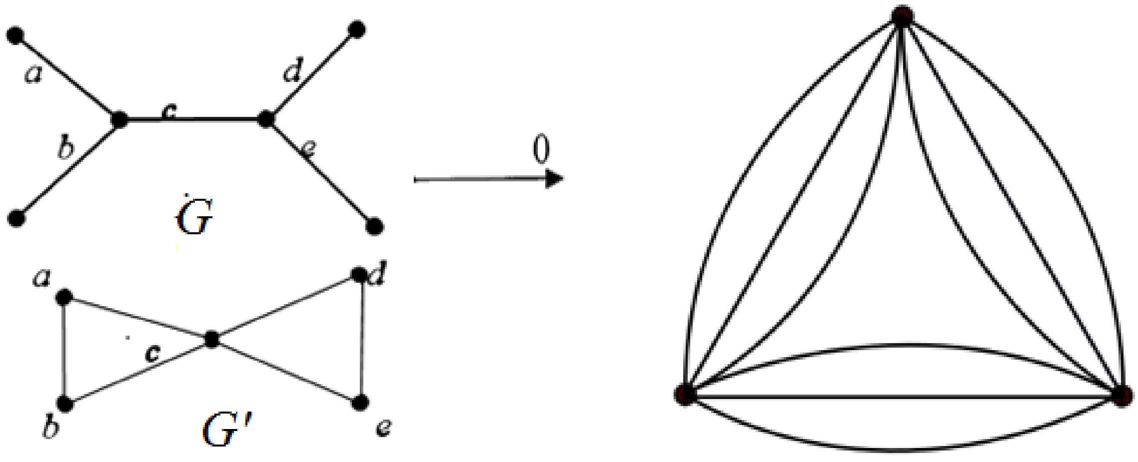
В 1878 г. Артур Кэли дал формальное изложение знаменитой **проблемы 4 красок**: доказать или опровергнуть, что для раскраски политической карты мира так, чтобы страны с общими границами имели разные цвета, достаточно 4 красок. Лишь 100 лет спустя удалось доказать это утверждение с привлечением компьютерных алгоритмов (не все математики считают подобные доказательства приемлемыми).

Граф называется p-хроматическим, если его вершины можно раскрасить p красками так, чтобы смежные вершины не были раскрашены одинаково. Наименьшее значение p называется *хроматическим числом графа* и обозначается $\gamma(G)$.

Соответственно, проблема 4 красок сводится к утверждению, что плоский граф (граф, который можно изобразить на плоскости без пересечений ребер вне вершин) является 4-хроматическим.

Минимальное число красок для раскрашивания ребер графа без одинакового окрашивания смежных ребер *называют хроматическим классом графа*.

Хроматический класс графа G совпадает с хроматическим числом графа G', который получается из G заменой ребер вершинами с сохранением смежности (рис. 21, *a*).



a *b*

Рис. 21. Хроматические класс и число графа

Изображенный на рис. 21, *b* мультиграф Шеннона шестой степени с хроматическим числом 3 имеет хроматический класс 9.

Что касается реализации процедуры окраски, то не бесполезны следующие соображения.

Теорема К. Шеннона (1949) утверждает, что мультиграфы можно раскрасить с использованием не более $3\lambda / 2$ цветов, где λ – максимальная степень вершин графа.

Теорема Визинга* (1964) утверждает, что ребра любого неориентированного графа (но не мультиграфа) могут быть раскрашены в число цветов, не более чем на единицу большее максимальной степени λ вершин графа. Здесь замечено, что неориентированные графы можно разбить на те, для которых достаточно λ цветов, и те, которые требуют $\lambda + 1$ цветов. Так, в графе, представляющем цикл с четным числом ребер, ребра можно раскрасить в два цвета, меняя эти цвета поочередно. При нечетном числе ребер степень вершин остается той же, искомое число красок равно 3.

Существуют алгоритмы окраски любого графа с помощью $\lambda + 1$ цветов, где λ – максимальная степень графа. Однако их трудоемкость достаточно велика и определяется степенями количества вершин.

Достаточно просто находится хроматическое число для суммы и произведения графов.

Так, если графы G и H соответственно $(p + 1)$- и $(q + 1)$-хроматические, то граф $G + H$ является $(r + 1)$-хроматическим, где

$$r = \max_{\substack{0 \le i \le p \\ 0 \le j \le q}} [i \oplus j]$$

(например, сумма 4- и 7-хроматических графов является 8-хроматической).

Если графы G и H соответственно p- и q-хроматические, то граф $G \times H$ является r-хроматическим, где $r = \min(p, q)$.

---

* *Визинг В. Г.* Об оценке хроматического класса p-графа // Дискретный анализ. – Новосибирск : Институт математики СО АН СССР, 1964. – Т. 3. – С. 25–30.

Визинг В. Г. Критические графы с данным хроматическим классом // Дискретный анализ. – Новосибирск : Институт математики СО АН СССР, 1965. – Т. 5. – С. 9–17.

Кстати, существует множество модификаций проблемы 4 красок. Так, если некоторые страны состоят из $m > 1$ несвязных частей и требование одинаковой раскраски для них сохраняется, то при $m = 2$ требуется 12 цветов, при $m = 3$ – 18 цветов, при $m = 4$ – 24 цвета (недоказанная гипотеза – при $m \geq 5$ – $6m$ цветов?) [7].

4.7.3. Число внутренней устойчивости

Множество $S \subset X$ называется внутренне устойчивым в графе (X, Γ), если *никакие две вершины этого множества не смежны*. Число $\alpha(G) = \max\limits_{S \subset X} |S|$ называется числом внутренней устойчивости.

Так для графа на рис. 22 можно построить несколько максимальных внутренне устойчивых множеств (например, {X0, X5, X6, X8} и $\alpha(G) = 4$).

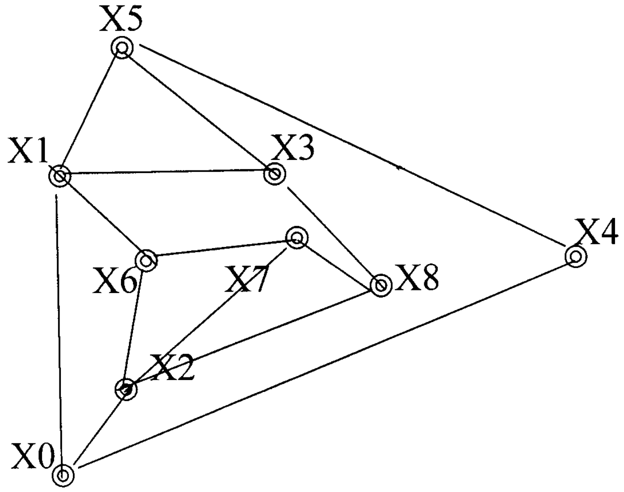
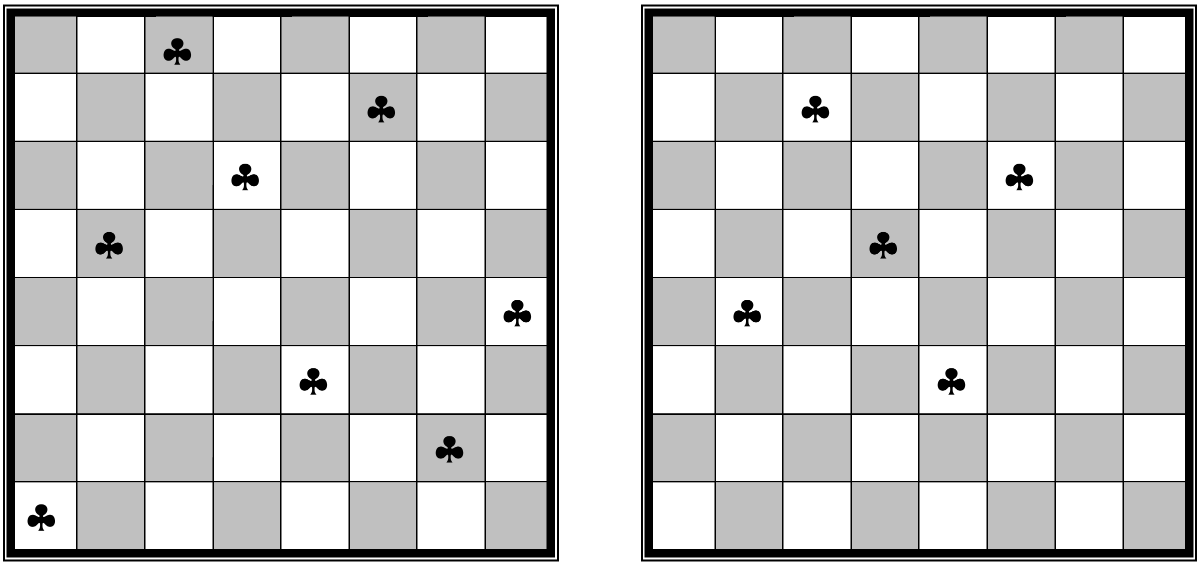


Рис. 22. Пример графа для построения внутренне устойчивых множеств

Другим примером поиска числа внутренней устойчивости служит задача непоражающего размещения 8 ферзей на шахматной доске (симметричный граф с 64 вершинами; 92 решения – 76 из них были найдены К. Гауссом) (рис. 23, *а*).

Аппарат внутренней устойчивости, в частности, используется при решении задачи помехозащищенной передачи сигналов (задача Шеннона об информационной емкости множества сигналов [1]).



a *b*

Рис. 23. Размещение ферзей на шахматной доске

4.7.4. Число внешней устойчивости

Множество $T \subset X$ называется внешне устойчивым в графе (X, Γ), если для каждой вершины $x \notin T$ существует хотя бы один образ в T. Число $\beta(G) = \min\limits_{T \subset X} |T|$ называется числом внешней устойчивости.

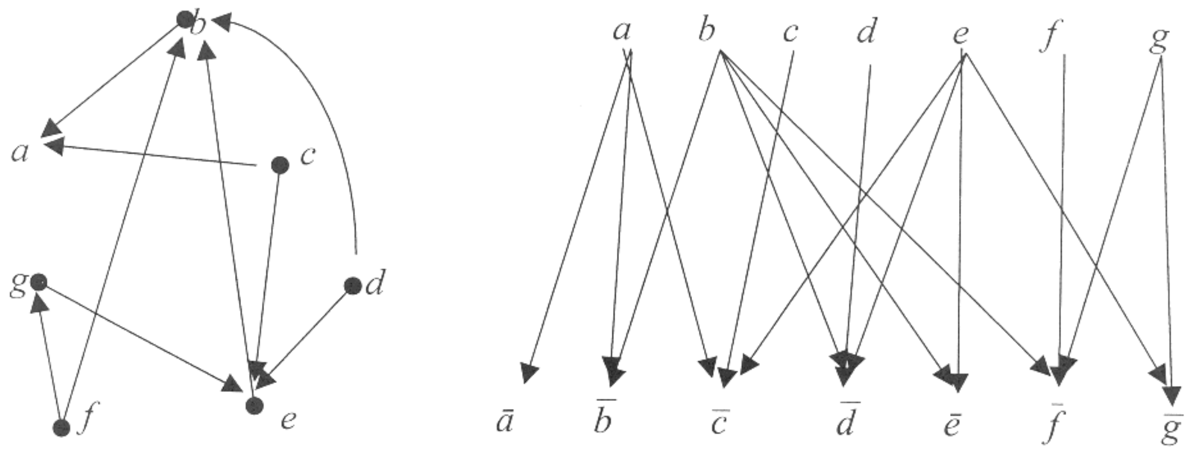
Примеры: сколько можно поставить ферзей (рис. 23, *b*) или других фигур, чтобы все поля доски были под ударом (5 ферзей, 8 ладей или слонов, 12 коней), сколько наблюдателей поставить на пересечениях улиц, чтобы все улицы были под присмотром.

Существует достаточно простой алгоритм поиска минимального внешне устойчивого множества [1]. Для графа $G = (X, \Gamma)$ (рис. 24, *a*) строим граф $(X, \overline{X}, \Delta)$, для которого определено отображение Δ множества X в $\overline{X}$ так, что $\overline{y} \in \Delta x$, если y совпадает с x или является прообразом x ($y \in \Gamma^{-1}x$) (рис. 24, *b*).

Дальнейшие действия:

1) удаляем из $(X, \overline{X}, \Delta)$ вершины x такие, что $\Delta x \subset \Delta y$ для $y \neq x$ (здесь мы удаляем c, d, f) и исходящие из них ребра;

2) если обнаружится «висячее ребро» (x, $\hat{y}$), включаем x в искомое множество T и удаляем эту вершину и ее образы (здесь $a \in T$ и из графа исключается a и $\Delta a = \{\bar{a}, \bar{b}, \bar{c}\}$);



a *b*

Рис. 24. Поиск минимального внешне устойчивого множества

3) повторяем предшествующие пункты до тех пор, пока не обнаружится невозможность дальнейшего приведения;

4) одну из оставшихся вершин (например, b) включаем во множество T и исключаем ее образы $\Delta b = \{\bar{d}, \bar{e}, \bar{f}\}$;

5) повторяем предшествующие пункты и получаем либо $T = (a, b, e)$, либо $T = (a, b, g)$.

4.8. Ядро графа и игры на графе

Множество $S \subset X$ называется ядром графа, если оно *устойчиво как внутренне, так и внешне*.

Например, для симметричного графа, изображенного на рис. 22, множество {X0, X5, X6, X8} с очевидностью внутренне устойчиво (вершины не смежные) и к тому же каждая из остальных вершин соседствует с избранными (имеет образ в этом множестве). Соответственно, это множество выступает как ядро графа.

Очевидно, что ядро не может содержать петель и должно включать все вершины, не имеющие образов.

Следует обратить внимание на то, что если граф G обладает ядром S_0, то $\alpha(G) \geq |S_0| \geq \beta(G)$. Так, для графа, выделенного на рис. 23, *а*, $\alpha(G) = 8$, $\beta(G) = 5$, и конструкция (рис. 23, *а*) полностью отвечает требованиям к свойствам ядра (все выбранные поля не смежны, а остальные «видят хотя бы одно из выделенных»).

Поиск ядра существенно упрощается при наличии функции Гранди.

Если граф допускает функцию Гранди $g(x)$, то множество $S = \{x \,/\, g(x) = 0\}$ служит ядром графа.

С очевидностью можно утверждать, что конечный граф (или локально конечный), не содержащий контуров нечетной длины, обладает ядром. Возможен случай нескольких ядер (см. рис. 20).

Так, в теории игр на основе понятия доминирования строится граф (X, Γ), где X – множество состояний и Γx определяет *эффективное предпочтение* для x. Основная теорема теории игр предлагает не выходить за пределы ядра S: внутренняя устойчивость означает, что никакой ситуации из S нельзя отдать предпочтение, а внешняя устойчивость говорит о том, что любой ситуации за пределами S можно найти предпочтительную ситуацию в S.

Красивым примером использования ядра служит игра типа Ним: два игрока поочередно берут спички из n кучек, причем можно брать любое число спичек из одной кучки. Забравший последние спички выигрывает.

Каждая кучка с m спичками, в сущности, представляет игру, в которой из любой вершины с некоторым значением функции Гранди можно перейти в любую вершину с меньшим значением.

Предлагаемая игра представляет сумму трех графов (в каждый очередной момент меняет свое состояние только один из графов), для которой функция Гранди определяется как

$$g(x_1 x_2 x_3) = g(x_1) \oplus g(x_2) \oplus g(x_3).$$

Если вам удастся попасть в ядро после своего хода (значение функции равно нулю), то ваш партнер вынужден перейти в ситуацию с ненулевым значением (выйти из ядра) и ваша чисто технически реализуемая задача состоит в возврате в ядро.

Пусть перед вашим ходом число спичек в кучках 15, 62 и 28. Выполняем действие

$$
\begin{array}{rr}
 & 15 = \quad 1111_2 \\
\oplus & 62 = 111110_2 \\
 & 28 = \;\; 11100_2 \\
\hline
 & 101101_2 \quad = 45 \neq 0
\end{array}
$$

Обнаружив ненулевое значение, получаем уверенность в своей победе и выясняем, в какой кучке удалять «лишние» единицы. Видим, что вторую следует привести к виду $010011_2 = 19$. Подвергнув преобразованию вторую кучку, удаляем 62 – 19 = 43 спички и получаем

$$
\begin{array}{rr}
 & 15 = \quad 1111_2 \\
\oplus & 19 = 10011_2 \\
 & 28 = 11100_2 \\
\hline
 & 000000_2 \quad = 0
\end{array}
$$

Очевидно, что соперник будет вынужден покинуть «ядро».

Существует обобщение игры на произвольное число кучек и возможность выбора из нескольких, но для этих случаев столь простых решений не найдено.

4.9. Эйлеровы циклы, цепи и контуры

Одна из головоломок для детей младшего школьного возраста с давних времен предлагала начертить какую-нибудь предложенную фигуру, «не отрывая пера от бумаги». Не миновало это развлечение и маститых математиков.

В 1736 г. великому Леонарду Эйлеру (1707–1783), свыше 30 лет работавшему в Петербургской Академии наук (1727–1741, 1766–1783), была предложена головоломка о кенигсбергских мостах. Дело в том, что в Кенигсберге было два острова, соединенных семью мостами с берегами реки Прегель и между собой (рис. 25, *a*). Жители города мечтали найти замкнутый маршрут прогулки по всем мостам, но их попытки были безуспешны.

Отвлекаясь от красот природы и зодчества, Эйлер рисует *мультиграф* (рис. 25, *b*) и ставит вопрос о возможности нарисовать эту фигуру, не отрывая пера от бумаги и не проводя линий дважды.

Цепь, проходящую *через все ребра графа только по одному разу*, ныне называют эйлеровой (соответствующая замкнутая цепь называется эйлеровым циклом).

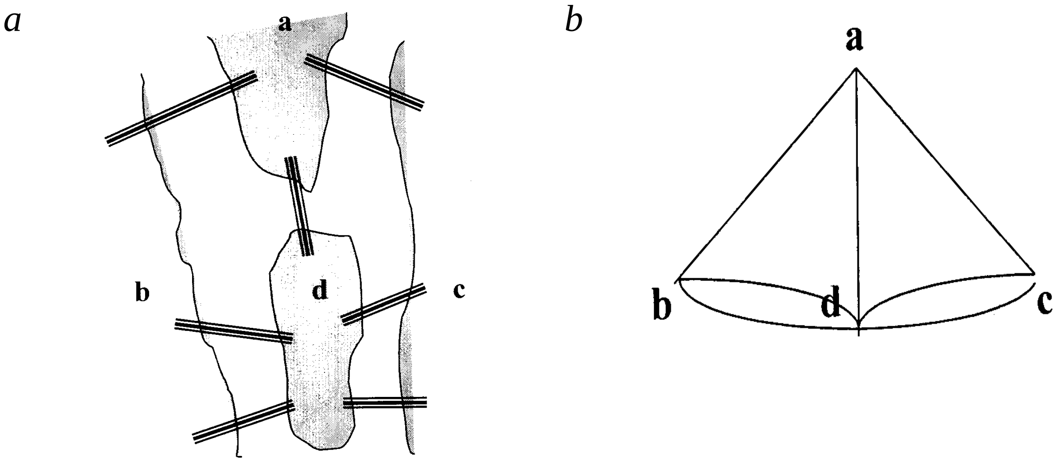


Рис. 25. Задача о кенигсбергских мостах

Мультиграф обладает эйлеровой цепью тогда и только тогда, когда он связен и число вершин нечетной степени (число ребер, инцидентных вершине) равно 2 или 0.

При отсутствии вершин нечетной степени существует эйлеров цикл, алгоритм построения которого разрешает выходить из любой вершины, последовательно удалять пройденные ребра и не ходить по ребру, которое в данный момент является перешейком (после его удаления граф распадается на несколько компонент связности).

Если граф является псевдосимметрическим (в любой вершине число входящих дуг совпадает с числом выходящих) и связным, то граф обладает *эйлеровым контуром*.

Результат Эйлера практически стал краеугольным камнем топологии и теории графов. Область его применения – от поиска маршрутов по выставочным залам Эрмитажа до психологических тестов и конструирования компьютеров.

4.10. Гамильтоновы пути и циклы

Путь в графе, проходящий *через все вершины графа только по одному разу*, называется гамильтоновым. Если он замкнут (конец пути совпадает с его началом), говорят о гамильтоновом контуре. При отсутствии требования ориентированности, соответственно, говорят о гамильтоновых цепях и циклах.

В 1859 г. ирландский математик Вильям Гамильтон (1805–1865) предложил игру «путешествие по додекаэдру» (рис. 26, *a*), где нужно было прогуляться по ребрам через все вершины, не попадая ни в одну из них дважды (додекаэдр – правильный 12-гранник, гранями которого служат пятиугольники, 20 вершин и 30 ребер). При желании додекаэдр можно (вырезав грань *ABCDE*) развернуть на плоскости (рис. 26, *b*) и обнаружить достаточно простой порядок перехода (рис. 26, *c*).

a *b* *c*

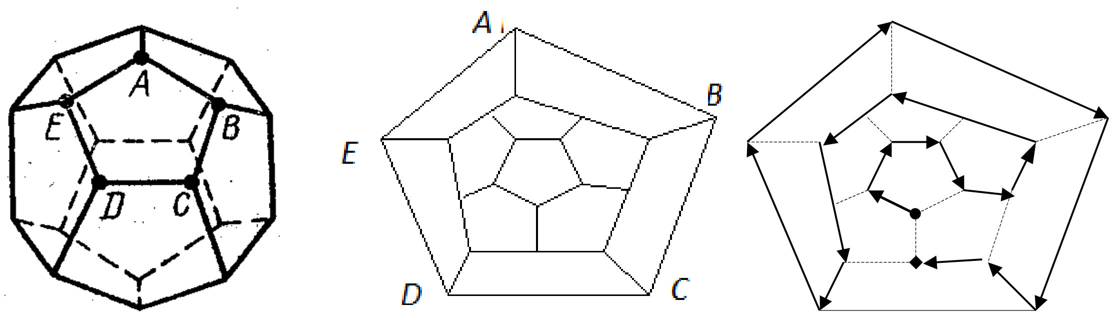


Рис. 26. Путешествие по додекаэдру

Существует достаточно много игр, связанных с поиском гамильтоновых маршрутов.

Примером может служить рассмотренная в [7, с. 108] задача о шахматном коне, который должен пройти от одного угла до противоположного через все клетки шахматной доски.

В полном графе (любая пара вершин соединена путем хотя бы в одном направлении) всегда есть гамильтонов путь.

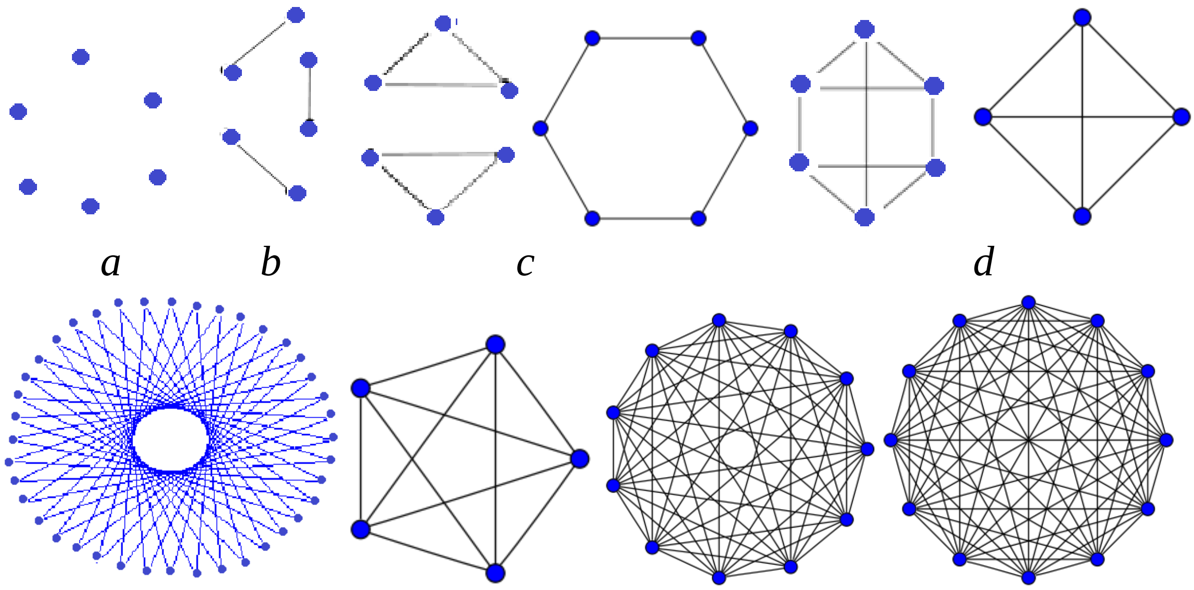
Задача поиска гамильтонова пути исключительно интересна в приложениях (в частности, в логистике). К сожалению, конструктивный алгоритм ее решения для любого графа, за исключением простейших, отсутствует.

4.11. Регулярные графы

Регулярным называют граф с равными степенями всех вершин. Такой граф с вершинами степени k называется *k-регулярным*.

Очевидно, что 0-регулярный граф состоит из изолированных вершин (рис. 27, *a*), 1-регулярный строится из изолированных ребер (рис. 27, *b*), 2-регулярный предполагает наличие циклов (рис. 27, *c*, *e*), 3-регулярный граф (рис. 27, *d*), принято называть кубическим.

Очевидно, что полным графам присуще свойство регулярности (рис. 27, *f*, *g*).



e *f* *g*

Рис. 27. Регулярные графы

Регулярные графы любопытны, в частности, при компьютерном построении некоторых орнаментов.

Так, для построения 2-регулярного графа, подобного изображению на рис. 27, *c*, можно воспользоваться известным уравнением окружности $x^2 + y^2 = R^2$, заданным в декартовых координатах:

$$x = R\cos(2\pi t),\ y = R\sin(2\pi t),\ t \in [0, 1].$$

Взяв шаг по углу поворота t, равный $h = 1 / N$, последовательно соединяя отрезками прямой точки (вершины) с координатами

$$x_k = R\cos(2\pi k h),\ y_k = R\sin(2\pi k h),\ k = 0, 1, \ldots, N,$$

получаем *правильный N-угольник*.

При выполнении соединения по принципу вложенного цикла

```
для k = от 0 до N – 1
   для l = от k + 1 до N
          соединение (xk, yk) с (xl, yl);
```

получим полный *N-регулярный граф* (рис. 27, *f*, *g*) (N = 5, 11, 12).

Если же выбирать шаг $h_t = 1 / k$ при нецелочисленном $k = m / n$ (m и n – целые, взаимно простые) и выполнять последовательный m-шаговый переход между вершинами, получаем 2-регулярный граф с m вершинами (звездоподобные фигуры типа рис. 27, *c* и другие орнаменты, вписанные в окружность).

Определенный интерес с позиций хроматичности, планарности и наличия эйлеровых или гамильтоновых циклов представляют кубические графы, в которых все вершины имеют степень 3.

Упоминавшаяся выше теорема о четырех красках (о раскраске вершин планарного графа) эквивалентна утверждению, что хроматическое число любого 3-регулярного графа «без мостов» равно 3.

Общеизвестен в связи с проблемой выявления планарности граф под названием *бикубический* или *кубический двудольный* (рис. 28).

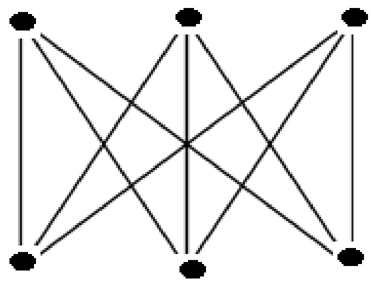
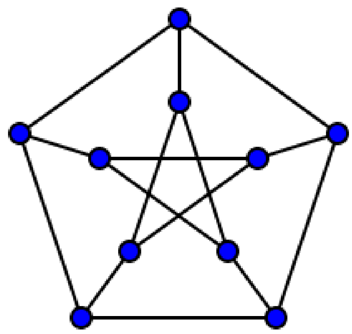


Рис. 28. Полный двудольный граф



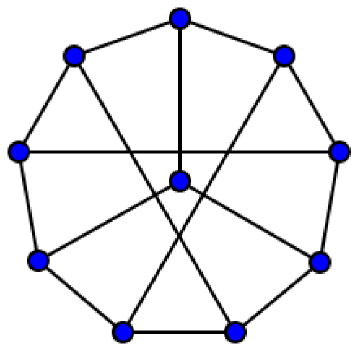


Рис. 29. Граф Петерсена

Любопытными свойствами характеризуется *кубический граф Петерсена* (тождественные представления см. на рис. 29). Для раскраски его вершин достаточно трех, а для ребер – четырех красок. Подграф, получаемый из него удалением какой-нибудь вершины, является гамильтоновым.

Существует гипотеза (Д. Эпштейн), что кубический граф с n вершинами имеет максимум $2^{n/3}$ различных гамильтоновых циклов.

Оригинальными из всех кубических графов являются лестницы Мёбиуса (рис. 30). Этот граф имеет четное число вершин, попарно соединенных ребрами-«перекладинами», своим видом напоминает лестницу, у которой начало и конец замкнуты с полупереворотом на 180°.

В связи с тем, что граф Мёбиуса похож на молекулы ДНК и к тому же синтезированы реальные молекулы со структурой в форме лестницы Мёбиуса, он интересен для химиков.

Не менее граф Мёбиуса интересен и для физиков, исследующих сверхпроводимость.

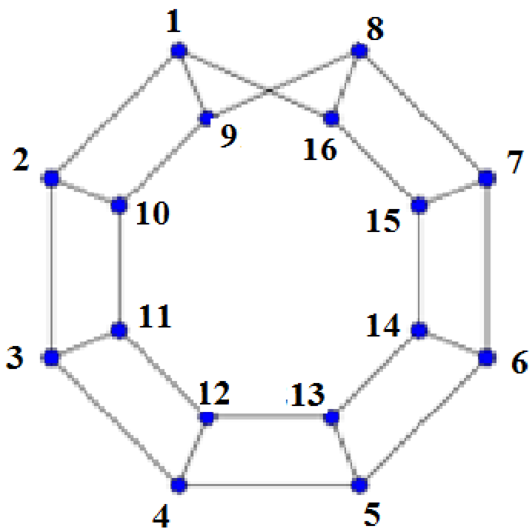


Рис. 30. Лестница Мёбиуса

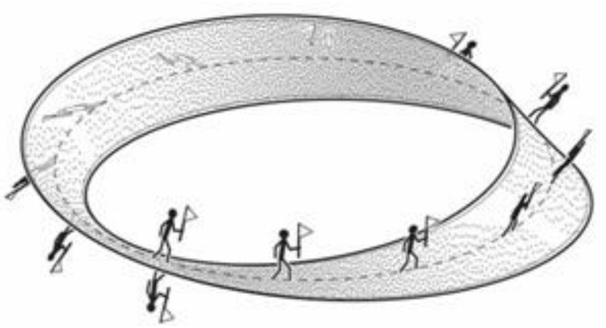


Рис. 31. Односторонняя поверхность (лента Мёбиуса)
Источник: http://demiurgcoach.ru/wp-content/uploads/2015/08/Petlya-mebiusa-800x350.jpg

4.12. Плоские графы и формула Эйлера

Граф, который можно изобразить на плоскости без пересечения ребер вне вершин, называется *плоским топологическим графом*.

Проблема построения плоского (планарного) графа возникает при создании печатных схем для компьютера, транспортных развязок в городе и др.

Теорема Л. С. Понтрягина – К. Куратовского. Чтобы граф был плоским, необходимо и достаточно, чтобы он не содержал в себе частичных подграфов типа приведенных на рис. 32.

Область плоскости, ограниченная ребрами и не содержащая в себе вершин или ребер, называется *гранью* графа.

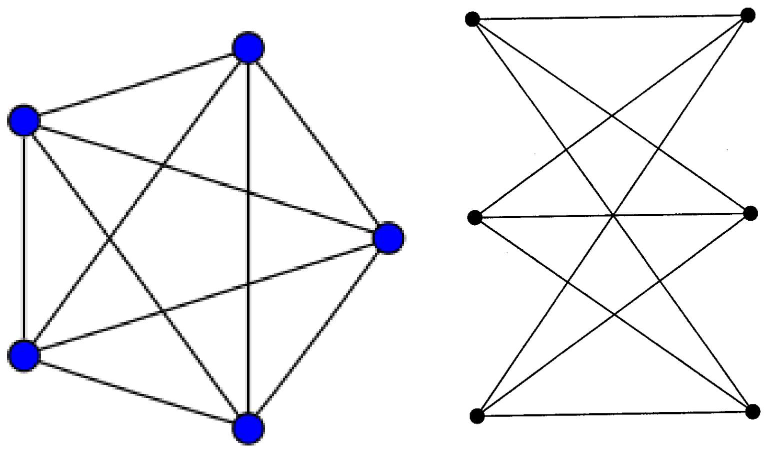


Рис. 32. Непланарные графы

Формула Эйлера. Если связный плоский топологический граф имеет n вершин, m ребер и f граней, то $n - m + f = 2$.

Из этой изящной формулы получено множество следствий. Так доказано, что в плоском графе (но не мультиграфе) хотя бы одна вершина имеет степень, не превышающую 5.

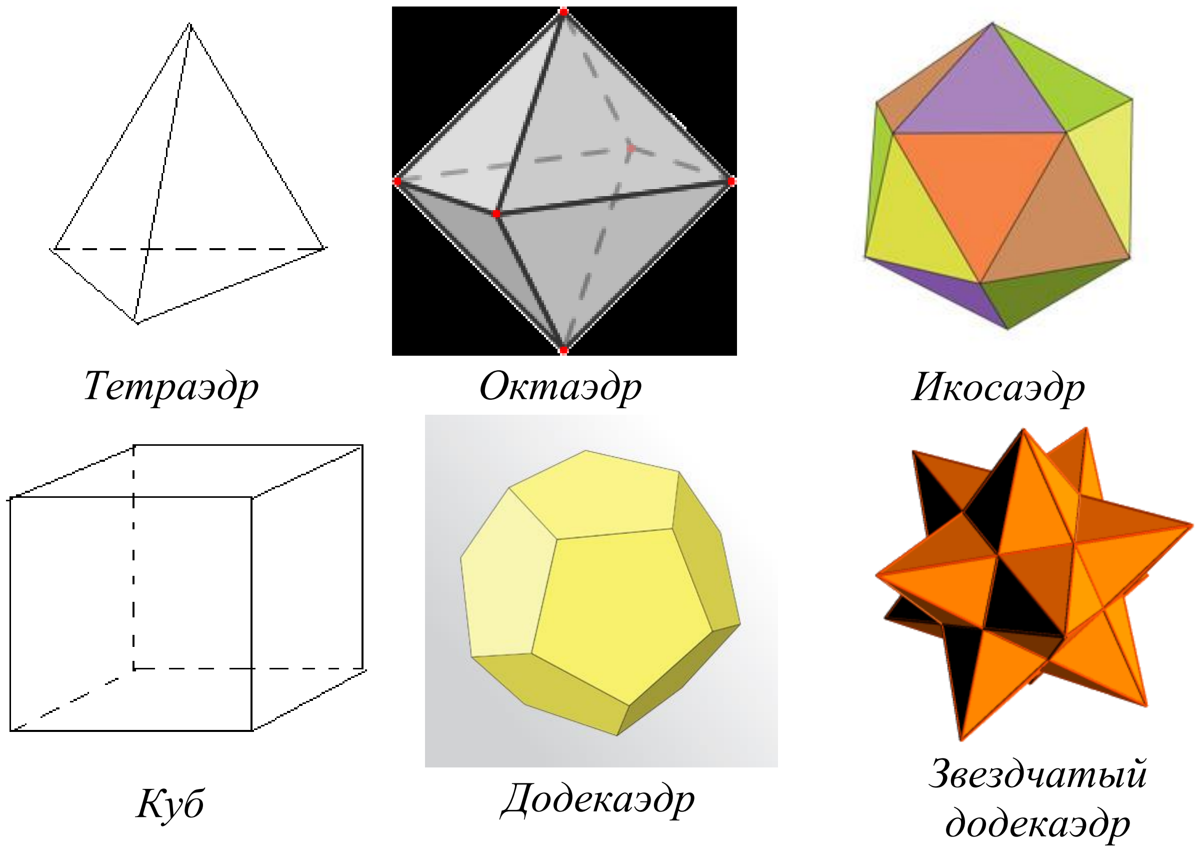


Рис. 33. Правильные многогранники

Если на географической карте принять за ребра участки границ государств, а за вершины – их стыки (число инцидентности для всех вершин не менее трех), то существует грань, край которой содержит не более 5 ребер.

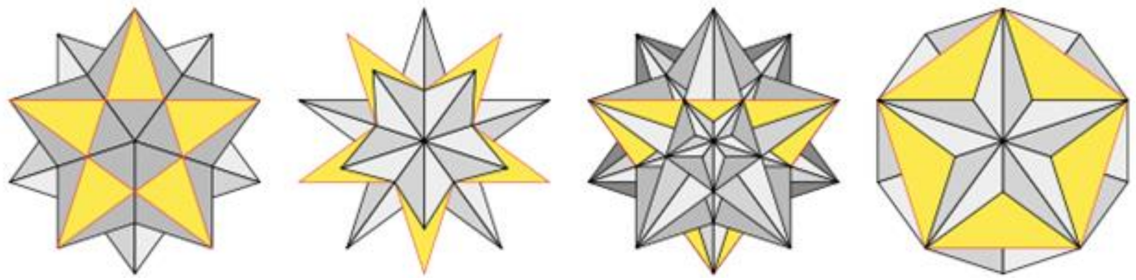
Существует обобщение формулы Эйлера и на пространственные фигуры. Так соотношение $n - m + f = 2$ выполняется в любом выпуклом многограннике.

Можно показать, что в любом выпуклом многограннике существует хотя бы одна грань в форме треугольника, четырехугольника или пятиугольника.

Еще к Евклиду и Платону восходит утверждение, что *единственными правильными многогранниками* (рис. 33) *являются тетраэдр, октаэдр, икосаэдр, куб и додекаэдр* (выпуклый многогранник считается правильным, если все его грани являются одинаковыми правильными многоугольниками и в каждой его вершине сходится одинаковое число ребер).

Для этих *платоновых* фигур

	n вершин	m ребер	f граней		n вершин	m ребер	f граней
Тетраэдр	4	6	4	**Октаэдр**	6	12	8
Куб	8	12	6	**Икосаэдр**	12	30	20
Додекаэдр	20	30	12				



Малый звездчатый додекаэдр *Большой звездчатый додекаэдр* *Большой икосаэдр* *Большой додекаэдр*

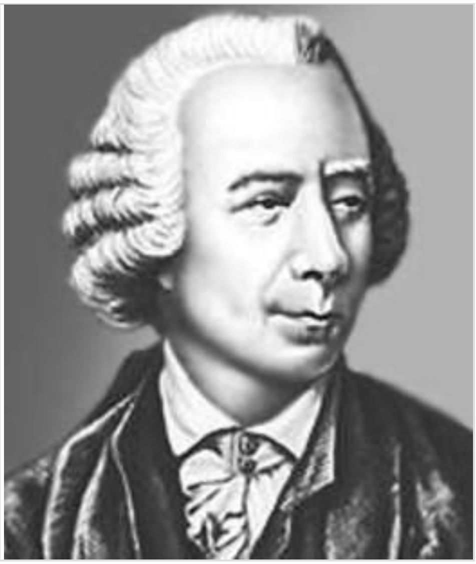
Рис. 34. Многогранники Кеплера – Пуансо

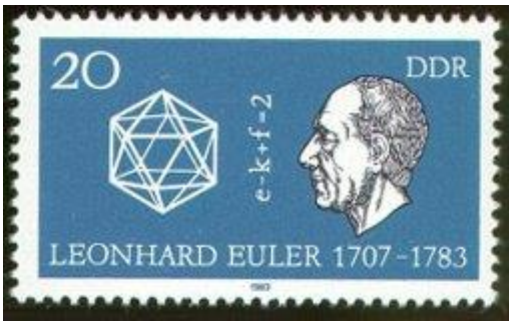
Источник: http://otherreferats.allbest.ru/mathematics/00154821_0.html

Наряду с указанными правильными выпуклыми многогранниками, заслуживают внимания звездчатые многогранники, которые представляют интерес для ювелиров, создателей елочных украшений и др.

Установлено О. Коши в 1811 г., что существуют всего 4 правильных звездчатых тела, которые не являются производными от соединения платоновых и звездчатых тел (рис. 34).

Заметим, что имя Эйлера всегда пользовалось исключительной популярностью, о чем свидетельствует многообразие почтовых марок, монет и публикаций, посвященных великому математику.





4.13. Глоссарий теории графов

Антисимметрический граф – граф, где две любые смежные вершины соединены только в одном направлении.

Бесконечный граф – граф, содержащий бесконечно много вершин и/или ребер.

Бинарное дерево – дерево, где каждая вершина инцидентна либо одному, либо трем ребрам (степень вершины равна 1 или 3). Вершины степени 1 называют его *листьями*.

Вершина, узел – базовое понятие: точка, где сходятся ребра (дуги).

Взвешенный граф – граф, каждому ребру которого поставлено в соответствие некое значение (вес).

Висячая вершина – вершина, степень которой равна 1.

Висячее ребро – ребро графа, не имеющее предшествующего или последующего.

Внешне устойчивое множество – подмножество вершин графа, где для каждой внешней для него вершины существует хотя бы один образ в нем.

Внутренне устойчивое множество – подмножество вершин графа, где никакие две его вершины не смежны.

Высота дерева – наибольшая длина пути от корня к листу.

Гамильтонов граф – граф, в котором есть гамильтонов цикл.

Гамильтонов путь – простой путь в графе, содержащий все вершины графа только по одному разу.

Гамильтонов цикл – простой цикл в графе, содержащий все вершины графа ровно по одному разу.

Грань – область, ограниченная ребрами в плоском графе и не содержащая внутри себя вершин и ребер графа. Внешняя часть плоскости тоже образует грань.

Граф – базовое понятие, определяемое как множество вершин и его отображение в себя.

Дерево – связный граф, не содержащий циклов.

Длина пути – число составляющих его дуг пути (или во взвешенном графе сумма длин его дуг, если таковые заданы).

Дуга – показатель связи вершин x и y в отображении $y = F(x)$, ориентированное ребро.

Изоморфизм. Два графа называются *изоморфными*, если существует взаимно-однозначное соответствие между их вершинами и ребрами, которое сохраняет смежность и инцидентность (графы отличаются только названиями своих вершин).

Инцидентность – понятие в отношении ребра (дуги) и вершины: если x и y – вершины, а $p = (x, y)$ – соединяющее их ребро(дуга), тогда вершина x (равно как и y) и ребро p инцидентны.

Клика – подмножество вершин графа, полностью соединенных друг с другом, т. е. подграф, являющийся полным графом.

Компонента связности графа – некоторое подмножество вершин графа, между любыми двумя вершинами которого существует цепь из одной в другую, и не существует цепи из его вершин в вершины не из этого множества.

Конечный граф – граф, содержащий конечное число вершин и ребер.

Контур – замкнутый путь в графе.

Корень дерева – выбранная вершина дерева; в ориентированном графе – вершина, не имеющая прообразов с нулевой степенью захода.

Кубический граф – регулярный граф степени 3, то есть граф, где каждой вершине инцидентно ровно три ребра.

Лес – неориентированный граф без циклов, компонентами связности которого являются деревья.

Лист дерева – вершина дерева с единственным ребром или входящей дугой.

Матрица инцидентности – матрица, элементы которой характеризуют инцидентность соответствующих вершин графа (по вертикали) и его ребер (по горизонтали). Для неориентированного графа элемент равен 1, если соответствующие вершина и ребро инцидентны; для ориентированного равен 1, если инцидентная вершина – начало ребра, и равен – 1, если эта вершина – конец ребра; в остальных случаях равен 0.

Матрица смежности – матрица, элементы которой характеризуют смежность вершин графа. Их значения равны количеству ребер (дуг), соединяющих соответствующие вершины.

Множество сочленения – непустое множество вершин, после удаления которого связный граф перестает быть таковым.

Мост – ребро, удаление которого увеличивает количество компонент связности в графе.

Мультиграф – граф, где есть пара вершин, соединенных более чем одним ребром или более чем двумя дугами противоположных направлений.

Образ – вершина y в отображении $y = F(x)$.

Петля – контур единичной длины.

Планарный (плоский) граф – граф, который может быть изображен (уложен) на плоскости без пересечения ребер.

Подграф графа получается выбором части вершин исходного графа с сохранением всех присутствующих связей между этими вершинами.

Полный граф – граф, где любые две вершины соединены хотя бы в одном направлении.

Порядковая функция – функция, сопоставляющая вершинам ориентированного графа индексы (номера) так, чтобы индексы образов не превышали индексов прообразов.

Произведение (пересечение) графов – граф $G = (X, \Gamma)$, где

$X = X_1 \times X_2 \times \ldots \times X_n = \{(x_1\ x_2 \ldots x_n) / x_\mathrm{i} \in X_i\}$,

$\Gamma(x_1\ x_2 \ldots x_n) = \Gamma_1 x_1 \times \Gamma_2 x_2 \times \ldots \times \Gamma_n x_n$.

Прообраз – вершина x в отображении $y = \Gamma(x)$.

Простая (элементарная) цепь – цепь, все вершины которой различны.

Простой путь – путь, в котором никакая дуга не встречается дважды.

Путь – это последовательность вершин $x_1, x_2, \ldots, x_n$, для которой существуют соответствующие дуги.

Ребро – базовое понятие. Дуга без учета ориентации.

Регулярный граф – граф с равными степенями всех вершин.

Связность – две вершины графа *связаны*, если существует соединяющая их цепь.

Связный граф – граф, в котором все вершины связаны.

Сильно связный граф – граф, в котором для любых двух различных его вершин x и y существует путь из x в y и из y в x.

Симметрический граф – граф, в котором две любые смежные вершины соединены в обоих направлениях.

Смежность – понятие в отношении двух ребер либо двух вершин. Два ребра, инцидентные одной вершине, *смежны*; две вершины, инцидентные одному ребру, также называются *смежными*.

Составной путь – путь, в котором какая-то дуга встречается неоднократно.

Степень вершины – количество дуг (ребер), входящих в вершину или исходящих из нее. Петля добавляет 2 к степени вершины.

Сумма (объединение) графов – граф $G = (X, \Gamma)$, в котором
$X = X_1 \times X_2 \times \ldots \times X_n = \{(x_1\, x_2 \ldots x_n) \,/\, x_i \in X_i\}$,
$\Gamma(x_1\, x_2 \ldots x_n) = [\Gamma_1 x_1 \times \{x_2\} \times \ldots \times \{x_n\}] \bigcup [\{x_1\} \times \Gamma_2 x_2 \times \ldots \times \{x_n\}] \bigcup$
$\bigcup \ldots \bigcup [\{x_1\}] \times \{x_2\} \times \ldots \times \Gamma_n x_n]$.

Точка простого сочленения – точка сочленения, связанная с каждой из компонент связности подграфа только одним ребром.

Точка сочленения – вершина связного графа, при удалении которой получаемый подграф несвязен.

Укладка – размещение графа на некоторой поверхности таким образом, чтобы его ребра при этом не пересекались (пересекались только в своих концевых вершинах).

Функция Гранди сопоставляет вершине наименьшее из целых неотрицательных чисел, отличное от значений функции Гранди на образах этой вершины.

Хроматический класс графа – минимальное число красок для раскрашивания ребер графа без одинакового окрашивания смежных ребер.

Хроматическое число графа – минимальное количество цветов, требуемое для раскраски вершин графа, при которой любые вершины, соединенные ребром, раскрашены в разные цвета.

Центроиды, *периферoccoиды*

Цепь в графе – аналог понятия пути в графе без ориентации.

Цикл – замкнутая цепь.

Цикломатическое число – минимальное число ребер, которые надо удалить, чтобы граф стал ациклическим.

Частичный граф получается выбором всех вершин исходного графа с потерей некоторых связей между ними.

Частичный подграф получается выбором части вершин исходного графа с сохранением лишь части присутствующих связей между этими вершинами.

Число внешней устойчивости – число вершин минимального внешне устойчивого в графе подмножества.

Число внутренней устойчивости – число вершин максимального внутренне устойчивого в графе подмножества.

Число связности – размерность наименьшего множества сочленения.

Шарнир – точка сочленения.

Эйлеров граф – это граф, в котором существует эйлеров цикл.

Эйлеров контур – это путь, содержащий все дуги графа по одному разу.

Эйлерова цепь (или *эйлеров цикл*) – это цепь (цикл), которая содержит все ребра графа (вершины могут повторяться).

Элементарный путь – путь, в котором никакая вершина не встречается дважды.

Ядро графа – подмножество вершин, устойчивое как внутренне, так и внешне.

4.14. Упражнения

1. Приведите примеры графов:

а) с пятью вершинами и пятью ребрами;

б) с пятью вершинами и тремя ребрами;

в) с тремя вершинами и пятью ребрами.

2. Вычислите, сколько существует различных простых графов с четырьмя вершинами (напоминаем, что граф называется простым при отсутствии кратных ребер и петель).

3. Нарисуйте все регулярные графы с шестью вершинами (напоминаем, что граф называется регулярным при равенстве степеней всех вершин). Выполните это же задание для графа с семью вершинами. Почему графов с семью вершинами, удовлетворяющих поставленным условиям, меньше?

4. Нарисуйте граф, представленный с помощью матрицы смежности.

1

	a	***b***	***c***	***d***
a	1	1	0	1
b	0	0	1	1
c	0	0	0	1
d	0	0	1	0

2

	1	**2**	**3**	**4**	**5**
1	0	0	1	0	1
2	0	0	0	1	0
3	1	0	0	0	1
4	0	1	0	0	0
5	1	0	1	0	0

3

	1	2	3	4	5
1	0	0	1	1	0
2	0	0	1	0	0
3	1	1	0	0	1
4	1	0	0	0	1
5	0	0	1	1	0

4

	1	**2**	**3**	**4**	**5**
1	0	1	1	1	0
2	1	0	0	0	1
3	1	0	0	0	0
4	1	0	0	0	1
5	0	1	0	1	0

5

	1	**2**	**3**	**4**	**5**
1	0	0	0	0	0
2	0	0	1	0	0
3	1	0	0	0	0
4	0	0	1	0	0
5	0	0	0	1	0

6

	1	**2**	**3**	**4**	**5**
1	0	1	1	1	1
2	0	0	1	1	1
3	0	0	0	1	1
4	0	0	0	0	1
5	1	0	0	1	0

7

	1	**2**	**3**	**4**	**5**	**6**
1	0	1	1	0	1	0
2	1	0	0	1	1	0
3	1	0	0	0	1	1
4	0	1	0	0	0	0
5	1	1	1	0	0	1
6	0	0	1	0	1	0

8

	1	**2**	**3**	**4**	**5**	**6**
1	0	1	0	0	0	0
2	0	0	1	0	1	0
3	0	1	0	1	0	0
4	0	0	1	0	1	1
5	0	1	0	1	0	0
6	0	0	0	0	0	0

9

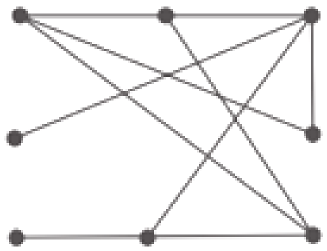
	1	**2**	**3**	**4**	**5**	**6**
1	1	1	1	0	1	0
2	0	1	0	0	0	1
3	1	0	0	0	1	0
4	0	0	0	0	0	1
5	0	0	1	1	1	0
6	1	0	0	0	1	1

10

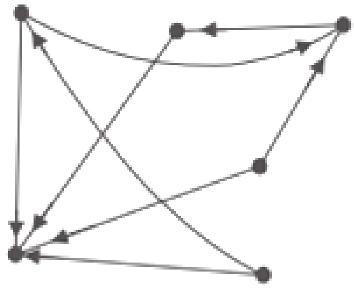
	1	**2**	**3**	**4**	**5**	**6**
1	0	1	1	0	1	0
2	0	1	0	0	1	0
3	0	0	0	0	0	0
4	0	0	1	0	0	0
5	0	0	0	1	0	0
6	1	0	0	0	1	1

5. Постройте матрицу смежности для следующих графов.

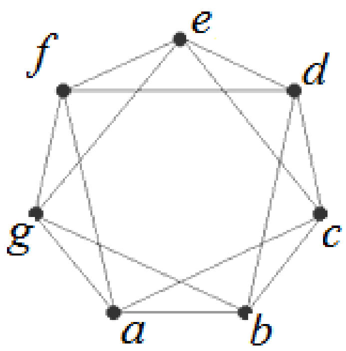
1



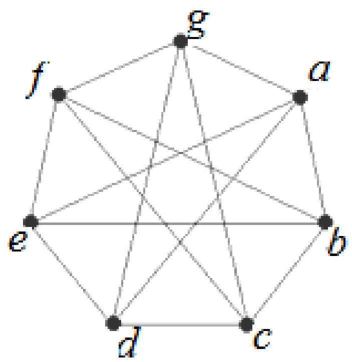
7



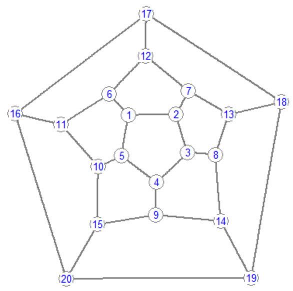
2



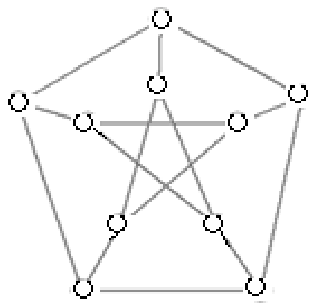
8



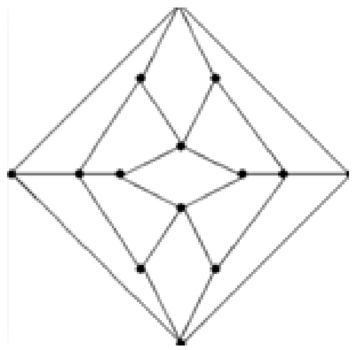
3



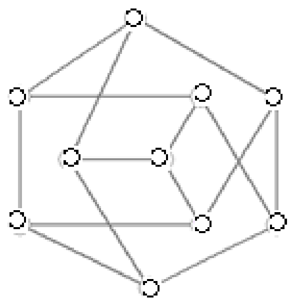
9



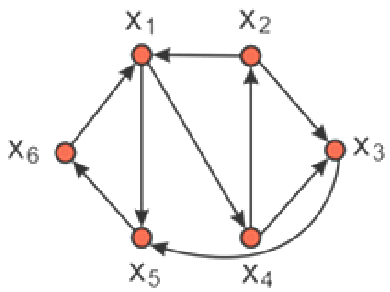
4



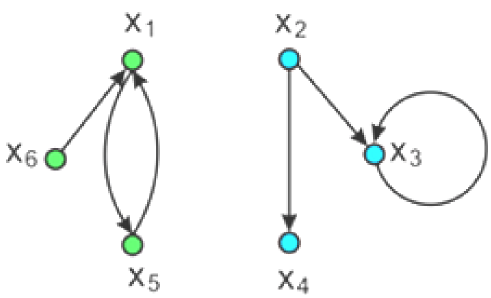
10

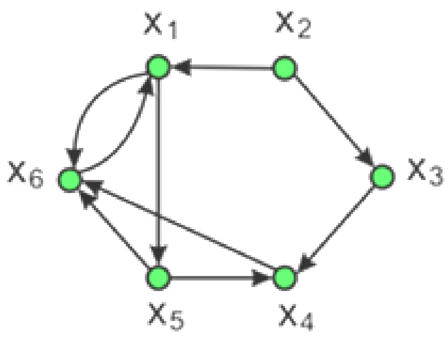


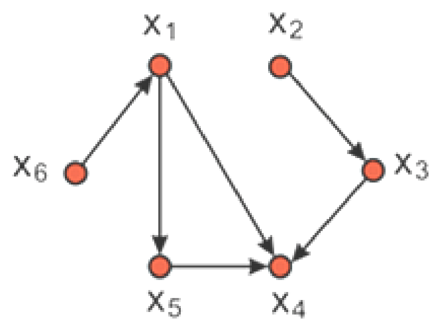
5



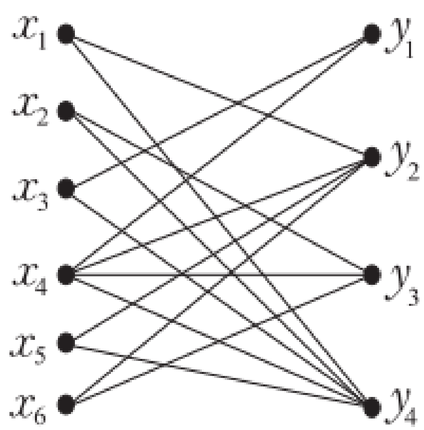
11

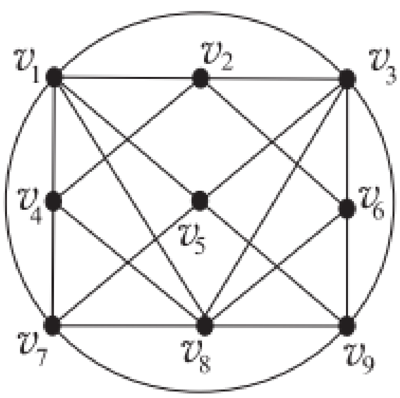


6 

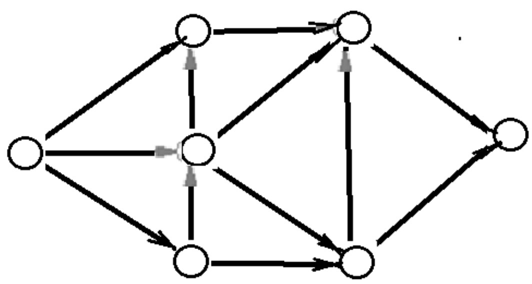
12 

6. В нижеприведенных графах найдите эйлеров цикл или цепь.

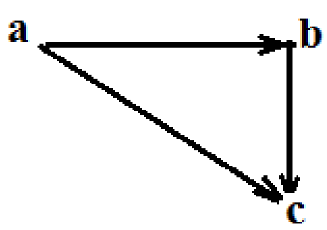


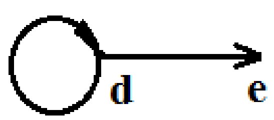


7. Для приведенного графа постройте порядковую функцию и функцию Гранди.

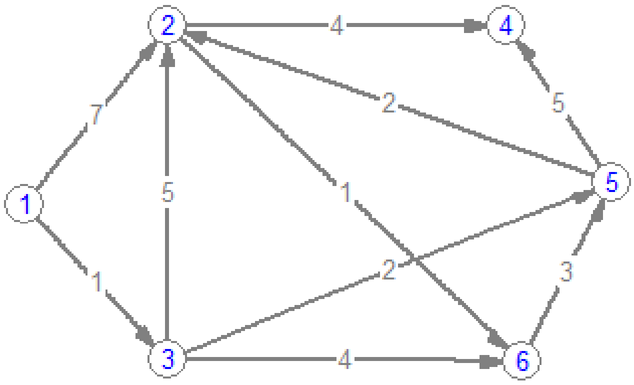


8. Найдите сумму и произведение нижеприведенных графов.

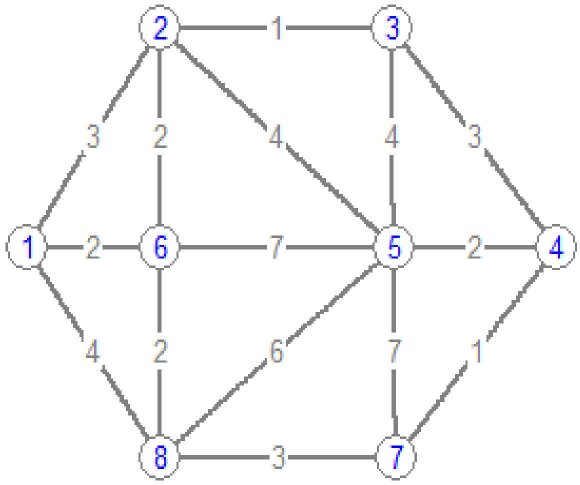




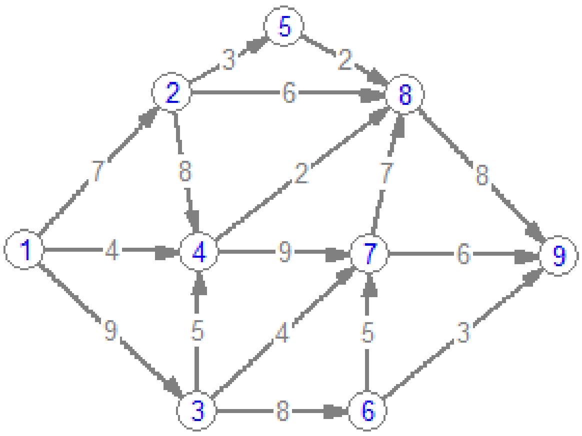
9. Найдите кратчайшие пути в графах.



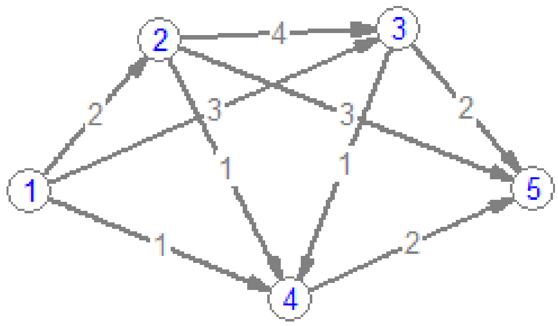
Из 1 в 4



Из 1 в 4

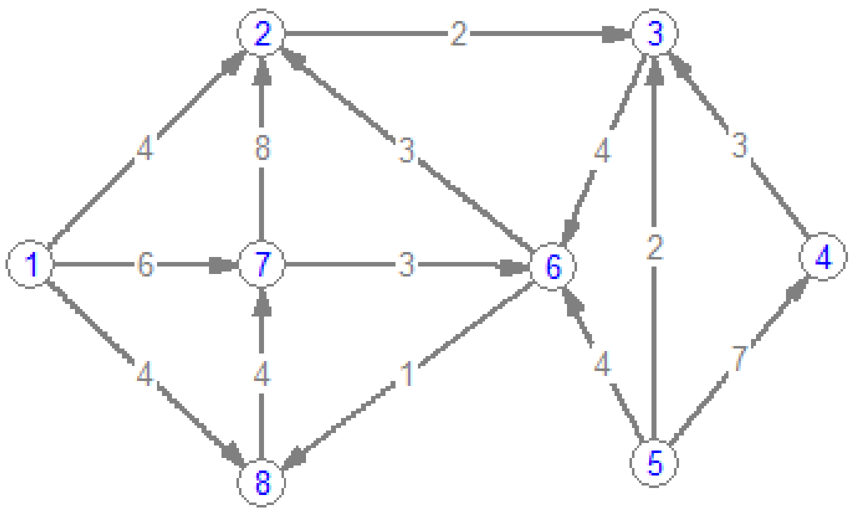


Из 1 в 9



Из 1 в 5

10. Найдите критический путь (1 – 8) в графе.



Глава 5. ОПТИМИЗАЦИЯ НА ГРАФАХ

Древнегреческому герою Тезею не понадобилась бы нить Ариадны, чтобы выбраться из лабиринта Минотавра, если бы он руководствовался простейшим правилом: для поиска пути в лабиринте, выйдя на перекресток, иди крайним левым путем и, зайдя в тупик, отступи на шаг и «удали тупиковый коридор». Но как долго пришлось бы ему пребывать в лабиринте? А сколько времени он провел бы в размышлениях о приемлемом пути даже в случае обнаружения «в кустах» карты этого циклопического сооружения?

А какой по расходу бензина маршрут по автодорогам страны от Кемерово до Санкт-Петербурга выбрать почитателю архитектуры Северной Пальмиры? А как быстрее и дешевле организовать доставку молока от гормолзавода до потребителей?

Сотни таких простых и непростых вопросов, и отнюдь не так много простых решений…

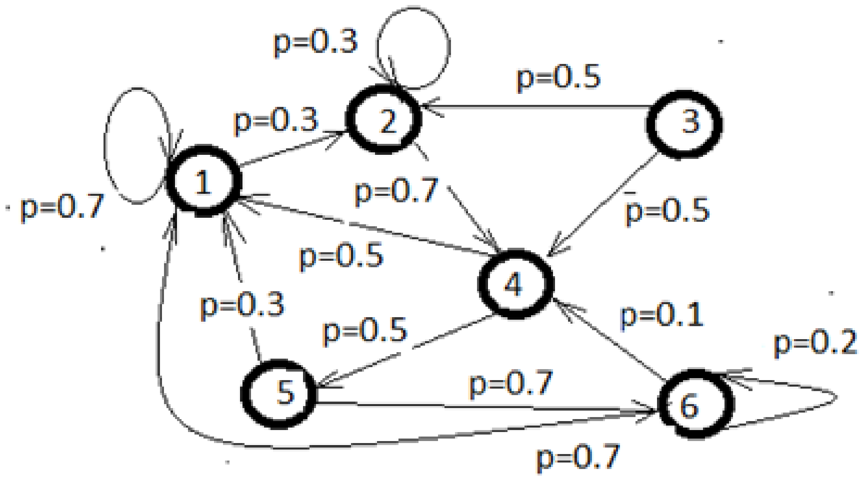
Выше мы рассматривали топологические графы, где отсутствует какая-то прозаическая метрика и под длиной пути имеется в виду число составляющих его дуг, а сами дуги (ребра) выступают как характеристики смежности вершин. Зная, что в полном графе (любая пара вершин соединена путем хотя бы в одном направлении) всегда есть гамильтонов путь, мы не гарантированы, что таких путей будет весьма много, и мы окажемся в положении буриданова осла, который умер от голода, не сумев отдать предпочтение одной из предложенных охапок сена.

Граф, в котором каждому ребру (дуге) сопоставлено некоторое число, называют *взвешенным*. Это число может характеризовать длину отрезка пути, пропускную способность газо- или водопровода, риск капиталовложения, время выполнения работы и т. п. Соответственно, для такого графа могут возникать так называемые оптимизационные задачи, связанные с выбором поведения, которое обеспечивает кратчайший (или самый длинный) маршрут, оптимальное давление газа в трубе, минимальный риск при формировании портфеля ценных бумаг и т. п.

5.1. Марковские системы и случайные процессы

Например, некоторая система (физическая или социальная) может находиться в одном из n состояний и в каждый очередной момент времени t переходить из одного состояния i в другое j с вероятностью $p(i, j)$*. Если эта вероятность не зависит от того, что происходило с системой до попадания в состояние i, такая система называется *марковской* (по имени одного из создателей теории случайных процессов А. А. Маркова (1856–1922)).

Например, систему, изображенную на рис. 35, можно представить матрицей вероятностей переходов (незаполненные клетки соответствуют невозможным переходам – нулевым вероятностям).



	1	2	3	4	5	6
1	0,7	0,5				
2		0,3		0,7		
3		0,5		0,5		
4	0,5				0,5	
5	0,3					0,7
6	0,7			0,1		0,2

Рис. 35. Система с матрицей вероятностей переходов

---

*Под вероятностью p события понимаем отношение числа благоприятствующих исходов к общему числу возможных исходов. При p = 0 наступление события невозможно, при p = 1 – оно гарантировано, p = 0,7 – наступает в 70 случаях из 100.

Для такой системы можно предсказать шансы оказаться в том или ином состоянии через какое-то время или в далеком будущем.

Так, если обозначить через $X_t(i)$ вероятность того, что в момент времени t система находится в состоянии i, то ожидаемая вероятность нахождения системы в любом состоянии в последующий момент равна $X_{t+1} = p \cdot X_t$ или в развернутой форме

$$X_{t+1}(j) = \sum_{i=1}^{n} P_{ij} X_t(i),\ j = 1, 2, \ldots, n.$$

Ниже приведены соответствующие вероятности для системы, начинающей жизнь в состоянии 1.

X_0	X_1	X_2	X_3	X_4	X_5	X_6	
1	0,7000	0,4900	0,5180	0,5901	0,6896	0,7653	…
0	0	0,3500	0,4550	0,5530	0,5652	0,5601	…
0	0	0,2500	0,4250	0,5250	0,5617	0,5616	…
0	0,5000	0,5000	0,5950	0,5705	0,5579	0,6181	…
0	0,3000	0,7000	0,6230	0,5257	0,5466	0,6099	…
0	0,7000	0,6800	0,5290	0,5279	0,5757	0,6536	…

Если процесс переходов между состояниями сопровождается некоторыми доходами (убытками), то можно предсказать и ожидаемый суммарный доход (убыток) [9].

5.2. Поиск кратчайшего пути (цепи)

Существует множество задач, связанных с поиском тех или иных путей в транспортных сетях (мы уже затрагивали вопрос обнаружения наличия путей и контуров в п. 4.9). Как правило, такая сеть представляет собой граф, дугам (ребрам) которого сопоставлены некоторые числа. Если эти числа задают длину соответствующих фрагментов сети, то могут быть поставлены и решены разнообразные оптимизационные задачи.

Так, при поиске кратчайшего пути (цепи) можно воспользоваться алгоритмом Э. В. Дейкстры [3], задающим поиск кратчайших путей от всех вершин сети до некоторой вершины N, принятой за конечную.

Здесь решается так называемое функциональное уравнение

$$f_i = \min_{(ij)}\left[L_{ij} + f_j\right],\ \ i \neq N;\ \ f_N = 0,$$

где L_{ij} – длина дуги (ребра) $(i\,j)$ между смежными вершинами, f_i выступает как минимальная длина пути от вершины i до вершины N [9, с. 143–144].

На начальном этапе выбираются завышенные начальные оценки $f_i^{(0)} \to \infty$ для $i \neq N$, а затем реализуется процесс последовательных приближений (итераций):

$$f_i^{(k+1)} = \min_{(ij)}\left[L_{ij} + f_j^{(k)}\right],\ \ i \neq N;\ \ f_N^{(k+1)} = 0;\ k = 0, 1, 2, ...,$$

продолжающийся до совпадения очередных оценок при всех i.

Если к тому же запоминать для каждого i индекс последующей вершины j, обеспечивающей минимум, то можно будет найти искомые кратчайшие пути.

Так, для сети, представленной на рис. 36, задав начальные приближения и перебирая вершины в порядке роста индексов, получаем очередные приближения до совпадения 4 и 5-й оценок:

$$F_6^{(1)} = \min(1 + F_5^{(1)},\ 4 + F_8^{(0)},\ 9 + F_9^{(0)}) =$$

$$= \min(1 + 5, 4 + \infty, 9 + 0) = 6\ , j = 5\ ; \ldots$$

$$F_2^{(2)} = \min(3 + F_1^{(2)},\ 6 + F_3^{(1)},\ 5 + F_5^{(1)},\ 3 + F_8^{(1)}) =$$

$$= \min(\infty, \infty, 5 + 5, 3 + 10) = 10\ , ...$$

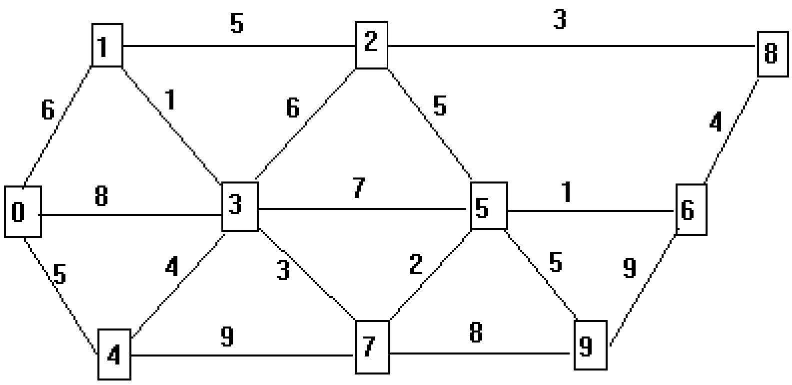


Рис. 36. Транспортная сеть

Из приведенной здесь таблицы по значениям j^5 легко выяснить кратчайший путь [0 – 1 – 3 – 7 – 5 – 9] с длиной $F_0 = 17$. Аналогично можно найти кратчайшие пути от любой вершины до 9-й.

Таблица 5. Поиск кратчайшего пути

i	F^0	F^1	j^1	F^2	j^2	F^3	j^3	F^4	j^4	F^5	j^5
0	∞	∞	4	∞	4	18	3	17	3	17	3
1	∞	∞	3	∞	3	11	3	11	3	11	3
2	∞	∞	8	10	5	10	5	10	5	10	5
3	∞	∞	1	10	7	10	7	10	7	10	7
4	∞	∞	3	14	3	14	3	14	3	14	3
5	∞	5	9	5	9	5	9	5	9	5	9
6	∞	6	5	6	5	6	5	6	5	6	5
7	∞	7	5	7	5	7	5	7	5	7	5
8	∞	10	6	10	6	10	6	10	6	10	6
9	0	0	–	0	–	0	–	0	–	0	–

5.3. Сетевое планирование

Середина XX столетия знаменовалась усложнением управления различными техническими и научными разработками, собирающими воедино десятки или сотни конструкторских бюро, заводов, поставщиков и т. п.

Соответственно, возникла необходимость создания системы, обеспечивающей возможность оценки текущего состояния и предсказания последующего хода разработки. Результатом исследований в этом направлении явилось создание систем, базирующихся на так называемых сетевых графиках.

5.3.1. Понятие о сетевом графике

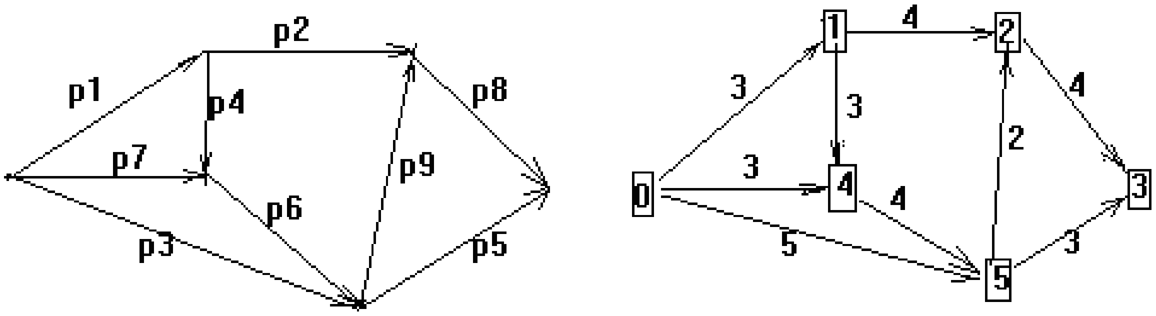
В терминологии теории графов *сетевым* графиком называют конечный ориентированный граф без контуров, в котором имеются единственная вершина с отсутствующими прообразами и единственная вершина, не имеющая образов.

Другими словами, *сетевым графиком* можно назвать ориентированную транспортную сеть с одним входом и одним выходом, в которой нет путей с повторяющимися вершинами. Дуги (стрелки) указанного графа понимаются как некоторые *работы* (табл. 6). Вершины графа называются *событиями*.

Таблица 6. Описание сетевого графика проекта

Работа	**Последующие работы**	**Продолжительность**	**Работа**	**Последующие работы**	**Продолжительность**
1	2, 4	3	6	5, 9	4
2	8	4	7	6	3
3	5, 9	5	8	–	4
4	6	3	9	8	2
5	–	3			

Информация о сетевом графике некоторого проекта может задаваться в виде рисунка или различных списков. Если информация дана списком, то ее можно представить в графическом виде (рис. 37, *a*). Если ввести обозначения или нумерацию для вершин (событий), то можно использовать и графическое представление (рис. 37, *b*), в котором числа на дугах определяют продолжительность работ.



a *b*

Работа	Продолжит.	Работа	Продолжит.
0–1	3	4–5	4
1–2	4	0–4	3
0–5	5	2–3	4
1–4	3	5–2	2
5–3	3		

c

Событие	0	1	2	3	4	5
0		3			3	5
1			4		3	
2				4		
3						
4						4
5			2	3		

d

Рис. 37. Представление информации о сетевом графике

При наличии нумерации вершин информацию о графике можно представить также перечнем работ с указанием начального и конечного событий и продолжительностей работ (рис. 37, *c*).

Можно ту же самую информацию представить и в виде матрицы смежности событий (элементы матрицы равны продолжительностям соответствующих работ или не указаны при отсутствии таких работ) (рис. 37, *d*).

Заметим, что составитель сетевого графика может использовать так называемые *фиктивные работы* – работы с нулевой продолжительностью, изображаемые пунктиром и служащие для указания порядка следования работ. Например, при наличии двух выходов их можно связать фиктивной работой. Могут использоваться и *фиктивные события*. Так если обнаружится, что пара событий связана более чем одной работой (имеем дело с мультиграфом), то можно воспользоваться фиктивными событиями и работами (рис. 38).

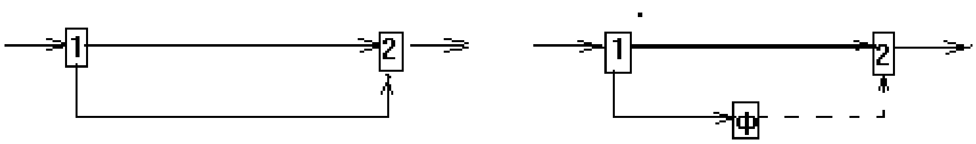


Рис. 38. Сетевой график с фиктивным событием и работой

Заметьте, что вышеуказанное требование отсутствия контуров не случайно, т. к. наличие таковых создавало бы возможность возврата к повторению ранее выполненных работ.

При обработке сетевых графиков на объем вычислительных работ существенно влияет порядок просмотра (нумерация) событий. Поэтому построение порядковой функции является значимым этапом обработки графика. Как показано в 4.6.1, оно связано с определением ранга событий. Начальному событию (входу) сопоставляется ранг 0. Ранг 1 получают события, в которые приводят работы, начинающиеся только в событии ранга 0. Ранг 2 получают события, в которые приводят работы, начинающиеся только в событиях ранга 0 и 1, и т. д. Последующая нумерация ведется в соответствии с рангами по возрастанию номеров (при одинаковом ранге порядок произволен; нумерация не обязательно сплошная, лишь бы для работы $[i - j]$ соблюдалось условие $i < j$).

Для приведенного выше примера (рис. 37) ранг 0 получает событие 0, ранг 1 – событие 1, ранг 2 – событие 4, ранг 3 – событие 5, ранг 4 – событие 2 и ранг 5 – событие 3.

5.3.2. Критический путь и другие параметры сетевого графика

Поскольку в сетевом графике продолжительность работы выступает как длина соответствующей дуги, то путь максимальной длины от входа до выхода графика (от момента начала работ до момента завершения) называют *критическим*. Длина этого пути определяет критическое время выполнения проекта, т. е. то *минимальное время*, за которое коллектив исполнителей в состоянии выполнить весь комплекс работ сетевого графика. Выше мы искали пути (цепи) кратчайшей длины, здесь в условиях полной ориентированности графа и отсутствия контуров нас интересуют пути максимальной длины.

Пусть для определенности начальное событие имеет номер 0, конечное – номер N, длины дуг обозначены как T_{ij}. Обозначим через L_j длину пути наибольшей протяженности от события 0 до события j. Основываясь на принципе оптимальности Р. Беллмана*, как бы обращенном в прошлое, получаем систему функциональных уравнений

$$L_j = \max_{(i\,j)} [\, L_i + T_{ij} \,],\ j > 0;\ L_0 = 0.$$

Здесь величина L_j соответствует *наиболее раннему* возможному времени $T_j^{\text{ран}}$ наступления j-го события, т. е. самому раннему сроку завершения всех работ, предшествующих этому событию. Значение L_N соответствует критическому времени выполнения проекта $T_{\text{крит}}$.

Обозначим через M_i длину пути наибольшей протяженности от события i до события N. Тогда по тому же принципу оптимальности

$$M_i = \max_{(i\,k)} [\, T_{ik} + M_k \,],\ i < N;\ L_N = 0.$$

* Принцип Беллмана составляет базу метода динамического программирования – одного из интереснейших методов исследования операций [9], с которым читатель сможет ознакомиться и позднее.

Величина $T_i^{\text{позд}} = T_{\text{крит}} - M_i$ соответствует *наиболее позднему* допустимому времени наступления i-го события, т. е. самому позднему сроку начала всех работ, последующих за этим событием. Совершенно очевидно, что для событий на критическом пути самое раннее и самое позднее времена их наступления будут совпадать.

Для примера рассмотрим сетевой график (рис. 39).

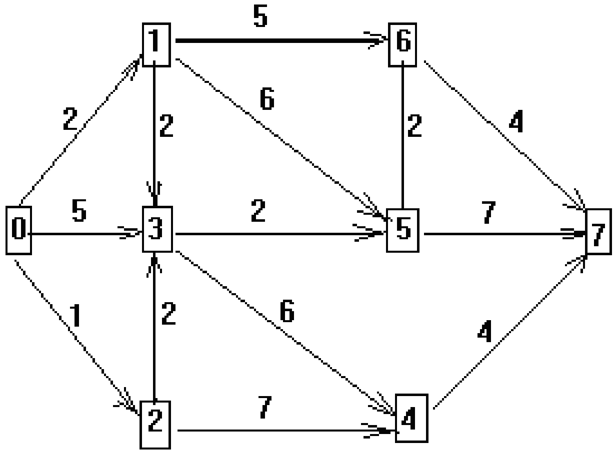


Рис. 39. Сетевой график

Составив для удобства таблицу продолжительностей работ, рассчитываем значения L_j в порядке роста номеров: $L_0 = 0$; $L_1 = 2$ ($i = 0$); $L_2 = 1$ ($i = 0$); $L_3 = \max[L_0 + 5, L_1 + 2, L_2 + 2] = 5$ ($i = 0$) и т. д.

Таблица 7. Продолжительности работ

Работа (i, j)	T_{ij}	Работа (i, j)	T_{ij}
0 – 1	2	2 – 4	7
0 – 2	1	3 – 4	6
0 – 3	5	3 – 5	2
1 – 3	2	4 – 7	4
1 – 5	6	5 – 6	2
1 – 6	5	5 – 7	7
2 – 3	2	6 – 7	4

Затем рассчитываем значения M_i в порядке убывания номеров: $M_7 = 0$; $M_6 = 4$ ($k = 7$); $M_5 = \max[2 + M_6, 7 + M_7] = 7$ ($k = 7$); $M_4 = 4$ ($k = 7$); $M_3 = \max[6 + M_4, 2 + M_5] = 10$ ($k = 4$) и т. д.

В итоге имеем сводную информацию о наиболее ранних и наиболее поздних моментах наступления событий и индексы предшествующих и последующих событий в путях наибольшей длины, проходящих через данное событие (табл. 8).

Таблица 8. Сводная информация

Событие *j*	$L_j = T_j^{\text{ранн}}$	***I* пред.**	M_j	***k* посл.**	$Tj^{\text{позд}}$
0	0	–	15	1,3	0
1	2	0	13	5	2
2	1	0	12	3	3
3	5	0	10	4	5
4	11	3	4	7	11
5	8	1	7	7	8
6	10	5	4	7	11
7	15	4,5	0	–	15

По информации из колонок 3 или 5 можно выявить критические пути с длиной 15: [0 – 1 – 5 – 7] и [0 – 3 – 4 –7].

Очевидно, что работы, не лежащие на критических путях, обладают *резервами* времени – их выполнение при некоторых оговорках может быть задержано на какое-то время.

Существуют 4 вида резервов:

– *полный резерв* $R_{ij} = T_j^{\text{позд}} - T_i^{\text{ран}} - T_{ij}$;

– *свободный резерв* $R_{ij} = T_j^{\text{ран}} - T_i^{\text{ран}} - T_{ij}$;

– *независимый резерв* $R_{ij} = \max[T_j^{\text{ран}} - T_i^{\text{позд}} - T_{ij}, 0]$;

– *частный резерв* $R_{ij} = T_j^{\text{позд}} - T_i^{\text{позд}} - T_{ij}$.

Так, полный резерв работы можно понимать как время, на которое можно замедлить выполнение работы, если предшествующие работы завершатся к самому раннему возможному сроку, но комплекс последующих работ начнется в самый последний приемлемый момент.

Независимый резерв предполагает весьма жесткую гипотезу – завершение предшествующих работ к самому позднему сроку, но начало последующих – в самый ранний возможный срок.

Все итоги обработки сетевого графика можно представить таблицей (табл. 9).

Таблица 9. Итоги обработки сетевого графика

Работа	Продолжит.	Раннее время		Позднее время		Резервы			
		начала	конца	начала	конца	полн.	своб.	незав.	част.
0–1	2	0	2	0	2	0	0	0	0
0–2	1	0	1	2	3	2	0	0	2
0-3	5	0	5	0	5	0	0	0	0
1–3	2	2	4	3	5	1	1	1	1
1–5	6	2	8	2	8	0	0	0	0
1–6	5	2	7	6	11	4	3	3	4
2–3	2	1	3	3	5	2	2	0	0
2–4	7	1	8	4	11	3	3	1	1
3–4	6	5	11	5	11	0	0	0	0
3–5	2	5	7	6	8	1	1	1	1
4–7	4	11	15	11	15	0	0	0	0
5–6	2	8	10	9	11	1	0	0	1
5–7	7	8	15	8	15	0	0	0	0
6–7	4	10	14	11	15	1	1	0	0

Здесь мы познакомились с азбукой сетевого планирования. Для реального проекта строят диаграмму распределения работ во времени (рис. 40). Эта диаграмма позволяет видеть распределение работ во времени и необходимое число исполнителей.

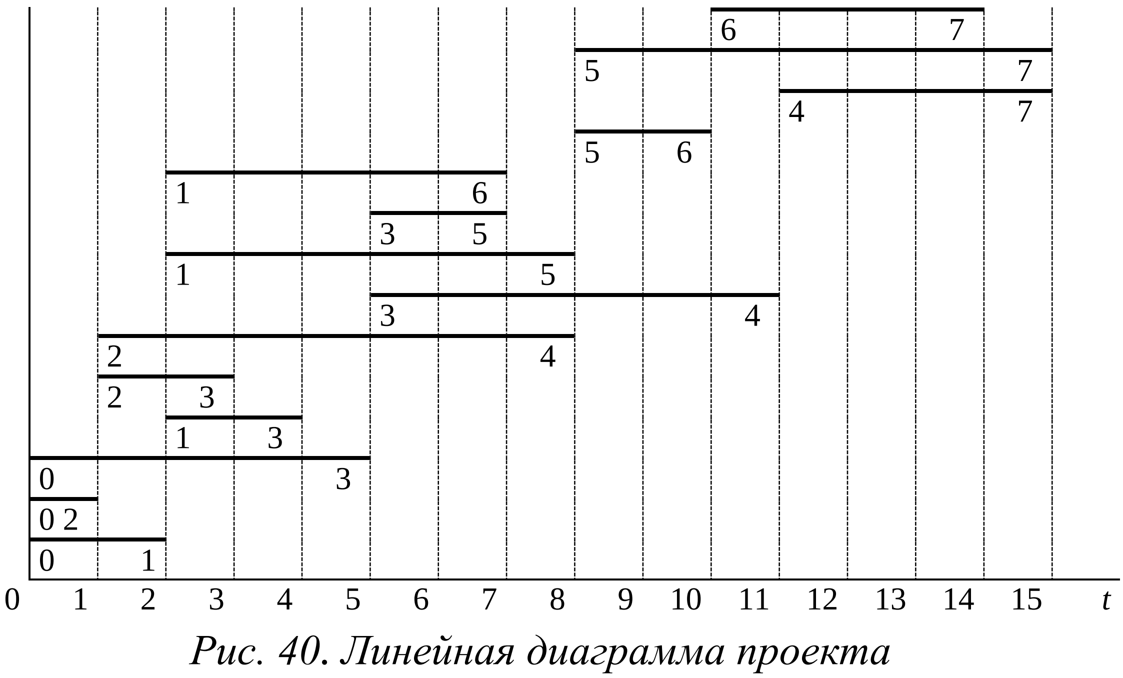


Рис. 40. Линейная диаграмма проекта

Если присутствует зависимость стоимости выполнения работы от времени, то закономерно возникают задачи минимизации стоимости проекта при его заданной продолжительности.

Методы теории графов могут использоваться и при решении задачи поиска максимального потока в транспортной сети с ограниченными пропускными способностями (каким может быть максимальное давление в водопроводной сети города, чтобы не лопались трубы; сколько газа в сутки можно перекачать из России в Европу при имеющейся сети трубопроводов и т. д.) [9, с. 75–79].

5.4. Кратчайшая транспортная сеть

Можно упомянуть и любопытную для приложений задачу построения *кратчайшей связывающей сети* [11]. Сохраняя терминологию автора, будем называть *изолированным полюсом* вершину, которая на данном этапе построения не связана с другими вершинами (полюсами), а фрагментом – подмножество полюсов, связанных звеньями (ребрами). За расстояние между фрагментами принимаем минимум расстояний между полюсами (элементами) этих фрагментов.

Принципы построения искомой сети определены следующими правилами:

1) всякий изолированный полюс соединять с ближайшим соседом;

2) всякий изолированный фрагмент соединять с ближайшим соседом.

Так, для примера на рис. 41 начнем с полюса 1. Перенесем его строку расстояний в рабочую таблицу, выбирая минимум и соответственно создавая фрагмент – ребро (1, 4).

2	3	**4**	5	6
6.7	5.2	**2.8**	5.6	3.6
1	1	**1**	1	1

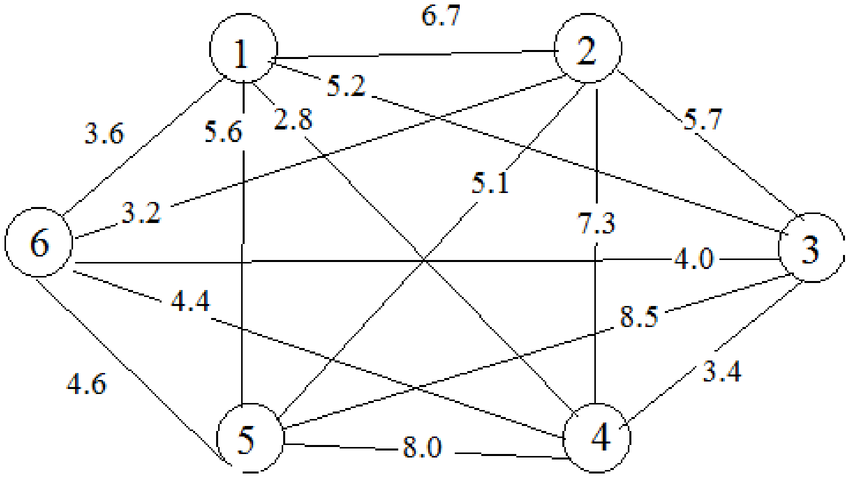
Вершина 4 наименее удалена от 3 (3.4), тогда как удаление 1 от 3 равно 5.2: соответственно к фрагменту (1 – 4) подключаем ребро (4, 3), корректируя рабочую таблицу к виду

2	**3**	5	6
6.7	**3.4**	5.6	3.6
1	**4**	1	1

Теперь для фрагмента (1 – 3 – 4) ищем ближайшего соседа среди оставшихся изолированных полюсов, расширяем фрагмент на ребро (1, 6)

2	5	6
5.7	5.6	3.6
3	1	1

Аналогичные действия предпринимаются при подключении ребер (6, 2) и (6, 1).



	1	**2**	**3**	**4**	**5**	**6**
1		6.7	5.2	2.8	5.6	3.6
2	6.7		5.7	7.5	5.1	3.2
3	5.2	5.7		3.4	8.5	4.0
4	2.8	7.5	3.4		8.0	4.4
5	5.6	5.1	8.5	8.0		4.6
6	3.6	3.2	4.0	4.4	4.6	

Итог

Звено	**Длина**
1 – 4	2.8
4 – 3	3.4
1 – 6	3.6
6 – 2	3.2
6 – 5	4.6

Рис. 41. Кратчайшая связывающая сеть

Решение многих задач теории графов (особенно достаточно большой размерности) удается получать их сведением к поиску целочисленных решений задач линейного программирования, для которых существуют численные процедуры, эффективные в сравнении с полным перебором вариантов, но не столь быстродействующие, как хотелось бы.

5.5. Задача коммивояжера

Эта популярная задача предполагает наличие N городов с известными расстояниями между ними. Коммивояжер должен посетить все N городов по одному разу, вернувшись в исходный город. В роли коммивояжера могут выступать как лидер некоторой политической партии в своей агитационной поездке, так и менеджер по продаже патентованного средства для выращивания волос, рискующий потерей своих волос при вторичном появлении в одном городе. Требуется «самая малость»: найти такой маршрут, при котором суммарное пройденное расстояние будет минимальным.

В терминах теории графов, задача коммивояжера определяет поиск «*кратчайшего*» *гамильтонова цикла в полном графе*.

Очевидно, что фиксация начальной вершины (города) связана с N выборами, на последующих этапах число выборов уменьшается от $N - 1$ до 1. Соответственно, достаточно рассмотреть все $N!$ возможных перестановок вершин, подсчитать для каждой перестановки длину маршрута и выбрать из них кратчайший. К тому же программист, знакомый с рекурсивными процедурами, имеет возможность написать короткую изящную программу.

Увы, как мы видели выше, $N!$ даже для сравнительно небольших N слишком велико ($10! = 3628800 >> 2^{10} = 1024$), и расчетное время работы программы сравнимо с продолжительностью кругосветного путешествия.

Мало что изменится и при отказе от требования возврата в исходную вершину.

Эта задача, интересная с точки зрения ее приложений в информатике, экономике, химии и биологии, принадлежит ко множеству так называемых NP-полных задач, для которых не найдены (и едва ли будут найдены) алгоритмы решения, имеющие полиномиальную (N^k) трудоемкость.

В 1950-е годы (период особого внимания к методам исследования операций и, в частности, к проблемам оптимизации транспортировки грузов) задача коммивояжера была сформулирована как задача дискретной оптимизации – задача целочисленного линейного программирования, где каждому ребру графа сопоставлена переменная x_{ij}, принимающая значение 1 при включении ребра в искомый маршрут и 0 при отказе от такового. Более того, Д. Данциг и его коллеги успешно использовали метод Р. Гомори (последовательных отсечений) в 1954 году для поиска пути коммивояжера в задаче с 49 городами. Хотя сегодня получены решения даже для N порядка десятков тысяч, решение задачи коммивояжера в повседневно меняющейся обстановке по затратам времени остается проблематичным.

Задача коммивояжера более полувека служит подопытным кроликом для апробации эффективности эвристических методов оптимизации – от метода ветвей и границ до генетических алгоритмов.

5.6. Задача Джонсона

Рассмотрим ситуацию *последовательной* обработки на двух машинах N различных деталей, если известно время A_i и B_i обработки i-й детали на первой и второй машинах [9]. Очевидно, что для первой машины порядок поступления деталей не имеет значения, так как она готова к работе сразу после обработки очередной детали и будет загружена полностью в течение времени $\sum_{i=1}^{N} A_i$. Вторая же может простаивать в ожидании конца обработки детали на первой машине. Хотелось бы найти порядок подачи деталей на обработку, минимизирующий время ее простоя и, тем самым, общее время обработки. Эта задача называется задачей Джонсона или задачей планирования производственной линии.

В сущности, здесь мы имеем дело с полным графом с N вершинами, где ребра определяют возможность перехода между ними, и выбором из $N!$ гамильтоновых путей. Обозначив через X_i простой второй машины в ожидании i-й детали, имеем

$X_1 = A_1$;

$X_1 + X_2 = \max(A_1 + A_2 - B_1, A_1)$;

$$X_1 + X_2 + X_3 = \max(A_1 + A_2 + A_3 - B_1 - B_2, A_1 + A_2 - B_1, A_1);$$

$$\sum_{i=1}^{N} X_i = \max_{1 \le k \le N} \left(\sum_{i=1}^{k} A_i - \sum_{i=1}^{k-1} B_i \right).$$

Можно показать, что условие необходимости перестановки в паре деталей (i, j) определяется отношением

$$\min(A_j, B_i) < \min(A_i, B_j).$$

Соответственно, среди всех значений A_i и B_i ищем наименьшее. Если найденное значение совпадает с некоторым A_i, то i-ю деталь ставим на обработку первой; если оно совпадает с некоторым B_i, то последней. Эту процедуру повторяем для всех остальных деталей.

Пусть, например, время обработки деталей задано таблицей

i	1	2	3	4	5	6	7	8
A_i	4	4	30	6	2	9	13	9
B_i	5	1	4	30	3	13	9	9

Минимальное из всех значений времен обработки равно 1 и соответствует B_2 – вторая деталь обрабатывается последней. Минимальное из оставшихся значений (без учета второго столбца) соответствует A_5 – пятая деталь обрабатывается первой. Минимальное из значений в столбцах, кроме 2 и 5, равно A_1 и, соответственно, среди рассматриваемых сейчас деталей эта деталь обрабатывается первой и т. д. В итоге такого упорядочения получаем

i	5	1	4	8	6	7	3	2
A_i	2	4	6	9	9	13	30	4
B_i	3	5	30	9	13	9	4	1

Время простоя второй машины при первичном порядке равно $\max(4, 4 + 4 - 5, 4 + 4 + 30 - 5 - 1, 4 + 4 + 30 + 6 - 5 - 1 - 4, 4 + 4 + 30 + 6 + 2 - 5 - 1 - 4 - 30, 4 + 4 + 30 + 6 + 2 + 9 - 5 - 1 - 4 - 30 - 3, 4 + 4 + 30 + 6 + 2 + 9 + 13 - 5 - 1 - 4 - 30 - 3 - 13, 4 + 4 + 30 + 6 + 2 + 9 + 13 - 5 - 1 - 4 - 30 - 3 - 13) = \max(4, 3, 32, 34, 6, 12, 12, 12) = 34$.

Простой при оптимальной же перестановке составит $\max(2, 2 + 4 - 3, 2 + 4 + 6 - 3 - 5, 2 + 4 + 6 + 9 - 3 - 5 - 30, 2 + 4 + 6 + 9 + 9 - 3 - 5 - 30 - 9, 2 + 4 + 6 + 9 + 9 + 13 - 3 - 5 - 30 - 9 - 13, 2 + 4 +$

$+ 6 + 9 + 9 + 13 + 30 - 3 - 5 - 30 - 9 - 13 - 9, 2 + 4 + 6 + 9 + 9 + 13 +$
$+ 30 + 4 - 3 - 5 - 30 - 9 - 13 - 9 - 4) = \max(2, 3, 4, -17, -17, -17, 4, 4) =$
$= 4.$

В процессе решения можно было заметить, что существует и другой оптимальный порядок обработки, связанный с неоднозначностью установки детали 8.

К сожалению, столь простого решения задачи Джонсона для случая последовательной обработки на более чем двух машинах не найдено.

ЦИТИРОВАННАЯ И РЕКОМЕНДУЕМАЯ ЛИТЕРАТУРА

1. Берж, К. Теория графов и ее применения. – Москва : Издательство иностранной литературы, 1962. – 320 с.

2. Кемени, Дж. Введение в конечную математику / Дж. Кемени, Дж. Снелл, Дж. Томпсон. – Москва : Мир, 1965. – 484 с.

3. Иглин, С. П. Математические расчеты на базе Matlab. – Санкт-Петербург : БХВ-Петербург, 2005. – 640 с.

4. Харари, Ф. Теория графов. – Москва : Мир, 1973. – 301 с.

5. Зыков, А. А. Теория конечных графов. – Новосибирск : Наука, Сибирское отделение, 1969. – 554 с.

6. Басакер, Р. Конечные графы и сети / Р. Басакер, Т. Саати. – Москва : Наука, 1974. – 368 с.

7. Альсина, Клауди. Мир математики Т. 11. Карты метро и нейронные сети. Теория графов. – Москва : Де Агостини, 2014. – 144 с.

8. Виленкин, Н. Я. Популярная комбинаторика. – Москва : Наука, 1975. – 208 с.

9. Тынкевич, М. А. Экономико-математические методы (Исследование операций) : учеб. пособие. – 3-е изд., испр. и доп. – Кемерово, 2011. – 222 с.

10. Тынкевич, М. А. Численные методы анализа / М. А. Тынкевич, А. А. Тайлакова. – Кемерово, 2012. – [Электронный ресурс]. – Режим доступа: http://library.kuzstu.ru/meto.php?n=90591&type=utchposob:common, свободный.

11. Прим, Р. К. Кратчайшие связывающие сети и некоторые обобщения // Кибернетический сборник, 2. – Москва : Издательство иностранной литературы, 1961. – С. 95–107.

12. Столл, Роберт Р. Множества. Логика. Аксиоматическая теория. – Москва : Просвещение, 1968. – 231 с.

13. Риордан, Дж. Введение в комбинаторный анализ. – Москва : Издательство иностранной литературы, 1963. – 287 с.

14. Хуанхо, Руэ. Мир математики. Т. 34. Искусство подсчета. Комбинаторика и перечисление. – Москва : Де Агостини, 2014. – 146 с.

15. Математический энциклопедический словарь / под ред. Ю. В. Прохорова. – Москва : Советская энциклопедия, 1988. – 847 с.

16. Гомес, Жуан. Мир математики. Т. 2. Математики, шпионы и хакеры. Кодирование и криптография. – Москва : Де Агостини, 2014. – 144 с.

17. Тынкевич, М. А. Исследование операций и имитационное моделирование : учеб. пособие / М. А. Тынкевич, А. Г. Пимонов, С. А. Веревкин. – Кемерово, 2015. – 248 с.

18. Берцун, В. Н. Математическое моделирование на графах. Ч. 1 : учеб. пособие. – Томск : Издательство ИТЛ, 2006. – 88 с.

19. Липский, В. Комбинаторика для программистов. – Москва : Мир, 1988. – 200 с.

20. Гуц, А. К. Математическая логика и теория алгоритмов : учеб. пособие. – Омск : Издательство Наследие. Диалог-Сибирь, 2003. – 108 с.

21. Лавров, И. А. Задачи по теории множеств, математической логике и теории алгоритмов / И. А. Лавров, Л. Л. Максимова. – 5-е изд., исправл. – Москва : ФИЗМАТЛИТ, 2004. – 256 с.

22. Тюрин, С. Ф. Дискретная математика и математическая логика : Практическая дискретная математика и математическая логика: учеб. пособие / С. Ф. Тюрин, Ю. А. Аляев. – Москва : Финансы и статистика, 2010. – 384 с.

23. Хаггарти, Р. Дискретная математика для программистов. – Москва : Техносфера, 2003. – 320 с.

24. Шевелев, Ю. П. Дискретная математика : учеб. пособие. – Санкт-Петербург : Лань, 2016. – 592 с.

25. Вороненко, А. А. Дискретная математика. Задачи и упражнения с решениями : учебно-методическое пособие. – Москва : ИНФРА-М, 2016. – 104 с.

26. Дехтярь, М. И. Основы дискретной математики. – Москва : Национальный Открытый Университет «ИНТУИТ», 2016. – 184 с.

27. Таланов, А. В. Графы и алгоритмы / А. В. Таланов, В. Е. Алексеев. – Москва : Национальный Открытый Университет «ИНТУИТ», 2016. – 154 с.

Тынкевич Моисей Аронович
Пимонов Александр Григорьевич
Прокопенко Евгения Викторовна

Введение в дискретную математику
(от теории к практике)

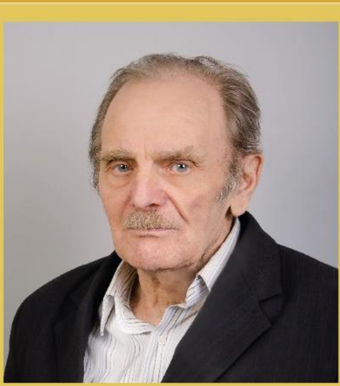
Учебное пособие

Редактор З. М. Савина

Подписано в печать 14.10.2016. Формат 60×84/16
Бумага офсетная. Гарнитура «Times New Roman». Уч.-изд. л. 6,50
Тираж 500 экз. Заказ №

КузГТУ, 650000, Кемерово, ул. Весенняя, 28

Издательский центр УИП КузГТУ, 650000, Кемерово, ул. Д. Бедного, 4А



Тынкевич Моисей Аронович – родился 28 февраля 1937 г. в г. Новосибирске, кандидат физико-математических наук, доцент, профессор кафедры прикладных информационных технологий. В 1959 году окончил механико-математический факультет Томского государственного университета в группе 24 первых за Уралом выпускников по новой специальности «Вычислительная математика». В 1959 – 1966 гг. работал на кафедре вычислительной математики Томского университета. С 1966 г. старший преподаватель кафедры экономики и организации производства Кузбасского политехнического института, с 1969 г. старший преподаватель новой в ВУЗе кафедры вычислительной техники и промэлектроники. Внес значительный вклад в становление и развитие кафедры прикладных информационных технологий. Подготовил более 80 научных работ и учебных пособий, несколько циклов методических разработок. Почетный работник высшего образования России (1997 г.). Награжден медалями «За особый вклад в развитие Кузбасса» (2001 г.), «За достойное воспитание детей» (2010 г.). Ведет занятия по курсам «Численные методы анализа», «Исследование операций и методы оптимизации», «Экономико-математические методы и модели», «Статистический анализ данных».

Пимонов Александр Григорьевич – родился 23 ноября 1959 г. в селе Чапаево Хакасской автономной области, профессор, доктор технических наук, заведующий кафедрой прикладных информационных технологий. В 1981 г. с отличием окончил факультет прикладной математики и кибернетики Томского государственного университета. В Кузбасском государственном техническом университете работает с 1985 г. (старший инженер, старший преподаватель, доцент, профессор, заместитель заведующего кафедрой, заведующий кафедрой, исполняющий обязанности декана факультета информационных технологий и менеджмента). Подготовил более 180 научных работ и учебно-методических разработок. Почетный работник высшего профессионального образования Российской Федерации (2010 г.), лучший профессор КузГТУ (2014 г.), лучший руководитель научно-исследовательской работы студентов КузГТУ (2014 г.), научный руководитель магистерской программы по направлению подготовки «Прикладная информатика» и программы подготовки аспирантов по направлению «Информатика и вычислительная техника». Ведет занятия по дисциплинам «Теория систем и системный анализ», «Математические и инструментальные методы поддержки принятия решений», «Математическое и имитационное моделирование».

Прокопенко Евгения Викторовна – родилась 9 февраля 1982 г. в г. Березовском Кемеровской области, доцент, кандидат физико-математических наук, доцент кафедры прикладных информационных технологий, начальник учебно-методического управления Кузбасского государственного технического университета имени Т. Ф. Горбачева. В 2004 году окончила математический факультет Кемеровского государственного университета. В Кузбасском государственном техническом университете работает с 2010 г. Подготовила более 150 научных работ и учебно-методических разработок. Лучший доцент КузГТУ (2013–2014 гг.). Ведет занятия по дисциплинам «Информатика», «Информационные технологии», «Дискретная математика», «Математика», «Теория вероятностей и математическая статистика».

ISBN 978-5-906805-97-3