Содержание

Введение 3

1 Теория графов 5

1.1 Понятие и терминология теории графов 5

1.2 Некоторые задачи теории графов 6

2 Математическая логика и теория типов 25

Заключение 27

Список использованной литературы 30

# Введение

В широком смысле информа́тика (ср. со сходными по звучанию и происхождению нем. Informatik и фр. Informatique, в противоположность традиционному англоязычному термину англ. computer science — наука о компьютерах - в США или англ. computing science — вычислительная наука -в Британии есть наука о вычислениях, хранении и обработке информации. Она включает дисциплины, так или иначе относящиеся к вычислительным машинам: как абстрактные, вроде анализа алгоритмов, так и довольно конкретные, например, разработка языков программирования.

Согласно тезису Чёрча — Тьюринга, все известные типы вычислительных машин качественно эквивалентны в своих возможностях: любое действие, выполнимое на одной вычислительной машине, также выполнимо и на другой. Тезис иногда преподносят как фундаментальный принцип информатики, обращая особое внимание на машину Тьюринга и машину фон-неймановской архитектуры, поскольку они имеют явное сходство с большинством из ныне действующих компьютеров. В рамках современной информатики учёные изучают также и другие типы машин, не только практически осуществимые (такие, как параллельные и квантовые компьютеры), но и сугубо абстрактные математические модели (к примеру, машина случайного доступа, которая имеет бесконечное число регистров).

Темами исследований в информатике являются вопросы: что можно, а что нельзя реализовать в программах (теория вычислимости и искусственный интеллект), каким образом можно решать специфические задачи с максимальной эффективностью (алгоритмы), в каком виде следует хранить и восстанавливать информацию специфического вида (структуры данных), как программы и люди должны взаимодействовать друг с другом (пользовательский интерфейс и языки программирования) и т. п.

Отдельной наукой информатика была признана лишь в 1970-х; до этого она развивалась в составе математики, электроники и других технических наук. Некоторые начала информатики можно обнаружить даже в лингвистике. С момента своего признания отдельной наукой информатика разработала собственные методы и терминологию.

Первый факультет информатики был основан в 1962 году в университете Пердью (Purdue University). Сегодня факультеты и кафедры информатики имеются в большинстве университетов мира.

Высшей наградой за заслуги в области информатики является премия Тьюринга.

# 1 Теория графов

1.1 Понятие и терминология теории графов

Тео́рия гра́фов — раздел дискретной математики, изучающий свойства графов. В наиобщем смысле граф представляется как множество вершин (узлов), соединённых рёбрами. В строгом определении графом называется такая пара множеств

**G={R,V},**

где V есть подмножество любого счётного множества,

а R - подмножество V×V.

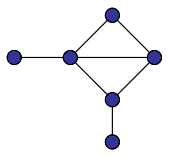


Рис. 1. Граф с шестью вершинами и семью рёбрами

Теория графов находит применение, например, в геоинформационных системах (ГИС). Существующие или вновь проектируемые дома, сооружения, кварталы и т. п. рассматриваются как вершины, а соединяющие их дороги, инженерные сети, линии электропередач и т. п. — как рёбра. Применение различных вычислений, производимых на таком графе, позволяет, например, найти кратчайший объездной путь или ближайший продуктовый магазин, спланировать оптимальный маршрут (см. Рис. 1).

Терминология теории графов поныне не определена строго. В частности в монографии Гудман, Хидетниеми, 1981 сказано «В программистском мире нет единого мнения о том, какой из двух терминов "граф" или "сеть". Мы выбрали термин "сеть", так как он, по-видимому, чаще встречается в прикладных областях»[[1]](#footnote-1).

1.2 Некоторые задачи теории графов

\* Семь мостов Кёнигсберга — один из первых результатов в теории графов, опубликован Эйлером в 1736.

\* Проблема четырёх красок — была сформулирована в 1852 году, но доказательство получено лишь в 1976 году (достаточно 4-х красок для карты на сфере (плоскости)).

\* Задача коммивояжёра — одна из наиболее известных NP-полных задач.

\* Задача о клике — ещё одна NP-полная задача.

\* Нахождение минимального стягивающего дерева.

**Задача коммивояжёра** заключается в отыскании самого выгодного маршрута, проходящего через указанные города хотя бы по одному разу. В условиях задачи указываются критерий выгодности маршрута (кратчайший, самый дешёвый, совокупный критерий и т. п.) и соответствующие матрицы расстояний, стоимости и т. п. Как правило указывается, что маршрут должен проходить через каждый город только один раз, в таком случае выбор осуществляется среди гамильтоновых циклов.

Существует масса разновидностей обобщённой постановки задачи, в частности геометрическая задача коммивояжёра (когда матрица расстояний отражает расстояния между точками на плоскости), треугольная задача коммивояжёра (когда на матрице стоимостей выполняется неравенство треугольника), симметричная и асимметричная задачи коммивояжёра.

Простейшие методы решения задачи коммивояжёра: полный лексический перебор, жадные алгоритмы (метод ближайшего соседа, метод включения ближайшего города, метод самого дешёвого включения), метод минимального остовного дерева. На практике применяются различные модификации более эффективных методов: метод ветвей и границ и метод генетических алгоритмов, а так же алгоритм муравьиной колонии.

Все эффективные (сокращающие полный перебор) методы решения задачи коммивояжёра — методы эвристические. В большинстве эвристических методов находится не самый эффективный маршрут, а приближённое решение. Зачастую востребованы так называемые any-time алгоритмы, то есть постепенно улучшающие некоторое текущее приближенное решение.

Задача коммивояжёра есть NP-полная задача[[2]](#footnote-2). Часто на ней проводят обкатку новых подходов к эвристическому сокращению полного перебора.

Назовём языком множество слов над алфавитом Σ. Задачей здесь является определение того, принадлежит данное слово языку или нет. Язык *L*1 называется сводимым (по Карпу) к языку *L*2, если существует функция, , вычислимая за полиномиальное время, обладающая следующим свойством: *f(x)* принадлежит *L*2 тогда и только тогда, когда *x* принадлежит *L*1. Язык *L*2 называется NP-трудным, если любой язык из класса NP сводится к нему. Язык называют NP-полным, если он NP-труден и при этом сам лежит в классе NP. Таким образом, если будет найден алгоритм, решающий хоть одну NP-полную задачу за полиномиальное время, все NP-задачи будут лежать в классе P.



Вернемся к задаче коммивояжера.

Математическая модель.

Исходные параметры модели.

Пусть i=0,1,2,...,m - номера городов, i=0 - номер выделенного города (начало и окончание маршрута). Обозначим через R=r(i,j) - (m+1)(m+1) матрицу расстояний, элемент которой r(i,j) - расстояние между городом с номером i и городом с номером j.



Варьируемые параметры модели.

Обозначим через X=x(i,j) - (m+1)(m+1) матрицу неизвестных, элемент которой x(i,j) =1, если коммивояжер из города с номером i переедет в город с номером j, x(i,j) = 0, в противном случае; u(i) - специальные переменные, i=1,2,...m.



Ограничения математической модели.

x(i,j) =1, j=1,2,...,m, (1)



x(i,j) =1, i=1,2,...,m, (2)



u(i) - u(j) + m x(i,j) m-1, i=1,2,...,m, j=1,2,...,m, ij., (3)



x(i,j) {0,1}. (4)



Здесь условия (1) означают, что коммивояжер ровно один раз въедет в каждый город (кроме города с номером 0); условия (2) означают, что коммивояжер ровно один раз выедет из каждого города (кроме города с номером 0), ограничения (3) означают существование лишь одного цикла, начинающегося в городе с номером 0, проходящего через все города и завершающегося в городе с номером 0; ограничения (4) являются естественными условиями на введенные переменные.

Покажем, что условия (3) являются необходимыми и достаточными условиями существования лишь одного цикла.

Действительно, пусть это не так и найдется подцикл с числом городов k<m, не проходящий через город с номером 0. Складывая все неравенства (3) при условиях, что x(i,j)=1 по городам подцикла, получим mk (m-1)k (все u(i) и u(j) взаимно уничтожаются), что противоречит существованию подцикла длины k<m.



С другой стороны, покажем, что для цикла, проходящего через все города, начинающегося и заканчивающегося в городе с номером 0, найдутся величины u(i), удовлетворяющие условиям (3).

Положим u(i)=p, если город с номером i будет посещен коммивояжером p-ым по порядку, p=1,2,...,m.

Пусть x(i,j) = 0. Тогда условия (3) примут вид:

u(i) - u(j) m-1, что верно, так как p<m+1 и p>0.



Пусть x(i,j) = 1. Тогда, так как если u(i) = p, то u(j)=p+1 (это следует из того, что город с номером j будет следующим в маршруте коммивояжера после города с номером i). Получим:

u(i) - u(j) + m x(i,j) = p - (p+1) +m = m - 1, что и доказывает правомочность присутствия в модели ограничений (3).

Постановка оптимизационной задачи.

Критерий оптимальности для задачи коммивояжера имеет вид:

F(X)= r(i,j) x(i,j) min . (5)



Задача (1) - (5) называется задачей коммивояжера или задачей бродячего торговца.

С помощью рассмотренной математической модели описываются следующие прикладные задачи: задача минимизации времени переналадок уникального оборудования; задача развозки готовой продукции по потребителям; задача управления работой снегоочистительных машин и др.

# Задача выполнимости булевых формул. Задача выполнимости булевых формул (SAT или ВЫП) — задача распознавания, важная для теории вычислительной сложности.

Экземпляром задачи SAT является булева формула, состоящая только из имен переменных, скобок и операций (*И*), (*ИЛИ*) и (*HE*). Задача заключается в следующем: можно ли назначить всем переменным, встречающимся в формуле, значения *ЛОЖЬ* и *ИСТИНА* так, чтобы формула стала истинной.



Согласно теореме Кука, доказанной Стивеном Куком в 1971-м году, проблема SAT NP-полна.

Чтобы четко сформулировать задачу распознавания необходимо условиться об алфавите, с помощью которого задаются экземпляры языка. Этот алфавит должен быть фиксирован и конечен. В своей книге Хопкрофт, Мотвани и Ульман предлагают использовать следующий алфавит: {«», «», «», «(», «)», «*x*», «0», «1»}.



При использовании такого алфавита скобки и операторы записываются естественным образом, а переменные получают следующие имена: x1, x10, x11, x100 и т. д., согласно их номерам, записанным в двоичной системе счисления.

Пусть некоторая булева формула, записанная в обычной математической нотации, имела длину *N* символов. В ней каждое вхождение каждой переменной было описано хотя бы одним символом, следовательно, всего в данной формуле не более *N* переменных. Значит, в предложенной выше нотации каждая переменная будет записана с помощью символов. В таком случае, вся формула в новой нотации будет иметь длину символов, то есть длина строки возрастет в полиномиальное число раз.



Например, формула примет вид .



**Семь мостов Кёнигсберга.** Семь мосто́в Кёнигсберга существовали в Кёнигсберге (нынешнем Калининграде) в XVI—XX веках (см. Рис. 2). Взаимное расположение мостов натолкнуло математика Леонарда Эйлера на размышления, приведшие к возникновению теории графов.

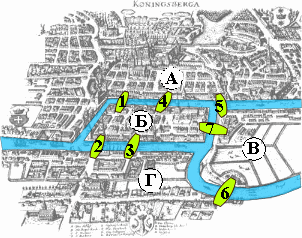


Рис. 2. Старинная карта Кёнигсберга. Буквами обозначены части города: А — Альтштадт, Б — Кнайпхоф, В — Ломзе, Г — Форштадт. Цифрами обозначены мосты (в порядке строительства): 1 — Лавочный, 2 — Зелёный, 3 — Рабочий, 4 — Кузнечный, 5 — Деревянный, 6 — Высокий, 7 — Медовый

Издавна среди жителей Кёнигсберга была распространена такая загадка: как пройти по всем мостам, не проходя ни по одному из них дважды? Многие кёнигсбержцы пытались решить эту задачу, как теоретически, так и практически, во время прогулок. Но никому это не удавалось, однако доказать, что это даже теоретически невозможно.

В 1736 году задача о семи мостах заинтересовала выдающегося математика, члена Петербургской академии наук Леонарда Эйлера, о чём он написал в письме итальянскому математику и инженеру Мариони от 13 марта 1736 года. В этом письме Эйлер пишет о том, что он смог найти правило, пользуясь которым легко определить, можно ли пройти по всем мостам, не проходя дважды ни по одному из них (в случае семи мостов Кёнигсберга это невозможно).

На упрощённой схеме части города (графе) мостам соответствуют линии (рёбра графа), а частям города — точки соединения линий (вершины графа). В ходе рассуждений Эйлер пришёл к следующим выводам:

• Число нечётных вершин (вершин, к которым ведёт нечётное число рёбер) графа всегда чётно. Невозможно начертить граф, который имел бы нечётное число нечётных вершин.

• Если все вершины графа чётные, то можно, не отрывая карандаша от бумаги, начертить граф, при этом можно начинать с любой вершины графа и завершить его в той же вершине.

• Граф с более чем двумя нечётными вершинами невозможно начертить одним росчерком.

Граф кёнигсбергских мостов имел четыре нечётные вершины, следовательно невозможно пройти по всем мостам, не проходя ни по одному из них дважды[[3]](#footnote-3).

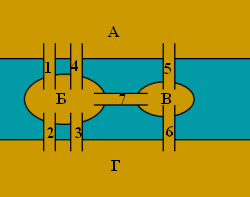


Рис. 3. Упрощённая схема мостов Кёнигсберга. Значение букв и цифр — см. Рис.2 - комментарий к старинной карте Кёнигсберга

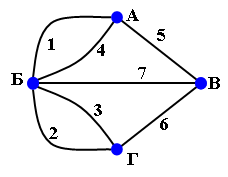


Рис. 4. Граф кёнигсбергских мостов

Созданная Эйлером теория графов нашла очень широкое применение: например, её используют при изучении транспортных и коммуникационных систем, в частности, для маршрутизации данных в Интернете.

**Проблема четырёх красок** — математическая задача, предложенная Ф. Госри (англ. Francis Guthrie) в 1852 году.

*Выяснить, можно ли всякую, расположенную на сфере карту, раскрасить четырьмя красками так, чтобы любые две области, имеющие общий участок границы в виде дуги, были раскрашены в разные цвета.*

К. Аппель и В. Хакен доказали в 1976 г., что так можно раскрасить любую карту. Это была первая крупная математическая теорема, для доказательства которой был применён компьютер. Не смотря на последующие упрощения, доказательство практически невозможно проверить не используя компьютер.

Раскрашивая географическую карту естественно пользоваться по возможности меньшим количеством цветов, однако так, чтобы две страны, имеющие общую часть границы (не только общую точку), были окрашены по-разному. В 1852 году Френсис Гутри (Guthrie), составляя карту графств Англии, обратил внимание, что для такой цели вполне хватает четырех красок. Его брат, Фредерик, сообщил об этом наблюдении известному математику О. Де Моргану (DeMorgan), а тот – математической общественности. Точная формулировка гипотезы опубликована А. Кэли (Cayley, 1878).

Первое доказательство появилось год спустя и принадлежало В. Кемпе (Kempe). Одиннадцать лет спустя П. Хивуд (Heawood) обнаружил в нем ошибку[[4]](#footnote-4). За первым ошибочным доказательством последовало множество других. В этом отношении проблема четырех красок уступала лишь знаменитой проблеме Ферма. До середины XX века, хотя проблемой четырех красок занимались многие выдающиеся математики, положение с доказательством изменилось несущественно: идеи Дж. Д. Биркгофа позволили П. Франклину в 1913 году доказать гипотезу для карты с не более чем 25 странами. Позже это число было увеличено до 38.

В 1977 году доказательство гипотезы четырех красок было наконец получено К. Аппелем и У. Хакеном (Appel, Haken) и опубликовано в двух статьях. Значительную часть рутинных проверок выполнил компьютер, и это революционное нововведение в сложившуюся практику дедуктивных рассуждений в чистой математике служит основанием для некоторого естественного скептицизма по отношению к данному доказательству и по сей день. Сначала мы приведем точные формулировки, докажем теорему о пяти красках и укажем некоторые эквивалентные проблемы.

Проблемы раскраски карты на глобусе и плоскости эквивалентны. Действительно, в случае карты на сфере можно вырезать кусок внутренней области какой-либо страны; продырявленную сферу можно деформировать (растянуть) в плоскую область – представим, что карта сделана из тонкой резины. На плоской карте отверстие превратится в "океан", омывающий со всех сторон одну страну. Разумеется, длины границ, их форма, размеры стран подвергнутся при растяжении значительным изменениям, но сетка границ останется, добавится лишь растянутая граница прорезанного отверстия, внешняя граница океана. Ее можно убрать, то есть раскрасить океан так же, как и окруженную им страну. Такие деформации стран и их границ, очевидно, не меняют задачи раскраски. Ниже рассматривается плоская карта.

Начнем с того, что заменим задачу раскраски плоской карты на эквивалентную ей проблему, касающуюся плоских графов. Выберем столицу у каждой страны (то есть выберем по одной внутренней точке в каждой из стран) и соединим дугами столицы стран, имеющих общий сегмент границы. В результате получится так называемый плоский граф.

Определение 1. Графом G называется конечное множество вершин V(G) и конечное множество ребер R(G ), так что каждое ребро имеет своими концами две различные вершины. Граф называется плоским, если вершины являются точками плоскости, а ребра – ломаными линиями (составленными из отрезков) в этой же плоскости, имеющими своими концами вершины, не пересекающимися между собой и не включающими других вершин, кроме своих концов.

Отметим, что в плоском графе не допускаются петли (ребра, имеющие началом и концом одну и ту же вершину).

Плоский граф разрезает плоскость на совокупность D(G) неперекрывающихся многоугольных областей, необязательно конечных (рис. 5).

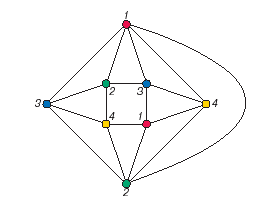


Рис. 5. 4-раскраска плоского графа

Если перенумеровать используемые краски 1, 2, ..., n, то на соответствующем карте плоском графе этими числами окажутся занумерованы вершины / столицы.

Определение 2. Правильной n-раскраской плоского графа G называется отображение f: V(G ) → {1, 2, ..., n}, причем f(u1) # f(u2) в случае, если существует такое ребро r k R(G ), что граница r состоит из u1 и u2 .

Наконец, можно сформулировать проблему четырех красок в виде следующего утверждения.

Теорема 1. Любой плоский граф допускает правильную 4-раскраску.

Вот решение этой-то проблемы и заняло более столетия. Однако на первый взгляд чуть более слабое утверждение о правильной 5-раскраске доказать достаточно просто. Для доказательства понадобится формула Эйлера, связывающая число вершин, ребер и областей. Пусть | M | обозначает число элементов конечного множества M.

Теорема Эйлера. Для любого плоского графа | V(G) | – | R(G) | + | D(G) | = 2.

Заметим, что если не учитывать внешнюю бесконечную область, то формула Эйлера для триангуляции конечного плоского графа имеет вид | V(G ) | – | R(G ) | + + | D(G ) | = 1.

Теорема 2. Любой плоский граф допускает правильную 5-раскраску.

Доказательство. Сначала упростим граф. Если есть несколько ребер, соединяющих некоторую пару вершин (такая ситуация может возникнуть, если две страны имеют несколько несвязанных между собой кусков границы, например как у России и Китая), то оставим только одно ребро: правильность раскраски такого уменьшенного графа все равно гарантирует правильную раскраску исходного.

Проведем теперь индукцию по числу вершин графа. Для графа с тремя вершинами утверждение теоремы очевидно. Пусть оно справедливо для всех графов с n – 1 вершиной.

Пусть D1 , D2 , ..., Dm – полный набор всех m = D(G ) областей графа, а N(Di) – число ребер, составляющих границу i-й области графа. При этом N(Di) ≥ 3 для любого i. Любое ребро входит в границу в точности двух областей, поэтому

N(D1) + N(D2)+ ... +N(Dm) = 2R(G ).

Вследствие неравенств N(Di) ≥ 3 имеем

2R(G ) = N(D1) + N(D2) + ... + N(Dm) ≥ 3m = 3D(G ),

откуда 2R(G ) ≥ 3D(G ).

По формуле Эйлера | V(G ) | – 2 + | D(G ) | = | R(G ) |, откуда

3 | R(G ) | = 3 | V(G ) | – 6 + 3 | D(G ) | ≤ 3 | V(G ) | – 6 + 2 | R(G ) |

и, следовательно,

| R(G ) | ≤ 3 | V(G ) | – 6.

Заметим, что удвоенное число ребер можно отождествить и с другой характеристикой графа. Пусть a1 , a2 , ... ..., an есть полный набор n = V(G ) вершин графа, а M(aj) – число ребер, сходящихся в вершине с номером j. Но каждое ребро заканчивается двумя вершинами, поэтому

2R(G) = M(a1) + M(a2) + ... + M(an).

Кроме того, как это следует из неравенства (1), 2 | R(G ) | ≤ 6 | V(G ) | – 12. Следовательно,

M(a1) + M(a2) + ... + M(an) ≤ 6 | V(G ) | – 12.

Из последнего неравенства можно вывести, что существует по крайней мере одна вершина, в которой сходится не более пяти ребер. Действительно, предположим противное, то есть "j M(aj) ≥ 6. Тогда

M(a1) + M(a2) + ... + M(an) ≥ 6n = 6V(G ),

что противоречит (2).

Перенумеруем вершины так, что в вершине a = an сходится не более пяти ребер.

Если в вершине a сходятся не более четырех ребер, то рассмотрим граф G \ a, который получается из G устранением вершины a и всех оканчивающихся в ней ребер. Граф G \ a допускает правильную 5-раскраску по предположению индукции. А так как ребра соединяют вершину a не более чем с четырьмя вершинами этого меньшего графа, то для правильной раскраски a остается по крайней мере один цвет (из пяти).

Пусть теперь в a сходится ровно пять ребер. Рассмотрим граф H ⊃ G, состоящий из тех пяти вершин, куда приходят ребра, выходящие из a и соединяющих их (в G) ребер. В графе H обязательно найдутся две вершины, не соединенные ребром. Действительно в противном случае в пятиугольнике H будет C52= 10 ребер (это нетрудно посчитать и непосредственно). Однако в силу (1)

| R(H ) | ≤ 3| V(H ) | – 6 = 3 " 5 – 6 = 9,

и мы приходим к противоречию.

Пусть b и g суть те вершины H, которые не соединены между собой. Не соединены они и в G \ a. Рассмотрим граф G ', который получается из G \ a при помощи деформации, которая отождествляет (совмещает) b и g. Граф G ' – это плоский граф, так как при отождествлении вершин в этой ситуации не может возникнуть петли. Но для G ' справедливо предположение индукции, и существует некоторая его правильная 5-раскраска j. Разъединим в этом раскрашенном графе вершины b и g. Тогда правильную 5-раскраску получает и граф G \ a, причем такую, что j(b) = j(g). Иными словами, b и g раскрашены одинаково и, следовательно, раскраска пяти соседних с a вершин графа H использует не более четырех цветов.

Используем оставшийся цвет для раскраски вершины a и получим правильную 5-раскраску G!

Проблема четырех красок кажется на первый взгляд изолированной задачей, мало связанной с другими разделами математики и практическими задачами. На самом деле это не так. Известно более 20 ее переформулировок, которые связывают эту проблему с задачами алгебры, статистической механики и задачами планирования. И это тоже характерно для математики: решение задачи, изучаемой из чистого любопытства, оказывается полезным в описании реальных и совершенно различных по своей природе объектов и процессов. Здесь мы рассмотрим одну задачу, эквивалентную проблеме четырех красок.

Пусть i, j и k суть стандартные единичные направляющие векторы координатных осей x, y и z соответственно. В трехмерном пространстве определено векторное произведение векторов, обозначаемое знаком i. Если u, u k R3 – два вектора, то их векторное произведение u i u имеет длину а направление определяется по правилу буравчика или правой руки. Легко убедиться в том, что это произведение не ассоциативно, то есть произведение u1 i u2 i ... i un , вообще говоря, не определено, если в нем не расставлены скобки. Действительно, например, 0 = i i (j i j) (i i j) i j = – i. Для того чтобы выражение u1 i u2 i ... i un имело определенный смысл, в нем нужно правильным образом расставить n – 2 пары скобок. Такое выражение со скобками называется ассоциацией.

Теорема 3. Для каждой пары ассоциаций, связанных с выражением u1 i u2 i ... i un , существует такая подстановка {u1 , u2 , ..., un} {i, j, k} (то есть {i, j, k} подставляются каким-то образом вместо всех uk), что значения ассоциаций будут равны между собой и отличны от нуля.

Утверждение касается только векторов в трехмерном пространстве и кажется простым, но доказать его так же трудно, как и теорему о 4-раскраске. Доказательство эквивалентности последней теоремы проблеме четырех красок дано Л. Кауфманом и объяснено на пальцах. Здесь мы только коротко объясним связь этих задач.

Во-первых, причем здесь графы? Графически ассоциацию можно представлять себе в виде дерева, то есть графа специального вида, устроенного следующим образом. Произведению всякой пары u i u соответствует пара ребер (веток), имеющих общую вершину, при этом концы ветвей соответствуют сомножителям, а общее начало – произведению. Шаг за шагом выполняя действия, предписанные в ассоциации скобками, приходим к корню этого дерева, соответствующему результату выполнения всех перемножений в заданной ассоциации. В верхней части рис. 2 представлены деревья, определяемые ассоциациями (u1 i u2) i (u3 i u4) (внизу, ветвями вверх) и (((u1 i u2) i u3) i u4) (вверху, ветвями вниз).

Склеим теперь оба эти дерева по концам веток (концы соответствуют всем элементам ассоциации u1 , u2 , u3 , u4) и затем соединим корни обоих деревьев дополнительным ребром. Получится плоский граф, изображенный в нижней части рис. 6. Отметим два специфических свойства такого графа: в любой его вершине сходится ровно три ребра (это свойство называется кубичностью графа). Удаление любого ребра не приводит к разделению графа на две не связанные между собой компоненты (это свойство назовем отсутствием разделяющего ребра).

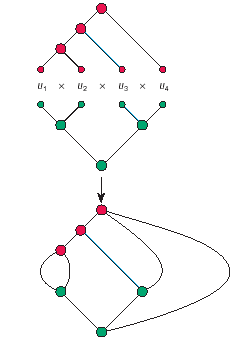


Рис. 6. Плоский граф

Эта конструкция обобщается для любой пары ассоциаций с одинаковым количеством сомножителей.

Во-вторых, причем здесь раскраски? Будем считать векторы i, j и k тремя красками, приписанными всем элементам ассоциации или, что то же, концевым веткам деревьев. Остальные ветки вплоть до корня окрасятся по правилам вычисления векторного произведения. Если мы хотим после вычислений получить ненулевой результат, то, как легко проверить, три ребра, сходящиеся в любой вершине, должны быть раскрашены по-разному.

Тем самым кубический плоский граф, полученный склеиванием двух деревьев различных ассоциаций, получит такую раскраску ребер, что в любой вершине сходятся три по-разному окрашенных ребра. Это так называемая правильная 3-раскраска ребер.

В-третьих, причем здесь проблема четырех красок? Дело в том, что проблемы правильной 4-раскраски вершин и правильной 3-раcкраски ребер эквивалентны. Точнее, справедлива

Теорема 4. Всякий кубический граф без разделяющих ребер допускает правильную 3-раскраску ребер.

Эта теорема эквивалентна теореме 1 о правильной 4-раскраске карт. Доказательство эквивалентности не очень сложно, и его можно найти в большинстве учебников по теории графов.

Объясним лишь, как 4-раскраска областей кубического графа порождает 3-раскраску его ребер. Пусть области кубического графа окрашены четырьмя красками. Вместо того чтобы нумеровать краски числами 1, 2, 3 и 4, занумеруем их парами (0, 0), (0, 1), (1, 0) и (1, 1). При отсутствии разделяющих ребер к каждому ребру примыкают две разные области. Припишем ребру, разделяющему области, окрашенные с помощью (i, j) и (k, l ), цвет по формуле (i + k, j + l )(mod 2). Здесь n(mod 2) означает остаток от деления n на 2. Различающиеся пары цветов областей порождают только три пары (0, 1), (1, 0) и (1, 1) для раскраски ребер.

Доказательство Аппеля и Хакена, в целом хотя и принятое математическим сообществом, вызывает до сих пор определенный скептицизм. Для читателя, поверхностно знакомого с математикой, предыдущая фраза должна вызвать изумление: а как же обязательный для математики принцип исключенного третьего, в соответствии с которым утверждение либо справедливо, либо нет? Дело обстоит не так просто. Вот что пишут сами авторы доказательства: "Читатель должен разобраться в 50 страницах текста и диаграмм, 85 страницах с почти 2500 дополнительными диаграммами, 400 страницами микрофишей, содержащими еще диаграммы, а также тысячи отдельных проверок утверждений, сделанных в 24 леммах основного текста. Вдобавок читатель узнает, что проверка некоторых фактов потребовала 1200 часов компьютерного времени, а при проверке вручную потребовалось бы гораздо больше. Статьи устрашающи по стилю и длине, и немногие математики прочли их сколько-нибудь подробно".

Говоря прямо, компьютерную часть доказательства невозможно проверить вручную, а традиционная часть доказательства длинна и сложна настолько, что ее никто целиком и не проверял. Между тем доказательство, не поддающееся проверке, есть нонсенс. Согласиться с подобным доказательством означает примерно то же, что просто поверить авторам. Но и здесь все сложнее.

Вернемся сначала к доказательствам формулы Эйлера и теоремы о 5-раскраске. Ее-то вроде бы нетрудно проверить, взяв лист бумаги и карандаш. Но рассуждения в ней основаны на очевидных соображениях типа: "Плоский граф разрезает плоскость на совокупность D(G) неперекрывающихся многоугольных областей". Между тем это утверждение не принадлежит к числу аксиом или школьных теорем плоской геометрии, и его нужно доказывать. Соответствующая теорема, сформулированная К. Жорданом, доказывается очень непросто, однако основные трудности связаны с тем, как следует понимать слова типа "разрезает", интуитивно вполне ясные, но с трудом поддающимся формализации. В свете этого замечания становится уже не совсем понятным, доказана ли теорема о пяти красках или мы поверили правдоподобным рассуждениям, основанным на интуитивных представлениях о свойствах геометрических фигур.

Долгое время идеалом математической строгости были формулировки и доказательства Евклида, в которых осуществлялась программа вывода теорем из аксиом по определенным правилам (метод дедукции, позволяющий получать неочевидные утверждения из очевидных посредством ряда явно предъявляемых элементарных, очевидно законных, умозаключений). Этот образец строгости и в наше время недостижим в курсе школьной математики, но для современной чистой математики стандарты Евклида недостаточны. Евклид, по-видимому, не задумывался над тем, почему прямая делит плоскость на две части (и что это значит), он не определял понятия "между", считая это понятие очевидным и т.д. Большая часть соответствующих утверждений доказана или включена в аксиоматику геометрии только в последнюю сотню лет. Формальные выводы из новой системы аксиом стали гораздо длиннее, чем в античные времена.

Трудно даже вообразить длину полного вывода теоремы о пяти красках в соответствии с современными стандартами математической логики и системы аксиом геометрии. Но совершенно точно, что такое рассуждение никто никогда не проделывал из-за бесполезности этого занятия: формальные выводы практически не поддаются проверке в силу свойств человеческой психики: помимо их гораздо большей (по сравнению с привычными рассуждениями) длины их сознательное усвоение идет гораздо медленнее. Поэтому обычно удовлетворяются уверенностью в том, что формальный вывод возможен в принципе.

В формуле Эйлера, например, математики не сомневаются. Вообще принятие доказательства есть некий социальный акт. Выдающийся алгебраист Ю.И. Манин в своей книге "Доказуемое и недоказуемое"[[5]](#footnote-5) пишет по этому поводу: "...отсутствие ошибок в математической работе (если они не обнаружены), как и в других естественных науках, часто устанавливается по косвенным данным: имеют значение соответствие с общими ожиданиями, использование аналогичных аргументов в других работах, разглядывание "под микроскопом" отдельных участков доказательства, даже репутация автора, словом, воспроизводимость в широком смысле. "Непонятные" доказательства могут сыграть очень полезную роль, стимулируя поиски более доступных рассуждений."

Именно такая история происходит и с доказательством теоремы о четырех красках. Не так давно появилось новое доказательство, причем та часть, которая выполнена не на компьютере, уже поддается проверке. Однако компьютерная часть все еще остается скорее предметом веры. Ведь даже проверка распечаток всех программ и всех входных данных не может гарантировать от случайных сбоев или даже от скрытых пороков электроники (вспомним, что ошибки при выполнении деления у первой версии процессора Pentium были случайно обнаружены спустя полгода после начала его коммерческих продаж – кстати, математиком, специалистом в теории чисел). По-видимому, единственный способ проверки компьютерных результатов – написать другую программу и для другого типа компьютера. Это, конечно, совсем непохоже на стандартный идеал дедуктивных рассуждений, но именно так осуществляется проверка утверждений во всех экспериментальных науках[[6]](#footnote-6).

Из которых математика, стало быть, исключена напрасно.

# 2 Математическая логика и теория типов

**Математическая логика** — раздел математики, изучающий доказательства. Согласно определению П. С. Порецкого, «математическая логика есть логика по предмету, математика по методу»[[7]](#footnote-7).

Применение в логике математических методов становится возможным тогда, когда суждения формулируются на некотором точном языке. Такие точные языки имеют две стороны: синтаксис и семантику. Синтаксисом называется совокупность правил построения объектов языка (обычно называемых формулами). Семантикой называется совокупность соглашений, описывающих наше понимание формул (или некоторых из них) и позволяющих считать одни формулы верными, а другие — нет.

Важную роль в математической логике играет понятие исчисления. Исчислением называется совокупность правил вывода, позволяющих считать некоторые формулы выводимыми. Правила вывода подразделяются на два класса. Одни из них непосредственно квалифицируют некоторые формулы как выводимые. Такие правила вывода принято называть аксиомами. Другие же позволяют считать выводимыми формулы *A*, синтаксически связанные некоторым заранее определённым способом с конечными наборами выводимых формул. Широко применяемым правилом второго типа является правило modus ponens: если выводимы формулы *A* и , то выводима и формула *B*.



Отношение исчислений к семантике выражается понятиями семантической пригодности и семантической полноты исчисления. Исчисление И называется семантически пригодным для языка Я, если любая выводимая в И формула языка Я является верной. Аналогично, исчисление И называется семантически полным в языке Я, если любая верная формула языка Я выводима в И.

Многие из рассматриваемых в математической логике языков обладают семантически полными и семантически пригодными исчислениями. В частности, известен результат К. Гёделя о том, что так называемое классическое исчисление предикатов является семантически полным и семантически пригодным для языка классической логики предикатов первого порядка. С другой стороны, имеется немало языков, для которых построение семантически полного и семантически пригодного исчисления невозможно. В этой области классическим результатом является теорема Гёделя о неполноте, утверждающая невозможность семантически полного и семантически пригодного исчисления для языка формальной арифметики.

**Теория типов** — математически формализованная база для проектирования, анализа и изучения систем типов данных в теории языков программирования (раздел информатики). Многие программисты используют это понятие для обозначения любого аналитического труда, изучающего системы типов в языках программирования. В научных кругах под **теорией типов** чаще всего понимают более узкий раздел дискретной математики, в частности λ-исчисление.

Современная теория типов была частично разработана в процессе разрешения парадокса Рассела и во многом базируется на работе Бертрана Рассела и Альфреда Уайтхэда «Principia Mathematica» (этот фундаметальный трёхтомник математической логики до сих пор не издан на русском языке)[[8]](#footnote-8).

# Заключение

Прародителем информатики является кибернетика, основанная американским математиком Норбертом Винером, опубликовавшим в 1948 году одноименную книгу. Основоположником советской школы кибернетики и информатики признан профессор МГУ Алексей Андреевич Ляпунов.

Слово «информатика» для обозначения комплекса компьютерных наук было введено в словарь русского языка в 1976 году академиком Андреем Петровичем Ершовым.

Несмотря на широкую распространенность термина «информатика», у специалистов до сих пор нет единого мнения о его толковании. Существуют три подхода:

• сверхширокий, включающий в информатику все, что связано с любыми процессами получения, преобразовании и передачи информации;

• широкий, включающий в информатику все, что связано с компьютерами, в том числе вопросы конструирования вычислительной техники;

• узкий, определяющий информатику только как науку о применении компьютеров, то есть как науку о компьютерных технологиях.

Таким образом, к настоящему времени имеются три толкования термина «информатика».

Первое – сверхширокое, при котором в сферу ее ведения попадает весь комплекс наук, так или иначе связанных с получением и обработкой информации, независимо от использования компьютеров. В этом значении термин часто используется в изданиях философской и методологической направленности, а также в непрофессиональной среде (журналистами, политиками).

Второе – информатика как полный набор компьютерных наук, точный эквивалент computer science. В данном значении термин объединяет самые разные стороны программирования и использования компьютеров, методов их конструирования и разработки программного обеспечения. Такое толкование чаще всего используется в обычном профессиональном языке и при обратном переводе на английский. Например, «факультет информатики» правильнее всего перевести как «computer science faculty» или «computer science department» в зависимости от того, на какую аудиторию рассчитан перевод (в британском английском более распространено слово «faculty», а в американском – «department»).

Третье – информатика в узком смысле, когда за рамки computer science выносятся детальные вопросы технического устройства компьютеров (hardware), а в составе науки остаются проблемы их применения. В таком значении термин обычно используется в узкопрофессиональной среде программистов, а также в учебных программах. Именно так его следует понимать в общепринятом в образовательной среде словосочетании «информатика и вычислительная техника», иначе получается логическая некорректность.

Как известно, всякая классификация условна и имеет некоторую цель. В свете всего изложенного мы, имея в виду подготовку специалистов в области компьютерных наук, будем пользоваться последним, узким толкованием и определим информатику как научную дисциплину, предметом которой являются компьютерные технологии. Вместо термина «компьютерные» часто используются аналогичные по смыслу определения «новые информационные» или просто «информационные», поэтому в специальной литературе можно встретить термины «ИТ-служба», «ИТ-специалист», «факультет ИТ» и т.п.

Для иллюстрации границ раздела между кибернетикой, вычислительной техникой и информатикой можно воспользоваться таким образным сравнением. Если уподобить кибернетика, разрабатывающего алгоритмы, композитору, сочиняющего музыку, а конструктора ЭВМ – скрипичному мастеру, то специалиста по информатике можно будет сравнить со скрипачом, реализующим замысел композитора и обогащающим его своим мастерством и талантом. Поэтому информатика – не просто отрасль знаний, а неделимый сплав ремесла, науки и искусства.

Для профессионального употребления рекомендуется руководствоваться узким подходом, выделяя в самостоятельные науки кибернетику, вычислительную технику и информатику[[9]](#footnote-9).

Возникновение информатики во второй половине XX столетия не является случайностью. Компьютер и электросвязь – два закономерных продукта и инструмента информационной революции, знаменующей переход от индустриальной к постиндустриальной (информационной) эпохе в истории человечества.

# Список использованной литературы

1. Апокин И. А., Майстров Л. Е. История вычислительной техники: От простейших счетных приспособлений до сложных релейных систем. М.: Наука, 2000.
2. Гладких Б. А. От абака до компьютера. Томск: Изд-во НТЛ, 2005.
3. Гутер Р. С., Полунов Ю. Л. От абака до компьютера. М.: Знание, 2001. .
4. Кук Д., Бейз Г. Компьютерная математика. М., Наука, 2000.
5. Марков А.А. Элементы математической логики. М.: Изд-во МГУ, 2004.
6. Пойа Д. Математическое открытие. М.: Наука, 2000.
7. Прилуцкий М.Х. Математические основы информатики. Нижний Новгород: Нижег.гос.ун-т, 2000.
8. Симонович С., Евсеев Г., Алексеев А. Общая информатика. М.: Дело, 1999.
9. Турецкий В.Я. Математика и информатика. Екатеринбург: Пропаганда, 2002.
10. Фор Р., Кофман А., Дени-Папен М. Современная математика. М.: Мир, 2006.
11. Частиков А. Архитекторы компьютерного мира. Спб: БХВ-Петербург, 2002.
12. Шенфилд Дж. Математическая логика. М.: Наука, 2005.

1. Прилуцкий М.Х. Математические основы информатики. Нижний Новгород: Нижег.гос.ун-т, 2000. С. 21. [↑](#footnote-ref-1)
2. В теории алгоритмов NP-полные задачи — это класс задач, лежащих в классе NP (то есть для которых пока не найдено быстрых алгоритмов решения, но проверка того, является ли данное решение правильным, проходит быстро), к которым сводятся все задачи класса NP. Это означает, что если найдут быстрый алгоритм для решения любой из NP-полных задач, любая задача из класса NP сможет быть решена быстро. [↑](#footnote-ref-2)
3. Если не прибегнуть к хитрости спрыгнуть в воду и доплыть до противоположного берега, что позволит решить задачу с мостами, но, к сожалению, изменит условие задачи с графами... [↑](#footnote-ref-3)
4. Однако из доказательства Хивуд понял, что пяти красок действительно достаточно. Тем не менее для любой конкретной карты хватало четырех красок! [↑](#footnote-ref-4)
5. Марков А.А. Элементы математической логики. М.: Изд-во МГУ, 2004. С. 211. [↑](#footnote-ref-5)
6. Кук Д., Бейз Г. Компьютерная математика. М., Наука, 2000. С. 156. [↑](#footnote-ref-6)
7. Турецкий В.Я. Математика и информатика. Екатеринбург: Пропаганда, 2002. С. 109. [↑](#footnote-ref-7)
8. Симонович С., Евсеев Г., Алексеев А. Общая информатика. М.: Дело, 1999. С. 231. [↑](#footnote-ref-8)
9. Гладких Б. А. От абака до компьютера. Томск: Изд-во НТЛ, 2005. С. 87. [↑](#footnote-ref-9)