

Министерство образования и науки Российской Федерации
Федеральное государственное бюджетное образовательное учреждение
высшего образования
«Владимирский государственный университет
имени Александра Григорьевича и Николая Григорьевича Столетовых»
(ВлГУ)

ИНСТИТУТ ИНФОРМАЦИОННЫХ ТЕХНОЛОГИЙ И РАДИОЭЛЕКТРОНИКИ

КАФЕДРА ИНФОРМАТИКИ И ЗАЩИТЫ ИНФОРМАЦИИ

Спирина Татьяна Венедиктовна

Дисциплина «Информатика»

1 семестр

конспект лекций по дисциплине «Информатика»
для бакалавров ВлГУ обучающихся по направлению 13.03.03 Энергетическое
машиностроение

(шифр направления, название)

Владимир – 20____г.

Целями освоения дисциплины «Информатика» являются обеспечение подготовки специалистов в соответствии с требованиями ФГОС ВО и учебного плана по направлению 13.03.03; ознакомление студентов с основными концептуальными идеями такой важной области человеческого знания как информатика, определяющей развитие общества на основе формирования информационной культуры человека; формирование у студентов обобщенного представления о возможности заимствования методов информатики для познания окружающего мира на основе математического моделирования, методов математической статистики и технологий автоматизированной обработки данных; развитие у студентов способности создания личностной интеллектуальной технологии как средства эффективного овладения знаниями и умениями в сфере профессиональной деятельности с помощью методов информатики.

Задачами дисциплины «Информатика» является:

- изучение основных понятий информационных технологий в объеме, необходимом для использования и анализа информационных и социокультурных процессов, общие принципы работы компьютеров, основы информационной безопасности;
- формирование умений использовать стандартные средства Windows, пакет программ MS Office, программные средства архивации, резервного копирования и защиты данных компьютера, автоматизировать решение практических задач, ставить и решать типовые задачи в области технологий профессиональной деятельности, подбирать и использовать адекватные методы и средства использования и обработки информации, оценивать эффективность используемых методов и компьютерных средств обработки информации;
- овладение навыками применения современных информационных технологий и средств к текущим реальным ситуациям.

Данная дисциплина относится к базовой части блока 1 ОПОП. В учебном плане предусмотрены виды учебной деятельности, обеспечивающие синтез теоретических лекций и лабораторных работ. Дисциплина тесно взаимосвязана с другими дисциплинами базовой части, призвана стимулировать творческое использование информационных технологий применительно к таким смежным дисциплинам как высшая математика, инженерная графика.

Дисциплина изучается на первом курсе, в связи с чем, требования к «входным» знаниям, умениям и готовностям обучающегося определяются требованиями к уровню подготовки выпускника в соответствии с программой общеобразовательной школы по предмету «Информатика и ИКТ».

В результате освоения дисциплины студент должен обладать:

ОПК-1 - способностью осуществлять поиск, хранение, обработку и анализ информации из различных источников и баз данных, представлять ее в требуемом формате с использованием информационных, компьютерных и сетевых технологий.

Знать: основные понятия информационных технологий в объеме, необходимом для использования и анализа информационных и социокультурных процессов, общие принципы работы компьютеров, основы информационной безопасности (ОПК-1).

Уметь: - использовать стандартные средства Windows, пакет программ MS Office, программные средства архивации, резервного копирования и защиты данных компьютера, автоматизировать решение практических задач, ставить и решать типовые задачи в области технологий профессиональной деятельности, подбирать и использовать адекватные методы и средства использования и обработки информации, оценивать эффективность используемых методов и компьютерных средств обработки информации (ОПК-1).

Владеть: навыками применения современных информационных технологий и средств к текущим реальным ситуациям (ОПК-1).

СОДЕРЖАНИЕ:

ГЛАВА 1. Основы теории информации.

Информация. Информационные объекты, данные, знания. Виды и типы информации. Информационные процессы. Количество информации. Аддитивная мера Хартли. Мера Шеннона.

ГЛАВА 2. Системы счисления.

Позиционные и непозиционные системы счисления. Алфавит, базис, основание системы счисления, порядок. Правила перевода чисел из десятичной системы счисления в q-ичную. Правила перевода чисел из q-ичной системы счисления в десятичную. Правила перевода чисел из двоичной системы счисления в восьмеричную и обратно. Правила перевода чисел из двоичной системы счисления в шестнадцатеричную и обратно. Арифметические операции с числами, записанными в различных системах счисления.

ГЛАВА 3. Представление информации в компьютере.

Представление целых чисел в компьютере. Представление целых чисел без знака. Представление целых чисел со знаком. Прямой, обратный и дополнительный коды числа. Арифметические действия с целыми числами. Переполнение разрядной сетки формата числа. Кодирование текстовой информации. Кодирование графической информации. Кодирование звуковой информации.

ГЛАВА 4. Алгебра логики.

Основные понятия алгебры логики. Высказывание. Логическая переменная. Логическая функция. Логические операции: инверсия, конъюнкция, дизъюнкция, импликация, эквиваленция, строгая дизъюнкция. Логические выражения. Тожественные преобразования логических выражений. Законы алгебры логики. Совершенные нормальные формы. Совершенная конъюнктивная нормальная форма (СКНФ). Совершенная дизъюнктивная нормальная форма (СДНФ) средствами алгебры логики. Табличный способ решения логических задач. Основные способы решения логических задач. Решение логических задач средствами алгебры логики. Табличный способ решения логических задач.

ГЛАВА 5. Элементы теории алгоритмов.

Алгоритм. Свойства алгоритма. Блок-схема. Код. Псевдокод. Язык программирования. Линейный алгоритм. Ветвление. Полное и неполное ветвление. Циклический алгоритм. Цикл с предусловием. Цикл с постусловием. Цикл с параметром.

ГЛАВА 6. Устройство компьютера.

Назначение компьютера. Функциональная организация работы компьютера. Системный блок. Устройства хранения информации. Устройства ввода информации. Устройства вывода информации.

ГЛАВА 7. Программное обеспечение компьютера.

Системное программное обеспечение: базовое программное обеспечение, сервисное программное обеспечение. Пакеты прикладных программ. Операционная система Windows. Пакет офисных программ MS Office.

ГЛАВА 8. Компьютерные сети. Интернет.

Локальная компьютерная сеть. Глобальная сеть. Internet.

ГЛАВА 9. Информационная безопасность.

Понятие и сущность информационной безопасности. Анализ угроз информационной безопасности. Обеспечение информационной безопасности.

ГЛАВА 1. Основы теории информации.

Термин «**информация**» происходит от латинского слова *information*, означающего сведения, разъяснения, изложение. Смысловое содержание термина «информация» разъясняется в зависимости от сферы, в которой он употребляется. В обиходе информацией называют любые данные или сведения, которые кого-либо интересуют. В «Энциклопедии кибернетики» информация определяется как «одно из наиболее общих понятий науки, обозначающее некоторые сведения, совокупность каких-либо данных, знаний и т.п.». В широком смысле информация – это отражение реального мира, в узком смысле – это любые сведения, являющиеся объектом хранения, передачи и преобразования. Понятие «информация» вместе с понятиями «вещество» и «энергия» относятся к основным понятиям науки, поэтому строгого и точного научного определения этому понятию дать невозможно. В информатике нас, прежде всего, интересует определение, сформировавшееся в математической теории информации. Согласно американскому ученому и инженеру Клоду Шеннону, информация – это **снятая неопределенность**. Согласно Шеннону, информативность сообщения характеризуется содержащейся в нем полезной информацией, т.е. той частью сообщения, которая снимает полностью или уменьшает существовавшую до ее получения неопределенность какой-либо ситуации. Величина неопределенности некоторого события – это количество возможных результатов (исходов) данного события. Такой подход к определению информации называют содержательным. Так, например, неопределенность погоды на завтра обычно заключается в диапазоне температуры воздуха и возможности выпадения осадков. Причем в разное время года диапазон возможных исходов данного события различен и колеблется в районе 16 ожидаемых результатов для значения дневной температуры и трех-четырех – для осадков (без осадков, кратковременные дожди, ливневые дожди с градом и т.п.).

У разных людей неопределенность знаний о каком-либо предмете может быть различной. Так, неопределенность знаний о погоде в Москве у рядового американца и российского синоптика может сильно различаться. Поэтому содержательный подход к трактовке понятия информации часто является субъективным. Если же число возможных исходов события не зависит от суждений различных людей, то получаемая информация о наступлении одного из возможных исходов является объективной. Примером такой информации является сообщение о результате падения подброшенной монеты или игрального кубика. Всякое сообщение можно закодировать с помощью конечной последовательности символов некоторого алфавита. Согласно Колмогорову, количество информации, содержащейся в последовательности символов, определяется минимально возможным количеством двоичных знаков, необходимых для кодирования этой последовательности безотносительно к содержанию представленного ею сообщения. Данный подход к определению информации называют алфавитным. Смысл сообщения при этом может быть учтен лишь на этапе выбора исходного алфавита для его записи или не учтен вообще. В частности, сообщения, записанные на естественном языке с помощью соответствующего алфавита (например, сейчас вы читаете сообщение на русском языке, записанное с помощью русского же алфавита), кодируется без учета их смыслового содержания. Такой подход является объективным.

На первый взгляд эти два подхода кажутся существенно различными, тем не менее они хорошо согласуются друг с другом.

Хотя информацию нельзя строго определить, ее можно измерить. Чтобы стандартизировать измерение количества информации, договорились за единицу измерения брать бит (от английского *binary digit*). При алфавитном подходе один бит – это количество информации, которое можно передать в сообщении, состоящем из одного

двоичного знака (0 или 1). С точки зрения содержательного подхода один бит – это количество информации, уменьшающее неопределенность знания о предмете в два раза. Содержательный подход отвечает на вопрос, какое количество «новой» информации мы получаем из сообщения. Например, сообщение о том, что подброшенная монета упала «решкой» вверх, содержит один бит информации. В самом деле, неопределенность знаний о результате падения монеты заключалась в двух возможных исходах. После того, как конкретный исход стал известен, неопределенность уменьшилась в два раза, что и соответствует одному биту информации. Но ведь результат падения монеты как раз можно закодировать одним из двух символов (например, используя все те же 0 и 1. Таким образом, несмотря на различие в трактовках, понятие бита является согласованным для различных подходов. На практике чаще используется более крупная единица – байт, равная 8 битам. Используются также следующие производные единицы измерения информации:

1 килобайт (1 Кб) = 2^{10} байт = 1024 байт;

1 мегабайт (1 Мб) = 2^{10} Кб = 2^{20} байт;

1 гигабайт (1 Гб) = 2^{10} Мб = 2^{30} байт;

1 терабайт (1 Тб) = 2^{10} Гб = 2^{40} байт.

Информация может существовать в самых разнообразных **формах**:

- в виде текстов, рисунков, чертежей, фотографий;
- в виде световых или звуковых сигналов;
- в виде радиоволн;
- в виде электрических и нервных импульсов;
- в виде магнитных записей;
- в виде жестов и мимики;
- в виде запахов и вкусовых ощущений;
- в виде хромосом, посредством которых передаются по наследству признаки и свойства организмов и т.д.

По способам представления рассматривают числовую, текстовую, графическую, звуковую и видеоинформацию.

Предметы, процессы, явления материального или нематериального свойства, рассматриваемые с точки зрения их информационных свойств, называются **информационными объектами**.

Информация обладает следующими **свойствами**:

1. Объективность (информация объективна, если она не зависит от чьего-либо мнения, суждения).
2. Достоверность (информация достоверна, если она отражает истинное положение дел).
3. Полнота (информацию можно назвать полной, если ее достаточно для понимания и принятия решения).
4. Актуальность (своевременность, важность, существенность для настоящего времени; только вовремя полученная информация может принести необходимую пользу).
5. Адекватность (определенный уровень соответствия создаваемого с помощью полученной информации образа реальному объекту, процессу, явлению).

Например, телефонный справочник обычно хранит информацию об абонентах городской телефонной сети. Однако, если мы возьмем телефонный справочник г.Владимира за 1975 год, то сведения в нем содержащиеся, будут по крайней мере не достоверны, следовательно, для абонента Владимирской городской телефонной сети 21 века информации такой справочник почти не содержит.

Процессы, связанные с определенными операциями над информацией, называются **информационными процессами**. Можно выделить три основных типа информационных процессов: хранение, передача и обработка информации.

Хранение информации – это распространение ее во времени. Человек хранит информацию либо в собственной памяти, либо на каких-либо внешних носителях.

Обработка информации составляет основу процесса преобразования информации (принятие решения).

Информация может быть передана для ее последующего использования, обработки или хранения.

С практической точки зрения информация всегда представляется в виде сообщения. Информационное сообщение связано с источником сообщения, получателем сообщения и каналом связи. Информация передаётся в виде сообщений от некоторого источника информации к её приёмнику посредством канала связи между ними. Источник посылает передаваемое сообщение, которое кодируется в передаваемый сигнал. Этот сигнал посылается по каналу связи. В результате в приёмнике появляется принимаемый сигнал, который декодируется и становится принимаемым сообщением.

Передача информации по каналам связи часто сопровождается воздействием помех, вызывающих искажение и потерю информации.

Основные черты информации заключаются в том, что она:

- не материальна, но для своего использования требует материального носителя;
- существует лишь для тех пользователей (приемников), которые способны ее воспринять;
- может быть представлена с помощью любых сигналов и знаков, которые могут содержать информацию сами по себе или в своем взаимном расположении.

В современном мире информация, как правило, обрабатывается на вычислительных машинах. Синоним словосочетания «вычислительная машина» (чаще электронная вычислительная машина (ЭВМ)) – компьютер. Компьютер – устройство преобразования информации посредством выполнения программой последовательности операций.

В зависимости от вида перерабатываемой информации вычислительные машины (ВМ) подразделяют на два основных класса: аналоговые и цифровые. Аналоговая вычислительная машина (АВМ) – это машина, оперирующая информацией, представленной в виде непрерывных изменений некоторых физических величин. Цифровая вычислительная машина (ЦВМ) – это машина, оперирующая информацией, представленной в дискретном виде. АВМ обычно предназначены для решения определенного класса задач, ЦВМ – универсальное вычислительное средство. Наибольшее распространение получили электронные вычислительные машины, выполненные с использованием новейших достижений электроники.

Очень широко употребляется еще один термин – «**данные**». Термин данные происходит от слова data - факт, в то время как информация (informatio) означает разъяснение, изложение, т.е. сведения или сообщение. Данные – это совокупность сведений, зафиксированных на определенном носителе в форме, пригодной для постоянного хранения, передачи и обработки. Преобразование и обработка данных позволяет получить информацию. Процесс такого преобразования называется кодированием. При этом термин «код» служит для обозначения системы символов, применяемой для хранения, передачи и обработки информации.

Информация – это результат преобразования и анализа данных. Отличие информации от данных состоит в том, что данные - это фиксированные сведения о событиях и явлениях, которые хранятся на определенных носителях, а информация появляется в результате обработки данных при решении конкретных задач. Например, в базах данных хранятся различные данные, а по определенному запросу система управления базой данных выдает требуемую информацию.

Информация существует не только в виде данных, но и в виде **знаний**. Здесь знания – это совокупность объективных фактов, способов и технологий, систематизированных и дающих реальное представление о предметах, процессах и явлениях, т.е. специальным образом структурированная информация. Знания – это зафиксированная и проверенная практикой обработанная информация, которая использовалась и может многократно использоваться для принятия решений. Знания – это вид информации, которая хранится в базе знаний и отображает знания специалиста в конкретной предметной области. Знания – это интеллектуальный капитал.

Принятие решений – это выбор наилучшего в некотором смысле варианта решения из множества допустимых на основании имеющейся информации. Для решения поставленной задачи фиксированные данные обрабатываются на основании имеющихся знаний, далее полученная информация анализируется с помощью имеющихся знаний. На основании анализа, предлагаются все допустимые решения, а в результате выбора принимается одно наилучшее в некотором смысле решение. Результаты решения пополняют знания

Данные и знания в виде информационных потоков циркулируют в **информационных системах** (ИС). Сбор, накопление, обработка, хранение и использование информации в ИС осуществляется с помощью **информационных технологий**. Информационные технологии (ИТ) – это машинизированные (инженерные) способы обработки информации, которые реализуются посредством автоматизированных (человеко-машинных, т.е. управляемых человеком) информационных систем (АИС). АИС основаны на ЭВМ и телекоммуникационной технике.

Чтобы измерить количество информации, содержащееся в сообщении или сигнале, можно воспользоваться так называемой *аддитивной мерой Хартли*. Введем понятия глубины сообщения и длины сообщения.

Глубина сообщения q – количество различных элементов (символов, знаков), принятых для представления сообщений. В каждый момент времени реализуется только один какой-либо элемент.

Длина сообщения n – количество позиций, необходимых и достаточных для представления сообщений заданной величины.

При заданных глубине и длине сообщения количество всех возможных сообщений (N), которое можно представить:

$$N = q^n.$$

Задача 1. Подсчитать количество всех трехбуквенных русских слов.

Решение.

Всего трехбуквенных русских слов $N = 32^3 = 32768$. Заметим, что в это количество включены и «совершенно невозможные» слова, например «aaa» или «ъъъ», но, подчеркнем, что при описываемом подходе смысл (семантика) сообщения не рассматривается.

Ответ: 32768 слов.

Определим логарифмическую меру, позволяющую вычислять количество информации в битах. Такую информационную емкость (I) называют *мерой Хартли*:

$$I = \log_2 N = n \log_2 q.$$

Другими словами количество информации, полученное при выборе одного предмета из N равнозначных предметов, равно $\log_2 N$. То есть именно такое количество информации необходимо для устранения неопределенности из N равновероятных вариантов.

Задача 2. Чему равно количество информации для трехбуквенных русских слов?

Решение

Для трехбуквенных русских слов $I = 3 \log_2 32 = 15$ (бит).

Ответ: 15 бит.

Значение I – количество двоичных разрядов. Оно не может быть дробным, поэтому в случае получения дробного значения, необходимо округлить результат.

Итак, 1 бит информации соответствует одному элементарному событию, которое может произойти или не произойти. Такая мера количества информации удобна тем, что она позволяет оперировать мерой как числом. Количество информации при этом эквивалентно количеству двоичных символов 0 или 1.

Закон аддитивности информации: при наличии нескольких источников информации общее количество информации равно

$$I = I_1 + I_2 + \dots + I_k = \sum_{i=1}^k I_i$$

где I_i – количество информации от источника i . Всего источников – k .

Задача 3. Используя закон аддитивности и формулу Хартли, подсчитайте, какое количество информации несет достоверный прогноз погоды.

Решение.

Предположим, что прогноз погоды на следующий день заключается в предсказании дневной температуры (обычно выбор делается из 16 возможных для данного сезона значений) и одного из четырех значений облачности (солнечно, переменная облачность, пасмурно, дождь). Получаемое при этом количество информации равно

$$I = \log_2 16 + \log_2 4 = 4 + 2 = 6 \text{ (бит)}$$

Ответ: 6 бит.

Логарифмическая мера информации позволяет измерить количество информации и используется на практике.

Задача 4. *Определить, какое количество информации в битах содержит книга, написанная на русском языке, содержащая 200 страниц (каждая страница содержит приблизительно 2000 символов).*

Решение

Для решения воспользуемся мерой Хартли:

$I = n \log_2 q = 200 \cdot 2000 \cdot \log_2 32 = 2000000$ бит. Здесь n – длина сообщения в знаках: $n = 200 \text{ страниц} \cdot 2000 \text{ знаков/страница} = 400000$ знаков.

Ответ: 2000000 бит.

В **задачах 1 и 2** речь шла о равновероятных событиях. Но в реальности очень часто приходится иметь дело с неравновероятными событиями (задача 3). Интуитивно понятно, например, что для студента–отличника получение оценки «отлично» и получение оценки «неудовлетворительно» – события не равновероятные. Для такого студента получить «отлично» – вполне вероятное событие, а получение «неуда» – маловероятное. Для «двоечника» – все наоборот.

Что же такое вероятность? Для примера возьмем получение оценок. Чтобы определить, какова вероятность получения каждой оценки, нужно подсчитать общее количество разных оценок, полученных студентом за достаточно длительный период времени, и определить, сколько из них «двоек», «троек», «четверок» и «пятёрок». Если допустить, что такое же распределение сохранится и в будущем, то можно рассчитать вероятности получения каждой оценки. Определив, какую часть от общего числа оценок составляют «двойки», найдем вероятность получения «двойки». Затем, определив, какую часть от общего количества составляют «тройки», найдем вероятность получения «тройки». Доля «четвёрок» среди всех оценок – это вероятность получения «четверки», а доля «пятёрок» – это вероятность получения «пятёрки».

Предположим, мы посчитали, что за два семестра студент получил 100 оценок. Среди них: 60 «пятёрок», 25 «четвёрок», 10 «троек» и 5 «двоек». Тогда:

- вероятность «пятерки»: $60/100=0,6$;
- вероятность «четверки»: $25/100=0,25$;
- вероятность «тройки»: $10/100=0,1$;

- вероятность «двойки»: $5/100=0,05$.
 Обратите внимание, что *сумма вероятностей возможных событий равна 1*.
 Значение вероятности будем обозначать буквой p . Зная вероятности событий, можно определить количество информации в сообщении о каждом из них.

Согласно теории информации, для этого нужно решить показательное уравнение

$$2^I = \frac{1}{p}, \text{ т.е. } I = \log_2 \frac{1}{p}. \text{ Эта формула получила название формулы Шеннона (меры Шеннона).}$$

Подсчитаем по этой формуле количество информации, содержащейся в сообщении о получении нашим студентом каждой из оценок.

$$I_5 = \log_2 \frac{1}{0,6} = \log_2 \frac{5}{3} = 0,737 \text{ (бит)}$$

$$I_4 = \log_2 \frac{1}{0,25} = \log_2 4 = 2 \text{ (бита)}$$

$$I_3 = \log_2 \frac{1}{0,1} = \log_2 10 = 3,322 \text{ (бит)}$$

$$I_2 = \log_2 \frac{1}{0,05} = \log_2 20 = 4,322 \text{ (бит)}$$

Посмотрите внимательно на результаты, и вы увидите, что чем меньше вероятность события, тем больше информации несёт сообщение о нём.

Вывод: *количество информации в сообщении о некотором событии зависит от вероятности этого события. Чем меньше вероятность, тем больше информации.*

Теперь без труда решим **задачу 3**.

Краткая запись условия	Решение
$N_{\Gamma} = 9$ $N_{\text{я}} = 3$ $I_{\text{я}} = ?$ $I_{\Gamma} = ?$	$p = \frac{N_{\text{частн}}}{N_{\text{об}}}$ $p_{\text{я}} = \frac{3}{12} = \frac{1}{4}, \quad p_{\Gamma} = \frac{9}{12} = \frac{3}{4}.$ $I_{\text{я}} = \log_2 \frac{1}{p_{\text{я}}} = \log_2 4 = 2 \text{ (бита)}$ $I_{\Gamma} = \log_2 \frac{1}{p_{\Gamma}} = \log_2 \frac{4}{3} = \log_2 4 - \log_2 3 = 2 - \log_2 3 \text{ (бит)}$

На первый взгляд, кажется, что мы имеем две совсем разные формулы для вычисления количества информации. Первая через количество событий $I = \log_2 N$, вторая – через вероятность $I = \log_2 \frac{1}{p}$.

На самом деле эти формулы связаны друг с другом, а именно первая формула является частным случаем второй, когда вероятность событий одинаковая. Другими словами формула Хартли является частным случаем формулы Шеннона.

Решим **задачу 2** с использованием формулы Шеннона. Всех фруктов поровну – по 4. Тогда вероятность выбора каждого вида фрукта равна $\frac{4}{8} = \frac{1}{2}$. Значит, и количество

информации будет одинаковым $I_{\text{с}} = I_{\text{я}} = \log_2 \frac{2}{1} = \log_2 2 = 1 \text{ (бит)}$. В задаче 1 мы получили такой же ответ.

Задача 4.

В пруду живут 8000 карасей, 2000 щук и 40 000 пескарей.

- Определите вероятность улова каждого вида рыб;
- Определите количество информации, полученной рыбаком при улове карася;
- Определите количество информации, полученной рыбаком при улове щуки;
- Определите количество информации, полученной рыбаком при улове пескаря.

Решение (б)

Краткая запись условия	Решение
$N_k=8000$ $N_{щ}=2000$ $N_p=40000$	Основные формулы: $I = \log_2 \frac{1}{p}, N_{об} = N_k + N_{щ} + N_p, P = \frac{N_k}{N_{об}}$
$I_k - ?$	$P = \frac{8000}{50000} = \frac{4}{25},$ $I_k = \log_2 \frac{25}{4} = \log_2 25 - \log_2 4 = 4,64 - 2 = 2,64(\text{бит})$

Ответ: количество информации, полученной рыбаком при улове карася равно 2,64 бита.

Задача 5.

В корзине лежат белые и черные шары. Среди них 18 черных шаров. Сообщение о том, что из корзины достали белый шар, несет 2 бита информации. Сколько всего в корзине шаров?

Решение.

Из условия можно увидеть, что количество черных и белых шаров различное, поэтому воспользуемся формулой Хартли для неравновероятных событий. Обозначим $N_{\text{ч}}$, $N_{\text{б}}$ – количество черных и белых шаров соответственно, N – общее количество шаров, $I_{\text{б}}$ – количество информации в сообщении, что из корзины достали белый шар, $p_{\text{б}}$ – вероятность выбора белого шара.

Краткая запись условия	Решение
$N_{\text{ч}}=18$ шт $I_{\text{б}}=2$ бита	Основные формулы: $I = \log_2 \frac{1}{p}, N = N_{\text{ч}} + N_{\text{б}}, 2 = \log_2 2^2 \Rightarrow$
$N - ?$	$\frac{1}{p_{\text{б}}} = 4 \Rightarrow p_{\text{б}} = \frac{1}{4}.$ С другой стороны по формуле $p_{\text{б}} = \frac{N_{\text{б}}}{N_{\text{б}} + 18}.$ Составим и решим уравнение $\frac{1}{4} = \frac{N_{\text{б}}}{N_{\text{б}} + 18} \Rightarrow N_{\text{б}} = 6,$ $N = 6 + 18 = 24.$

Ответ: всего 24 шара.

Задача 6.

Каждый аспирант кафедры "Информационные системы" изучает только один из трех языков: английский, немецкий или французский. Причем 30 аспирантов не изучают английский язык. Информационный объем сообщения "Аспирант Петров изучает английский язык" равен $(1 + \log_2 3)$ бит. Количество информации, содержащееся в сообщении "Аспирант Иванов изучает французский язык", равно двум битам. Иностраный студент, приехавший в университет, знает только немецкий язык. Чему равно количество аспирантов кафедры, с которыми сможет общаться иностранный студент?

Решение.

Из условия видно, что количество аспирантов, изучающих английский, немецкий и французский языки различное и вопрос задачи указывает на конкретное изучение языка, поэтому воспользуемся формулой Хартли для неравновероятных событий.

Обозначим N_n , N_ϕ , N_a – количество абитуриентов, изучающих немецкий, французский и английский языки соответственно, I_a – количество информации в сообщении "Аспирант Петров изучает английский язык", I_ϕ – количество информации в сообщении "Аспирант Иванов изучает французский язык".

Краткая запись условия	Решение
$N_n + N_\phi = 30$ $I_a = (1 + \log_2 3)$ бита $I_\phi = 2$ бита $N_n - ?$	Основные формулы: $I = \log_2 \frac{1}{p} \cdot 1 + \log_2 3 = \log_2 2^{1 + \log_2 3} \Rightarrow$ $\frac{1}{p_a} = 2^{1 + \log_2 3} = 2^1 \cdot 2^{\log_2 3} = 2 \cdot 3 = 6 \Rightarrow p_a = \frac{1}{6}.$ С другой стороны $p_a = \frac{N_a}{N_a + 30}.$ Составим и решим уравнение: $\frac{1}{6} = \frac{N_a}{N_a + 30} \Rightarrow N_a = 6.$ Аналогично $p_\phi = \frac{1}{4}$ и $p_\phi = \frac{N_\phi}{N_a + 30} \Rightarrow N_\phi = 9,$ $N_n = 30 - N_\phi = 21$

Ответ: иностранный студент сможет общаться с 21 аспирантом кафедры.

ГЛАВА 2. Системы счисления.

Рассмотрим основные арифметические операции: **сложение, вычитание, умножение и деление**. Правила выполнения этих операций в десятичной системе хорошо известны — это сложение, вычитание, умножение столбиком и деление углом. Эти правила применимы и ко всем другим позиционным системам счисления. Только таблицами сложения и умножения надо пользоваться особыми для каждой системы.

Сложение

При сложении цифры суммируются по разрядам, и если при этом возникает избыток, то он переносится влево.

Сложение в двоичной системе

+	0	1
0	0	1
1	1	10

Пример 1. Сложите числа: 11101_2 и 1011_2 .

Решение.

$$\begin{array}{r}
 1 \quad 1 \quad 1 \quad 1 \\
 1 \quad 1 \quad 1 \quad 0 \quad 1_2 \\
 + \quad 1 \quad 0 \quad 1 \quad 1_2 \\
 \hline
 1 \quad 0 \quad 1 \quad 0 \quad 0 \quad 0_2
 \end{array}$$

Ответ: $11101_2 + 1011_2 = 101000_2$.

Пример 2. Сложите числа: 1111_2 и 1

$$\begin{array}{r}
 1 \quad 1 \quad 1 \\
 \hline
 \end{array}$$

$$\begin{array}{r}
 1 \ 1 \ 1 \ 1 \\
 + \quad \quad \quad 1 \\
 \hline
 1 \ 0 \ 0 \ 0 \ 0
 \end{array}$$

Ответ: $1111_2 + 1_2 = 10000_2$

Пример 3. Сложите числа: $101,011_2$ и $1,11_2$

$$\begin{array}{r}
 1 \ 1 \ 1 \\
 1 \ 0 \ 1, \ 0 \ 1 \ 1 \\
 + \quad \quad \quad 1, \ 1 \ 1 \\
 \hline
 1 \ 1 \ 1, \ 0 \ 0 \ 1
 \end{array}$$

Ответ: $101,011_2 + 1,11_2 = 111,001_2$

Сложение в восьмеричной системе

+	0	1	2	3	4	5	6	7
0	0	1	2	3	4	5	6	7
1	1	2	3	4	5	6	7	10
2	2	3	4	5	6	7	10	11
3	3	4	5	6	7	10	11	12
4	4	5	6	7	10	11	12	13
5	5	6	7	10	11	12	13	14
6	6	7	10	11	12	13	14	15
7	7	10	11	12	13	14	15	16

Пример 4. Сложите числа: 173_8 и 56_8 .

Решение.

$$\begin{array}{r}
 1 \ 1 \\
 1 \ 7 \ 3_8 \\
 + \quad \quad 5 \ 6_8 \\
 \hline
 2 \ 5 \ 1_8
 \end{array}$$

Ответ: $173_8 + 56_8 = 251_8$

Пример 5. Сложите числа: $215, 4_8$ и $73, 61_8$.

$$\begin{array}{r}
 1 \ 1 \ 1 \\
 2 \ 1 \ 5, \ 4_8 \\
 + \quad \quad 7 \ 3, \ 6 \ 1_8 \\
 \hline
 3 \ 1 \ 1, \ 2 \ 1_8
 \end{array}$$

Ответ: $215, 4_8 + 73, 61_8 = 311, 21_8$.

Сложение в шестнадцатеричной системе

+	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10
2	2	3	4	5	6	7	8	9	A	B	C	D	E	F	10	11
3	3	4	5	6	7	8	9	A	B	C	D	E	F	10	11	12
4	4	5	6	7	8	9	A	B	C	D	E	F	10	11	12	13
5	5	6	7	8	9	A	B	C	D	E	F	10	11	12	13	14
6	6	7	8	9	A	B	C	D	E	F	10	11	12	13	14	15
7	7	8	9	A	B	C	D	E	F	10	11	12	13	14	15	16
8	8	9	A	B	C	D	E	F	10	11	12	13	14	15	16	17
9	9	A	B	C	D	E	F	10	11	12	13	14	15	16	17	18
A	A	B	C	D	E	F	10	11	12	13	14	15	16	17	18	19
B	B	C	D	E	F	10	11	12	13	14	15	16	17	18	19	1A
C	C	D	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B
D	D	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C
E	E	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D
F	F	10	11	12	13	14	15	16	17	18	19	1A	1B	1C	1D	1E

Пример 6: Сложите числа: $A97_{16}$ и $1D_{16}$.

Решение.

$$\begin{array}{r} 1 \\ A \ 9 \ 7_{16} \\ + \quad 1 \ D_{16} \\ \hline A \ B \ 4_{16} \end{array}$$

Ответ: $A97_{16} + 1D_{16} = AB4_{16}$.

Пример 7. Сложите числа: $7C_8$, 8_{16} и $2D, E_{16}$.

Решение.

$$\begin{array}{r} 1 \quad 1 \\ 7 \ C, \ 8_{16} \\ + \quad 2 \ D, \ E_{16} \\ \hline A \ A, \ 6_{16} \end{array}$$

Ответ: $7C_8, 8_{16} + 2D, E_{16} = AA,6_{16}$.

Вычитание.

Вычитание в двоичной системе

При выполнении операции вычитания всегда из большего по абсолютной величине числа вычитается меньшее и у результата ставится соответствующий знак (* означает заем из старшего разряда).

Пример 8. Вычтите число 111_2 из числа 1011_2 .

$$\begin{array}{r} 1 \ 0 \ 1 \ 1_2 \\ - \quad 1 \ 1 \ 1_2 \\ \hline 1 \ 0 \ 0_2 \end{array}$$

Ответ: $1011_2 - 10,1_2 = 1001,1_2$.

Пример 9. Вычтите число $10,1_2$ из числа 1100_2 .

$$\begin{array}{r} * \quad * \\ 1 \ 1 \ 0 \ 0_2 \\ - \quad \quad 1 \ 0, \ 1_2 \\ \hline 1 \ 0 \ 0 \ 1, \ 1_2 \end{array}$$

Ответ: $1100_2 - 10,1_2 = 1001,1_2$.

Пример 10. Вычтите число 1011_2 из числа 11_2 .

По абсолютной величине большее из чисел – число 1011_2 . Вычтем из него число 11_2 .

$11_2 - 1011_2 = -(1011_2 - 11_2)$

$$\begin{array}{r} 1 \ 0 \ 1 \ 1_2 \\ - \quad \quad 1 \ 1_2 \\ \hline - \ 1 \ 0 \ 0 \ 0_2 \end{array}$$

Ответ: $11_2 - 1011_2 = -1000_2$.

Вычитание в восьмеричной системе

Пример 11. Вычтите единицу из числа 100_8

$$\begin{array}{r} * \\ 1 \ 0 \ 0_8 \\ - \quad \quad 1_8 \\ \hline 7 \ 7_8 \end{array}$$

Ответ: $100_8 - 1_8 = 77_8$.

Пример 12. Вычтите число $73,6_8$ из числа $311,2_8$.

$$\begin{array}{r} * \quad * \quad * \\ 3 \ 1 \ 1, \ 2_8 \\ - \quad 7 \ 3, \ 6_8 \\ \hline 2 \ 1 \ 5, \ 4_8 \end{array}$$

Ответ: $311,2_8 - 73,6_8 = 215,4_8$.

Вычитание в шестнадцатеричной системе

Пример 13. Вычтите единицу из числа 100_{16} .

$$\begin{array}{r} * \\ 1 \ 0 \ 0_{16} \\ - \quad \quad 1_{16} \\ \hline F \ F_{16} \end{array}$$

Ответ: $100_{16} - 1_{16} = FF_{16}$.

Пример 14. Вычтите число $2A, C_{16}$ из числа $D8, 4_{16}$.

$$\begin{array}{r} * \quad * \\ D \ 8, \ 4_{16} \\ - \ 2 \ A, \ C_{16} \\ \hline A \ D, \ 8_{16} \end{array}$$

Ответ: $D8, 4_{16} - 2A, C_{16} = AD, 8_{16}$

Умножение

Операция умножения выполняется с использованием таблицы умножения по обычной схеме, применяемой в десятичной системе счисления, с последовательным умножением множимого на очередную цифру множителя.

Умножение на ноль можно не производить, а все оставшиеся справа нули, не участвующие в умножении, приписываются справа к результату умножения.

Умножение в двоичной системе

В двоичной системе счисления операция умножения сводится к сдвигам множимого и сложению промежуточных результатов.

Таблица умножения

×	0	1
0	0	0
1	0	1

Пример 15. Выполните умножение чисел 1011_2 и 101_2 .

$$\begin{array}{r} \quad \quad 1 \ 0 \ 1 \ 1_2 \\ \times \quad \quad 1 \ 0 \ 1_2 \\ \hline \quad \quad 1 \ 0 \ 1 \ 1 \\ + \ 1 \ 0 \ 1 \ 1 \\ \hline 1 \ 1 \ 0 \ 1 \ 1 \ 1_2 \end{array}$$

Ответ: $1011_2 \times 101_2 = 110111_2$

Пример 16. Выполните умножение чисел 1011_2 и 111_2 .

$$\begin{array}{r} \quad \quad 1 \ 0 \ 1 \ 1_2 \\ \times \quad \quad 1 \ 1 \ 1_2 \\ \hline \quad \quad 1 \ 0 \ 1 \ 1 \\ + \quad \quad 1 \ 0 \ 1 \ 1 \\ + \quad \quad 1 \ 0 \ 1 \ 1 \\ \hline 1 \ 0 \ 0 \ 1 \ 1 \ 0 \ 1_2 \end{array}$$

Ответ: $1011_2 \times 111_2 = 1001101_2$

Пример 17. Выполните умножение чисел 10100_2 и 1010_2 .

$$\begin{array}{r} \quad \quad 1 \ 0 \ 1 \ 0 \ 0_2 \\ \times \quad \quad 1 \ 0 \ 1 \ 0_2 \\ \hline \quad \quad 1 \ 0 \ 1 \ 0 \\ + \ 1 \ 0 \ 1 \ 0 \\ \hline 1 \ 1 \ 0 \ 0 \ 1 \ 0 \ 0 \ 0_2 \end{array}$$

Ответ: $10100_2 \times 1010_2 = 11001000_2$.

Умножение в восьмеричной системе

Таблица умножения

×	0	1	2	3	4	5	6	7
0	0	0	0	0	0	0	0	0

Формат числа в байтах	Диапазон	
	Запись с порядком	Обычная запись
1	$-2^7 \dots 2^7 - 1$	$-128 \dots 127$
2	$-2^{15} \dots 2^{15} - 1$	$-32768 \dots 32767$
4	$-2^{31} \dots 2^{31} - 1$	$-2147483648 \dots 2147483647$

Рассмотрим особенности записи целых чисел со знаком на примере однобайтового формата, при котором для знака отводится один разряд, а для цифр абсолютной величины – семь разрядов.

В компьютерной технике применяются три формы записи (кодирования) целых чисел со знаком: *прямой код*, *обратный код*, *дополнительный код*.

Последние две формы применяются особенно широко, так как позволяют упростить конструкцию арифметико-логического устройства компьютера путем замены разнообразных арифметических операций операцией сложения.

Положительные числа в прямом, обратном и дополнительном кодах изображаются одинаково — двоичными кодами с цифрой 0 в знаковом разряде. Например:

Число $1_{10} = 1_2$	Число $127_{10} = 1111111_2$
0 0 0 0 0 0 0 1	0 1 1 1 1 1 1 1
Знак числа "+"	Знак числа "+"

Отрицательные числа в прямом, обратном и дополнительном кодах имеют разное изображение.

1. Прямой код. В знаковый разряд помещается цифра 1, а в разряды цифровой части числа — двоичный код его абсолютной величины. Например:

Прямой код числа - 1	Прямой код числа - 127
1 0 0 0 0 0 0 1	1 1 1 1 1 1 1 1
Знак числа "-"	Знак числа "-"

2. Обратный код. Получается инвертированием всех цифр прямого кода абсолютной числа, за исключением разряда знака: нули заменяются единицами, а единицы — нулями. Например:

Число: -1	Число: -127
Код модуля числа: 0 0000001	Код модуля числа: 0 1111111
Обратный код числа: 1 1111110	Обратный код числа: 1 0000000
1 1 1 1 1 1 1 0	1 0 0 0 0 0 0 0

3. Дополнительный код. Получается из обратного кода при помощи прибавления единицы к его младшему разряду. Например:

Дополнительный код числа - 1	Дополнительный код числа - 127
1 1 1 1 1 1 1 1	1 0 0 0 0 0 0 1

Алгоритм перевода дополнительного кода отрицательного числа в прямой аналогичен алгоритму перевода прямого кода в дополнительный.

Обычно *отрицательные* десятичные числа при вводе в машину *автоматически* преобразуются в *обратный* или *дополнительный* двоичный код и в таком виде хранятся, перемещаются и участвуют в операциях. При выводе таких чисел из машины происходит *обратное преобразование* в отрицательные десятичные числа.

Вещественными числами (в отличие от целых) в компьютерной технике называются числа, имеющие дробную часть.

Для удобства отображения чисел, принимающих значения из достаточно широкого диапазона (то есть, как очень маленьких, так и очень больших), используется форма записи чисел с порядком основания системы счисления. Например, десятичное число 1,25 можно в этой форме представить так:

$$1,25 \cdot 10^0 = 0,125 \cdot 10^1 = 0,0125 \cdot 10^2 = \dots,$$

или так:

$$12,5 \cdot 10^{-1} = 125,0 \cdot 10^{-2} = 1250,0 \cdot 10^{-3} = \dots$$

Любое число N в системе счисления с основанием q можно записать в виде $N = M \cdot q^p$, где M называется мантиссой числа, а p — порядком. Такой способ записи чисел называется представлением с плавающей точкой.

Если “плавающая” точка расположена в мантиссе перед первой значащей цифрой, то при фиксированном количестве разрядов, отведённых под мантиссу, обеспечивается запись максимального количества значащих цифр числа, то есть максимальная точность представления числа в машине. Из этого следует: Мантисса должна быть правильной дробью, первая цифра которой отлична от нуля: M из $[0.1, 1)$. Такое, наиболее выгодное для компьютера, представление вещественных чисел называется *нормализованным*.

Мантиссу и порядок q -ичного числа принято записывать в системе с основанием q , а само основание — в десятичной системе.

Примеры нормализованного представления:

Десятичная система

Двоичная система

$$753,15 = 0,75315 \cdot 10^3; \quad -101,01 = -0,10101 \cdot 2^{11} \text{ (порядок } 11_2 = 3_{10})$$

$$-0,000034 = -0,34 \cdot 10^{-4}; \quad -0,000011 = 0,11 \cdot 2^{-100} \text{ (порядок } -100_2 = -410)$$

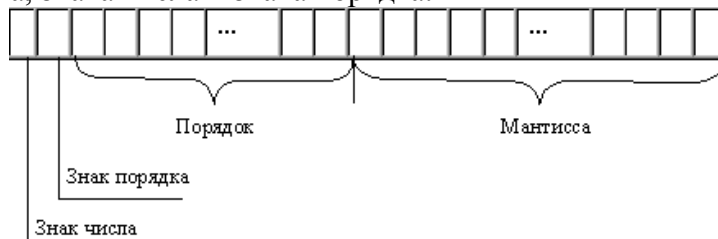
Вещественные числа в компьютерах различных типов записываются по-разному. При этом компьютер обычно предоставляет программисту возможность выбора из нескольких числовых форматов наиболее подходящего для конкретной задачи — с использованием четырех, шести, восьми или десяти байтов.

В качестве примера приведем характеристики форматов вещественных чисел, используемых IBM-совместимыми персональными компьютерами:

Форматы вещественных чисел	Размер в байтах	Примерный диапазон абсолютных значений	Количество значащих десятичных цифр
Одинарный	4	$10^{-45} \dots 10^{38}$	7 или 8
Вещественный	6	$10^{-39} \dots 10^{38}$	11 или 12
Двойной	8	$10^{-324} \dots 10^{308}$	15 или 16
Расширенный	10	$10^{-4932} \dots 10^{4932}$	19 или 20

Из этой таблицы видно, что форма представления чисел с плавающей точкой позволяет записывать числа с высокой точностью и из весьма широкого диапазона.

При хранении числа с плавающей точкой отводятся разряды для мантиссы, порядка, знака числа и знака порядка:

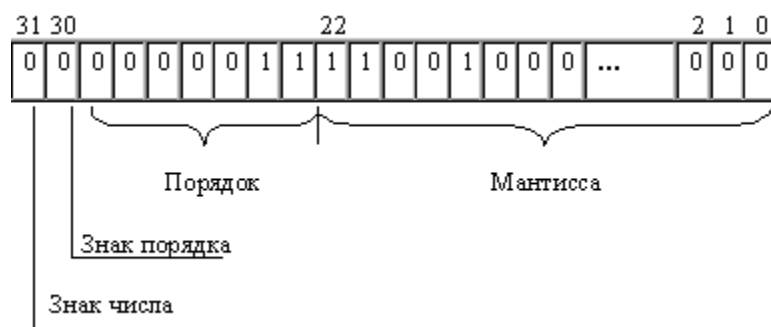


Чем больше разрядов отводится под запись мантиссы, тем выше точность представления числа.

Чем больше разрядов занимает порядок, тем шире диапазон от наименьшего

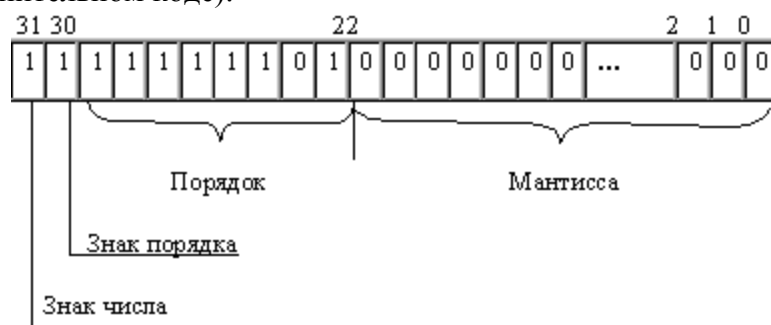
отличного от нуля числа до наибольшего числа, представимого в машине при заданном формате. Покажем на примерах, как записываются некоторые числа в нормализованном виде в четырехбайтовом формате с семью разрядами для записи порядка.

Пример 1. Число $6,25_{10} = 110,01_2 = 0,11001 \cdot 2^{11}$:



Пример 2. Число $-0,125_{10} = -0,0012 = -0,1 \cdot 2^{-10}$ (отрицательный порядок записан в

дополнительном коде):



ГЛАВА 4. Алгебра логики.

Математическая логика - это современная форма логики, которая полностью опирается на формальные математические методы. Частью математической логики является *алгебра логики* - раздел, изучающий строение сложных логических высказываний и способы установления их истинности с помощью алгебраических методов.

Алгебру логики иногда называют алгеброй Буля, или *булевой алгеброй*, по имени ее создателя - Джоржа Буля - английского математика (кстати, отца Этель Л. Войнич, написавшей "Овод"). Основные результаты теории были сформулированы им в книге "Исследование законов мышления", изданной в 1854 году. Во введении Буль писал, что назначение трактата - исследовать основные законы тех операций ума, посредством которых производится рассуждение, выразить их на символическом языке некоторого исчисления и на этой основе установить науку логику и построить ее метод.

Основным объектом изучения математической логики являются элементарные *высказывания*. Под термином "высказывание" мы будем понимать повествовательное предложение. При этом нас будет интересовать не построение: подлежащее - сказуемое - дополнение, а характерные свойства рассматриваемых образований: являются они *истинными* или *ложными*. Высказывания отличаются от других языковых образований тем, что мы можем присвоить им определенное значение истинности - "*истина*", если они истинны, значение "*ложь*", если они ложны. При этом мы исходим из "принципа исключенного третьего", или "третьего не дано", который состоит в том, что каждое высказывание или истинно, или ложно, и других возможностей нет. Ситуация, когда ни истинно, ни ложно, была бы и в обычном смысле неразрешимой. Этот принцип называют *принципом двужначности*.

Все научные знания (законы и явления физики, химии и биологии, математические теоремы и т.д.), события повседневной жизни, ситуации, возникающие в процессах управления, формулируются в виде высказываний.

Примеры высказываний:

"35 делится на 7",

"Москва - столица России".

Все они имеют значение истинности "истина".

Следующие высказывания имеют значение истинности "ложь": "Мышь больше слона",

"Молодые лошади называются щенятами",

"6 больше 8".

Повелительные ("Войдите, пожалуйста!"), вопросительные ("Знаешь ли ты информатику?") и бессмысленные предложения не являются высказываниями.

Если логика имеет дело со смыслом высказываний, то в алгебре логики работают с формулами. Любое элементарное высказывание обозначается малой буквой латинского алфавита (в нашем повествовании - x_i), совершенно так же, как в элементарной алгебре обозначаются величины, когда мы абстрагируемся от того, какие именно предметы изучаются, нас интересует только их количество и соотношение между ними.

Поскольку высказывание может принимать одно из двух значений, то говорят о "переменных высказываниях". Это означает, что рассматривается не только конкретно определенное высказывание x_i , но также некоторая *логическая переменная* x_i , которую можно использовать для обозначения произвольного высказывания. Замещение переменной конкретным высказыванием означает предоставления одного из значений "истина" или "ложь".

Итак, в алгебре логики в каждом высказывании мы будем отвлекаться от всех особенностей высказывания, кроме одного - истинно оно или ложно. Истинное высказывание условно обозначается единицей (если x_1 - высказывание "35 делится на 7", то $x_1 = 1$), а ложное - нулем (если x_2 - высказывание "мышь больше слона", то $x_2 = 0$).

Таким образом, диапазон изменения переменного - x_i в алгебре логики существенно меньше, чем изменение переменного в элементарной алгебре: оно принимает только одно из двух значений - или 1, или 0.

Из элементарных высказываний с помощью логических связок "и", "или", "не", "если - то" и других (*логических операций*) строятся *сложные высказывания* - *формулы* (или *функции*) *алгебры логики*.

Способы построения новых высказываний из заданных с помощью логических связок, их преобразования и установления истинности изучаются в логике высказываний с помощью алгебраических методов.

Сущность алгебраического подхода к логике поясним на примере элементарной алгебры - алгебры арифметики. Основные объекты алгебры арифметики - выражения (формулы), состоящие из букв, знаков операций и скобок. С формулами производят процедуры двух типов: вычисления и преобразования.

Вычисления - вместо букв подставляют числа, знаки указывают действия, а скобки их порядок. Каждая формула задает функцию. Переменные - буквы формулы.

Преобразование формулы происходит так: исходная формула или ее часть F_1 заменяется другой, в результате получается новая формула F_2 , которая *эквивалентна* F_1 (т.е. при любых значениях аргументов F_1 и F_2 дают один и тот же результат). Преобразование выражений производится на основе законов арифметики, а также полученных из них соотношений типа $(a + b)^2 = a^2 + 2ab + b^2$ и т.п.

По аналогии строится алгебра логики. Она рассматривает логические выражения как алгебраические, которые можно преобразовать по определенным правилам. Разница заключается в том, что в выражениях алгебры логики переменные являются логическими (0 и 1). Знаки операций обозначают логические операции (логические связки).

В логических задачах исходными данными являются суждения, подчас неожиданные и нередко весьма запутанные. Высказывания и связи между ними бывают иногда столь противоречивыми, что такие "твердые орешки" не под силу "раскусить" и вдумчивому математику.

В данных случаях большую помощь может оказать ЭВМ как необходимый инструмент эффективного решения логических задач с многими переменными. В настоящее время нет ни одного языка программирования, который не включал бы логических операций.

В алгебре логики основными (элементарными) операциями являются *отрицание*, *логическое сложение (дизъюнкция)*, *логическое умножение (конъюнкция)*, *импликация*, *эквивалентность*.

Отрицание. Операция отрицания соответствует в обычном языке частице *не*. Функция, получающаяся в результате применения операции отрицания к высказыванию x , обозначается так: $\bar{x}$. Читается "не x ". Заметим, что в математической логике до сих пор нет единых обозначений, поэтому вместо $\bar{x}$ можно встретить и другие обозначения, смысл которых тот же самый, например, $\neg x$ или $\sim x$.

Новое сложное высказывание $\bar{x}$ - ложно при условии, что высказывание x истинно и истинно при условии, что x - ложно. Запишем эти зависимости в виде таблицы истинности (формальное определение дадим позднее) для операции отрицания:

x	$\bar{x}$
1	0
0	1

Пример 1. Обозначим суждение (высказывание) "Я иду в кино" через x_1 , а суждение "Я достаю билеты в театр" через x_2 . Тогда в сложном суждении "Если я завтра достаю билеты в театр, то не иду в кино ($x_2 = 1, x_1 = 0$), а если я не достану билеты в театр, то пойду в кино ($x_2 = 0, x_1 = 1$)" высказывания x_1 и x_2 связаны между собой операцией отрицания: $x_1 = \bar{x}_2$.

Логическое умножение (конъюнкция). Обозначается $x_1 \wedge x_2$. Другие обозначения: $x_1 * x_2, x_1 \& x_2$. Читается "и x_1 и x_2 ". Это сложное суждение, которое истинно только в том случае, когда и суждение x_1 , и суждение x_2 истинны. В остальных случаях сложное высказывание ложно.

Таблица истинности для конъюнкции (логического умножения):

x_1	x_2	$x_1 \wedge x_2$
0	0	0
0	1	0
1	0	0
1	1	1

Пример 2. Рассмотрим условия, необходимые для получения человеком водительских прав. Суждение "Человек получает права водителя" обозначим через y . Для получения прав необходимо получить положительное заключение медицинской комиссии о состоянии здоровья. Сформулируем высказывание "Человек здоров" и обозначим его x_1 . Кроме здоровья, надо еще сдать два экзамена: по вождению и по правилам дорожного движения (ПДД). Высказывания: x_2 - "Экзамен по вождению сдан успешно"; x_3 - "Экзамен по ПДД сдан успешно". В символической записи имеем:

$$y = x_1 \wedge x_2 \wedge x_3 \text{ (или } y = x_1 x_2 x_3 \text{)}.$$

Суждение y будет истинным ($y = 1$) только в том случае, если истинны все три суждения: $x_1 = 1, x_2 = 1$ и $x_3 = 1$. При всех других комбинациях $y = 0$, то есть высказывание y ложно: человек не получит прав водителя.

Логическое сложение (дизъюнкция). Обозначается $x_1 \vee x_2$. Читается "икс один или икс два". Знак " $\vee$ " взят из латинского языка, в котором есть союз "Vel", означающий или то, или другое, или то и другое вместе. Vel более точно определяет суть логического сложения, чем русский союз или, так как последний, кроме значения или то, или другое, или то и другое вместе, имеет еще и другое значение - или только то, или только другое.

Логическое сложение выражает суждение (сложное высказывание), которое истинно в том случае, если хотя бы одно из суждений истинно.

Таблица истинности логического сложения:

x_1	x_2	$x_1 \vee x_2$
0	0	0
0	1	1
1	0	1
1	1	1

Пример 3. Боря давно хотел иметь книгу "Основы информатики и вычислительной техники", где рассматриваются вопросы алгебры логики. Он попросил своих товарищей Андрея и Валерия, чтобы они обязательно купили для него эту книгу, если она им попадется. У Бориса будет книга (суждение y) при выполнении равенства:

$$y = x_1 \vee x_2,$$

где x_1 - высказывание "Андрей купил книгу",

x_2 - высказывание "Валерий купил книгу".

Высказывание $y = x_1 \vee x_2$, не исключает случая, что книгу купят и Андрей, и Валерий.

Импликация (следование). Обозначается $x_1 \rightarrow x_2$. Логически реализует связку естественной речи "если ..., то...". Импликация определяется следующим образом:

x_1	x_2	$x_1 \rightarrow x_2$
0	0	1
0	1	1
1	0	0
1	1	1

Например, в профессиональной речи следователь часто логически рассуждает примерно так: "Если кража произошла до полуночи, то наверняка это совершил Вольдемар". Нетрудно заметить, что два высказывания: x_1 - "Кража произошла до полуночи" и x_2 - "Вор - Вольдемар" связаны операцией импликации $x_1 \rightarrow x_2$.

Данное умозаключение будет истинным, если при истинном x_1 истинно x_2 . Логическое выражение будет ложным, если при истинном x_1 оказывается ложным x_2 .

Замечание. Если $x_1 = 0$ (высказывание - предположение ложно, или, говорят, ложная посылка), то x_2 (следствие) может быть как истинным ("Вор - Вольдемар", $x_2 = 1$), так и ложным ("Вор - не Вольдемар", $x_2 = 0$). Но логического смысла выражение не имеет. Данное положение вещей выражается с помощью поговорки "ex falso quod libet", или "из ложной посылки можно вывести всякое".

В обычной речи такие высказывания не имеют значения, поскольку выражения типа "Если кто-то сумасшедший, то 5 меньше 2" рассматриваются как бессмысленные. Это связано с тем, что в обычной речи предусматривается содержательная связь, причинные отношения между посылкой (причиной) x_1 и заключением (следствием) x_2 , которые здесь не заданы. Такая содержательная связь не может быть описана с помощью логики высказываний.

Эквиваленция (равносильность). Эквиваленцией (или эквивалентностью) двух высказываний x_1 и x_2 называется новое высказывание, которое считается истинным, когда оба высказывания x_1 и x_2 либо одновременно истинны, либо одновременно ложны, и ложным - во всех остальных случаях.

Эквиваленция высказываний x_1 и x_2 обозначается символом $x_1 \leftrightarrow x_2$ (или $x_1 = x_2$, или $x_1 \sim x_2$), читается "для того, чтобы x_1 , необходимо и достаточно, чтобы x_2 " или " x_1 тогда и только тогда, когда x_2 ". Высказывания x_1 и x_2 называются членами эквивалентности.

Логические значения операции эквивалентности описываются следующей таблицей истинности.

x_1	x_2	$x_1 \leftrightarrow x_2$
0	0	1
0	1	0
1	0	0
1	1	1

Например, эквиваленция "Треугольник SPQ с вершиной S и основанием PQ равнобедренный тогда и только тогда, когда угол P равен углу Q " является истинной, так как высказывания "Треугольник SPQ с вершиной S и основанием PQ равнобедренный" и "В треугольнике SPQ с вершиной S и основанием PQ угол P равен углу Q " либо одновременно истинны, либо одновременно ложны.

Эквиваленция играет важную роль в математических доказательствах. Известно, что значительное число теорем формулируется в форме необходимых и достаточных условий, то есть в форме эквиваленции. В этом случае, зная об истинности или ложности одного из двух членов эквиваленции и доказав истинность самой эквиваленции, мы заключаем об истинности или ложности второго члена эквиваленции.

Итак, мы дали определения элементарных логических операций. Они позволяют строить сложные (составные) высказывания на основе заданных высказываний.

Пусть x_1, x_2, x_3, x_4 - переменные высказывания (логические переменные), тогда следующие высказывания:

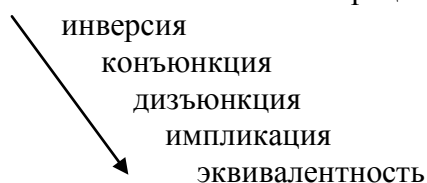
$$x_1 \wedge \bar{x}_2 \rightarrow x_3$$

$$\overline{x_1 \vee x_3} \wedge (x_1 \leftrightarrow x_4) \rightarrow x_2$$

$$(x_1 \vee x_2) \wedge (x_2 \wedge (x_1 \vee x_3 \vee x_4))$$

представляют формулы алгебры высказываний.

Так же, как и в алгебре арифметики, в алгебре логики устанавливается приоритет выполнения логических операций. Они упорядочены в следующей последовательности:



При этом отрицание является наиболее сильной операцией, эквивалентность - самой слабой. С учетом указанного порядка две формулы:

$$((x_1 \wedge x_2) \rightarrow (x_3 \vee x_4)) \leftrightarrow ((x_1 \vee x_2) \wedge x_3) \rightarrow x_4$$

и

$$x_1 \wedge x_2 \rightarrow x_3 \vee x_4 \leftrightarrow (x_1 \vee x_2) \wedge x_3 \rightarrow x_4$$

имеют одинаковые значения.

Истинность составных высказываний (логических формул), образованных в результате выполнения логических операций над простыми высказываниями, зависит от истинности исходных высказываний.

В алгебре логики выполняются некоторые законы, позволяющие производить тождественные преобразования логических выражений. Эти законы сведены в таблицу 6.1.

Таблица 6.1

Закон	Для «ИЛИ»	Для «И»
Переместительный	$X \vee Y = Y \vee X$	$X \cdot Y = Y \cdot X$
Сочетательный	$X \vee (Y \vee Z) = (X \vee Y) \vee Z$	$X \cdot (Y \cdot Z) = (X \cdot Y) \cdot Z$
Распределительный	$X \cdot (Y \vee Z) = X \cdot Y \vee X \cdot Z$	$X \vee (Y \cdot Z) = (X \vee Y) \cdot (X \vee Z)$
Правила де Моргана	$\overline{X \vee Y} = \overline{X} \cdot \overline{Y}$	$\overline{X \cdot Y} = \overline{X} \vee \overline{Y}$
Идемпотенции	$X \vee X = X$	$X \cdot X = X$
Поглощения	$X \vee (X \cdot Y) = X$	$X \cdot (X \vee Y) = X$
Склеивания	$(X \cdot Y) \vee (\overline{X} \cdot Y) = Y$	$(X \vee Y) \cdot (\overline{X} \vee Y) = Y$
Операция переменной с ее инверсией	$X \vee \overline{X} = 1$	$X \cdot \overline{X} = 0$
Операция с константами	$X \vee 0 = X; \quad X \vee 1 = 1$	$X \cdot 0 = 0; \quad X \cdot 1 = X$
Двойного отрицания	$\overline{\overline{X}} = X$	

Согласно определению, *таблица истинности логической формулы выражает соответствие между всевозможными наборами значений переменных и значениями формулы.*

Для формулы, которая содержит две переменные, таких наборов значений переменных всего четыре: (0, 0), (0, 1), (1, 0), (1, 1).

Если формула содержит три переменные, то возможных наборов значений переменных восемь: (0, 0, 0), (0, 0, 1), (0, 1, 0), (0, 1, 1), (1, 0, 0), (1, 0, 1), (1, 1, 0), (1, 1, 1).

Количество наборов для формулы с четырьмя переменными равно шестнадцати и т.д.

Удобной формой записи при нахождении значений формулы является таблица, содержащая кроме значений переменных и значений формулы также и значения промежуточных формул.

Задача 6.1. Выяснить, является ли формула $\overline{X} \cdot Y \vee \overline{X \vee Y} \vee X$ тождественно истинной.

Решение

Составим таблицу истинности для формулы $\overline{X} \cdot Y \vee \overline{X \vee Y} \vee X$, которая содержит две переменные **X** и **Y**. В первых двух столбцах таблицы запишем четыре возможных пары значений этих переменных, в последующих столбцах — значения промежуточных формул и в последнем столбце — значение формулы. В результате получим таблицу 6.2. Из таблицы видно, что при всех наборах значений переменных **x** и **y** формула $\overline{X} \cdot Y \vee \overline{X \vee Y} \vee X$ принимает значение 1, то есть является тождественно истинной.

Таблица 6.2

Переменные		Промежуточные логические формулы					Формула
X	Y	$\overline{X}$	$\overline{X} \cdot Y$	$X \vee Y$	$\overline{X \vee Y}$	$\overline{X} \cdot Y \vee \overline{X \vee Y}$	$\overline{X} \cdot Y \vee \overline{X \vee Y} \vee X$
0	0	1	0	0	1	1	1
0	1	1	1	1	0	1	1

1	0	0	0	1	0	0	1
1	1	0	0	1	0	0	1

Задача 6.2. Выяснить, является ли формула $\overline{X \vee \bar{Y}} \cdot (X \cdot \bar{Y})$ тождественно ложной.

Решение

Таблица истинности для формулы $\overline{X \vee \bar{Y}} \cdot (X \cdot \bar{Y})$ показана в таблице 6.3. Из таблицы видно, что при всех наборах значений переменных x и y формула $\overline{X \vee \bar{Y}} \cdot (X \cdot \bar{Y})$ принимает значение 0, то есть является тождественно ложной.

Таблица 6.3

Переменные		Промежуточные логические формулы				Формула
X	Y	$X \vee Y$	$\overline{X \vee Y}$	$\bar{Y}$	$X \cdot \bar{Y}$	$\overline{X \vee \bar{Y}} \cdot (X \cdot \bar{Y})$
0	0	0	1	1	0	0
0	1	1	0	0	0	0
1	0	1	0	1	1	0
1	1	1	0	0	0	0

Задача 6.3. Является ли выполнимой формула $\overline{X \vee \bar{Y}} \vee \bar{X} \cdot Z$?

Решение

Таблица истинности для формулы $\overline{X \vee \bar{Y}} \vee \bar{X} \cdot Z$ дана в таблице 6.4.

Таблица 6.4

Переменные			Промежуточные логические формулы					Формула
X	Y	Z	$\bar{Y}$	$X \vee \bar{Y}$	$\overline{X \vee \bar{Y}}$	$\bar{X}$	$\bar{X} \cdot Z$	$\overline{X \vee \bar{Y}} \vee \bar{X} \cdot Z$
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

Из таблицы видно, что формула $\overline{X \vee \bar{Y}} \vee \bar{X} \cdot Z$ в некоторых случаях принимает значение 1, а в некоторых — 0, то есть является выполнимой.

Равносильные преобразования логических формул имеют то же назначение, что и преобразования формул в обычной алгебре. Они служат для упрощения формул или приведения их к определённому виду путем использования основных законов алгебры логики.

Под *упрощением формулы* понимают *равносильное преобразование, приводящее к формуле, которая либо содержит по сравнению с исходной меньшее число операций либо содержит меньшее число вхождений переменных*.

Некоторые преобразования логических формул похожи на преобразования формул в обычной алгебре (вынесение общего множителя за скобки, использование переместительного и сочетательного законов и т.п.), тогда как другие преобразования основаны на свойствах, которыми не обладают операции обычной алгебры (использование распределительного закона для конъюнкции, законов поглощения, склеивания, де Моргана и др.).

Покажем на примерах некоторые приемы и способы, применяемые при упрощении логических формул:

$$1) \overline{X \vee Y} \cdot (X \cdot \overline{Y}) = \overline{X} \cdot \overline{Y} \cdot (X \cdot \overline{Y}) = \overline{X} \cdot X \cdot Y \cdot \overline{Y} = 0 \cdot Y \cdot \overline{Y} = 0 \cdot Y \cdot \overline{Y} = 0 \cdot Y = 0$$

(законы алгебры логики применяются в следующей последовательности: правило де Моргана, сочетательный закон, правило операций переменной с её инверсией и правило операций с константами);

$$2) \overline{X} \cdot Y \vee \overline{X \vee Y} \vee X = \overline{X} \cdot Y \vee \overline{X} \cdot \overline{Y} \vee X = \overline{X} \cdot (Y \vee \overline{Y}) \vee X = \overline{X} \vee X = 1$$

(применяется правило де Моргана, выносится за скобки общий множитель, используется правило операций переменной с её инверсией);

$$3) (X \vee Y) \cdot (\overline{X} \vee \overline{Y}) \cdot (\overline{X} \vee \overline{Y}) = (X \vee Y) \cdot (\overline{X} \vee \overline{Y}) \cdot (\overline{X} \vee \overline{Y}) \cdot (\overline{X} \vee \overline{Y}) = Y \overline{X}$$

(повторяется второй сомножитель, что разрешено законом идемпотенции; затем комбинируются два первых и два последних сомножителя и используется закон склеивания);

$$4) \overline{\overline{X \cdot Y \vee Z}} = \overline{\overline{X} \cdot \overline{Y} \cdot \overline{Z}} = (\overline{X} \vee \overline{Y}) \cdot Z$$

(сначала добиваемся, чтобы знак отрицания стоял только перед отдельными переменными, а не перед их комбинациями, для этого дважды применяем правило де Моргана; затем используем закон двойного отрицания);

Из этих примеров видно, что при упрощении логических формул не всегда очевидно, какой из законов алгебры логики следует применить на том или ином шаге.

Решение логических задач.

Логика широко применяется при решении логических задач. Разнообразие логических задач очень велико. Способов их решения тоже немало. Но наибольшее распространение получили следующие три способа решения логических задач:

- ~ средствами алгебры логики;
- ~ табличный;
- ~ с помощью рассуждений.

Познакомимся с ними поочередно.

1. Решение логических задач средствами алгебры логики

Обычно используется следующая схема решения:

1. изучается условие задачи;
2. вводится система обозначений для логических высказываний;
3. конструируется логическая формула, описывающая логические связи между всеми высказываниями условия задачи;
4. определяются значения истинности этой логической формулы;
5. из полученных значений истинности формулы определяются значения истинности введенных логических высказываний, на основании которых делается заключение о решении.

Задача 1 Трое друзей, болельщиков автогонок "Формула-1", спорили о результатах предстоящего этапа гонок.

— Вот увидишь, Шумахер не придет первым, — сказал Джон. Первым будет Хилл.

— Да нет же, победителем будет, как всегда, Шумахер, — воскликнул Ник. — А об Алезе и говорить нечего, ему не быть первым.

Питер, к которому обратился Ник, возмутился:

— Хиллу не видать первого места, а вот Алезе пилотирует самую мощную машину.

По завершении этапа гонок оказалось, что каждое из двух предположений двоих друзей подтвердилось, а оба предположения третьего из друзей оказались неверны. Кто выиграл этап гонки?

Решение.

Введем обозначения для логических высказываний: **Ш** — победит Шумахер; **Х** — победит Хилл; **А** — победит Алезе.

Реплика Ника "Алезе пилотирует самую мощную машину" не содержит никакого утверждения о месте, которое займёт этот гонщик, поэтому в дальнейших рассуждениях не учитывается.

Зафиксируем высказывания каждого из друзей:

Джон: $\overline{Ш} \cdot X$, Ник: $Ш \cdot \overline{А}$, Питер: $\overline{Х}$.

Учитывая то, что предположения двух друзей подтвердились, а предположения третьего неверны, запишем и упростим истинное высказывание

$$(\overline{Ш} \cdot X) \cdot (Ш \cdot \overline{А}) \cdot \overline{Х} \vee (\overline{Ш} \cdot X) \cdot (\overline{Ш} \cdot \overline{А}) \cdot \overline{Х} \vee (\overline{Ш} \cdot X) \cdot (Ш \cdot \overline{А}) \cdot \overline{Х} = (\overline{Ш} \vee \overline{Х}) \cdot Ш \cdot \overline{А} \cdot \overline{Х} = Ш \cdot \overline{А} \cdot \overline{Х}$$

Высказывание $Ш \cdot \overline{А} \cdot \overline{Х}$ истинно только при **Ш=1, А=0, Х=0**.

Ответ. Победителем этапа гонок стал Шумахер.

II. Решение логических задач табличным способом

При использовании этого способа условия, которые содержит задача, и результаты рассуждений фиксируются с помощью специально составленных таблиц.

Задача 2. В симфонический оркестр приняли на работу трёх музыкантов: Брауна, Смита и Вессона, умеющих играть на скрипке, флейте, альте, кларнете, гобое и трубе.

Известно, что:

1. Смит самый высокий;
2. играющий на скрипке меньше ростом играющего на флейте;
3. играющие на скрипке и флейте и Браун любят пиццу;
4. когда между альтистом и трубачом возникает ссора, Смит мирит их;
5. Браун не умеет играть ни на трубе, ни на гобое.

На каких инструментах играет каждый из музыкантов, если каждый владеет двумя инструментами?

Решение.

Составим таблицу и отразим в ней условия задачи, заполнив соответствующие клетки цифрами 0 и 1 в зависимости от того, ложно или истинно соответствующее высказывание. Так как музыкантов трое, инструментов шесть и каждый владеет только двумя инструментами, получается, что каждый музыкант играет на инструментах, которыми остальные не владеют. Из условия 4 следует, что Смит не играет ни на альте, ни на трубе, а из условий 3 и 5, что Браун не умеет играть на скрипке, флейте, трубе и гобое.

Следовательно, инструменты Брауна — альт и кларнет. Занесем это в таблицу 5.10, а оставшиеся клетки столбцов "альт" и "кларнет" заполним нулями.

ГЛАВА 5. Элементы теории алгоритмов.

Появление алгоритмов связывают с зарождением математики. Более 1000 лет назад (в 825 году) ученый из города Хорезма Абдулла (или Абу Джафар) Мухаммед бен Муса аль-Хорезми создал книгу по математике, в которой описал способы выполнения арифметических действий над многозначными числами. Само слово алгоритм возникло в Европе после перевода на латынь книги этого математика.

Алгоритм – описание последовательности действий (план), строгое исполнение которых приводит к решению поставленной задачи за конечное число шагов.

Если вы внимательно оглянитесь вокруг, то обнаружите множество алгоритмов которые мы с вами постоянно выполняем. Мир алгоритмов очень разнообразен. Несмотря на это, удастся выделить общие свойства, которыми обладает любой алгоритм.

Свойства алгоритмов:

1. Дискретность (алгоритм должен состоять из конкретных действий, следующих в определенном порядке);
2. Детерминированность (любое действие должно быть строго и недвусмысленно определено в каждом случае);
3. Конечность (каждое действие и алгоритм в целом должны иметь возможность завершения);
4. Массовость (один и тот же алгоритм можно использовать с разными исходными данными);
5. Результативность (отсутствие ошибок, алгоритм должен приводить к правильному результату для всех допустимых входных значениях).

На практике чаще всего встречаются следующие *формы* представления алгоритмов:

- словесная – записывается на естественном языке;
- графическая – с помощью изображения из графических символов;
- псевдокоды – полуформализованные описания алгоритмов на некотором условном алгоритмическом языке, которые включают в себя как элементы языка программирования, так и фразы естественного языка, общепринятые математические обозначения и др.;
- программная – тексты на языках программирования.

Словесный способ записи алгоритмов является описанием последовательных этапов обработки данных. Алгоритм может быть задан в произвольном изложении на естественном языке. Например, алгоритм нахождения наибольшего общего делителя двух натуральных чисел можно представить как следующую последовательность действий:

- 1) задание двух чисел;
- 2) если числа равны, то выбор любого из них в качестве ответа и остановка, в противном случае – продолжение выполнения алгоритма;
- 3) определение большего из чисел;
- 4) замена большего из чисел разностью большего и меньшего из чисел;
- 5) повтор алгоритма с шага 2.

Приведенный алгоритм используется для любых натуральных чисел и должен приводить к решению поставленной задачи.

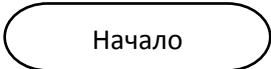
Словесный способ не имеет широкого распространения, так как обладает некоторыми недостатками:

- данные описания строго не формализуемы;
- отличаются многословностью записей;
- допускают неоднозначность толкования отдельных предписаний.

Графический способ представления алгоритмов оказывается более компактным и наглядным по сравнению со словесным. При данном виде представления алгоритм изображается в виде последовательности связанных между собой функциональных блоков, каждый из которых соответствует выполнению некоторого числа действий.

Для графического представления алгоритм использует изображение в виде последовательности связанных между собой функциональных блоков, каждый из которых соответствует выполнению одного или нескольких действий. Это графическое представление называется *схемой алгоритма*, или *блок-схемой*.

В блок-схеме каждый из типов действий (ввод исходных данных, вычисление значений выражений, проверка условий, управление повторением действий, окончание обработки и т. п.) соответствует геометрической фигуре, представленной в виде блочного символа. Блочные символы соединены линиями переходов, которые определяют очередность выполнения действий.

Вид стандартного графического объекта	Назначение
	Начало алгоритма
	Конец алгоритма
	Выполняемое действие записывается внутри прямоугольника
	Условие выполнения действий записывается внутри ромба
	Счетчик (количество повторов)
	Последовательность выполнения действий.

Псевдокод является системой обозначений и правил, которая предназначена для единообразной записи алгоритмов. Он занимает промежуточное место между естественным и формальным языками. С одной стороны, псевдокод похож на обычный естественный язык, поэтому алгоритмы могут на нем записываться и читаться как обычный текст. С другой стороны, в псевдокоде используются некоторые формальные конструкции и математическая символика, благодаря чему запись алгоритма приближается к общепринятой математической записи.

В псевдокоде не применяются строгие синтаксические правила для записи команд, которые присущи формальным языкам, что облегчает запись алгоритма на стадии его проектирования и дает возможность использовать более широкий набор команд, рассчитанный на абстрактного исполнителя. Однако в псевдокоде чаще всего имеются некоторые конструкции, присущие формальным языкам, что облегчает переход от записи на псевдокоде к записи алгоритма на формальном языке. Например, в псевдокоде, также как и в формальных языках, существуют служебные слова, смысл которых определен раз и навсегда.

Например, алгоритмы на алгоритмическом языке записываются с помощью служебных слов, представленных в таблице 1.7.

Таблица 1.7. Служебные слова алгоритмического языка.

алг(алгоритм)	сим (символьный)	дано	да	нет
арг(аргумент)	лит (литерный)	надо	для	при
рез(результат)	лог(логический)	если	от	до
нач(начало)	таб (таблица)	то	знач	выбор
кон(конец)	нц(начало цикла)	иначе	и	или
цел (целый)	кц (конец цикла)	всё	ввод	вывод
вещ (вещественный)	длин(длина)	пока	утв	не

Общий вид алгоритма:

алг название алгоритма (аргументы и результаты)

дано условия применимости алгоритма

надо цель выполнения алгоритма

нач описание промежуточных величин

последовательность команд (тело алгоритма)

кон.

Часть алгоритма от слова **алг** до слова **нач** называется заголовком, а часть, заключённая между словами **нач** и **кон** — телом алгоритма.

Единый или формальный подход к определению псевдокода не существует, поэтому используются различные псевдокоды, отличающиеся набором служебных слов и основных (базовых) конструкций.

Программная форма представления алгоритмов предполагает, что алгоритм, предназначенный для исполнения на компьютере, должен быть записан на понятном ему языке. В этом случае язык для записи алгоритмов должен быть формализован. Такой язык принято называть языком программирования, а запись алгоритма на этом языке — программой.

Стадии создания алгоритма:

1. Алгоритм должен быть представлен в форме, понятной человеку, который его разрабатывает.
2. Алгоритм должен быть представлен в форме, понятной тому объекту (в том числе и человеку), который будет выполнять описанные в алгоритме действия.

Исполнитель алгоритма - объект, который выполняет алгоритм.

Идеальными исполнителями являются машины, роботы, компьютеры...

Исполнитель способен выполнить только ограниченное количество команд. Поэтому алгоритм разрабатывается и детализируется так, чтобы в нем присутствовали только те команды и конструкции, которые может выполнить исполнитель.

Исполнитель, как и любой объект, находится в определенной среде и может выполнять только допустимые в нем действия. Если исполнитель встретит в алгоритме неизвестную ему команду, то выполнение алгоритма прекратится.

Компьютер – автоматизированный исполнитель алгоритмов.

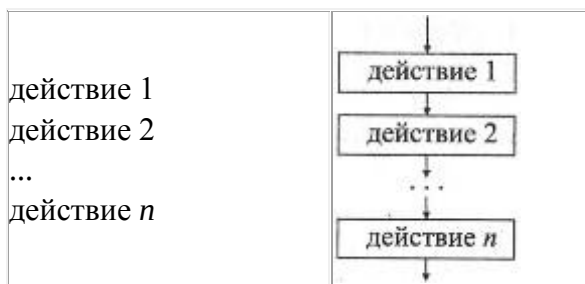
Алгоритм, записанный на «понятном» компьютеру языке программирования, называется программой.

Программирование - процесс составления программы для компьютера. Для первых ЭВМ программы записывались в виде последовательности элементарных операций. Это была очень трудоемкая и неэффективная работа. Поэтому в последствии были разработанные специальные языки программирования. В настоящее время существует множество искусственных языков для составления программ. Однако так и не удалось создать идеальный язык, который бы устроил бы всех.

Основные алгоритмические конструкции.

Структура следование. Образуется последовательностью действий, следующих одно за другим:

Алгоритмический язык	Блок-схема
----------------------	------------



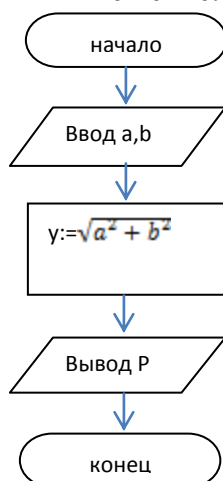
Пример 1. Определить значение переменной c после выполнения фрагмента алгоритма.

Алгоритмический язык	Блок-схема
<pre> a:=3 c:=4 c:=a+c/2 </pre>	<pre> graph TD Start(()) --> A[a:=3] A --> B[c:=4] B --> C[c:=a+c/2] C --> End(()) </pre>

Ответ: 5

Пример 2. Составьте блок-схему алгоритма вычисления периметра прямоугольного треугольника по его катетам a и b .

Решение.



Структура **ветвление**. В зависимости от результата проверки условия («да» или «нет») осуществляет выбор одного из альтернативных путей работы алгоритма. Каждый из путей ведёт к общему выходу, поэтому работа алгоритма будет продолжаться независимо от того, какой путь будет выбран. Структура «ветвление» бывает четырёх видов: «если-то»; «если-то-иначе»; «выбор»; «выбор-иначе».

Структура **«если-то»**.

Алгоритмический язык

если условие
то действия
все

Блок-схема



Пример 1. Определить значение переменной a после выполнения фрагмента алгоритма при $a=5$ и $a=10$.

Алгоритмический язык

Блок-схема

Ввод а
если $a > 5$
то $a := a + 20$
все



Ответ: 5 и 30.

Структура «если-то-иначе».

Алгоритмический язык

Блок-схема

если условие
то действия 1
иначе действия 2
все

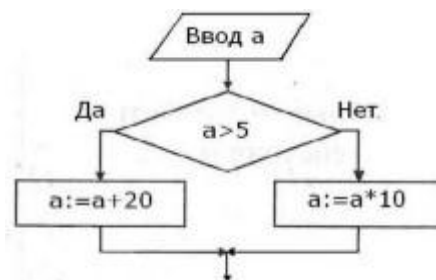


Пример 2. Определить значение переменной **а** после выполнения фрагмента алгоритма при $a=5$ и $a=10$.

Алгоритмический язык

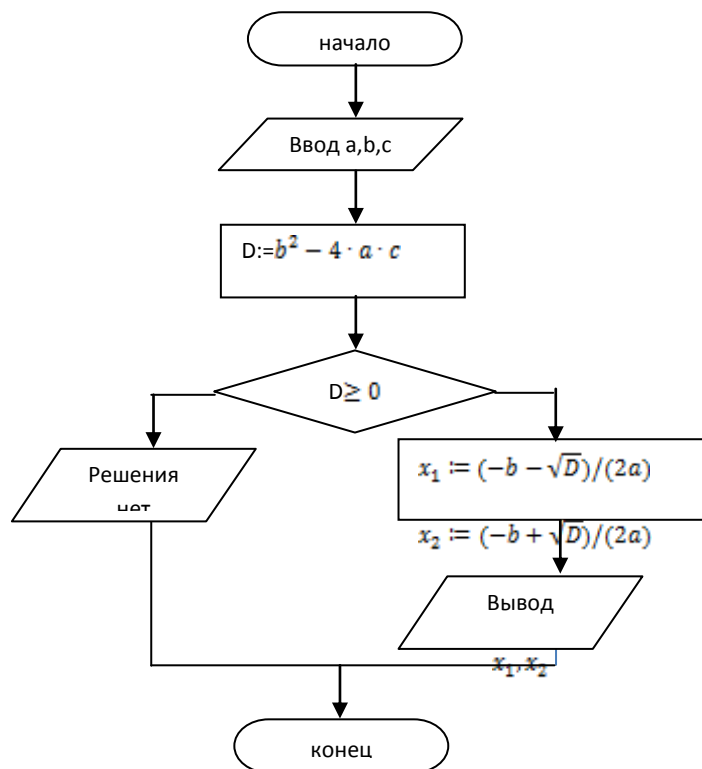
Блок-схема

Ввод а
если $a > 5$
то $a := a + 20$
иначе $a := a * 10$
все



Ответ: 50 и 30.

Пример 3. Составьте блок-схему алгоритма решения квадратного уравнения.
 Решение.



Структура «выбор».

Алгоритмический язык

Блок-схема

выбор

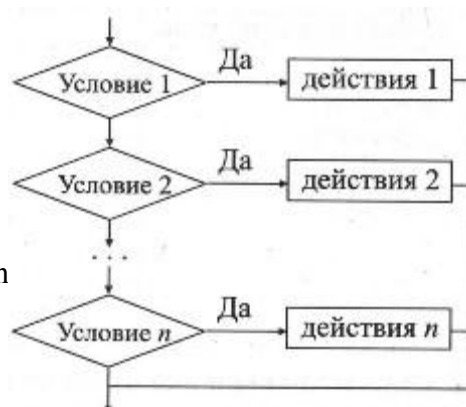
при условии 1: действия 1

при условии 2: действия 2

...

при условии n: действия n

всё

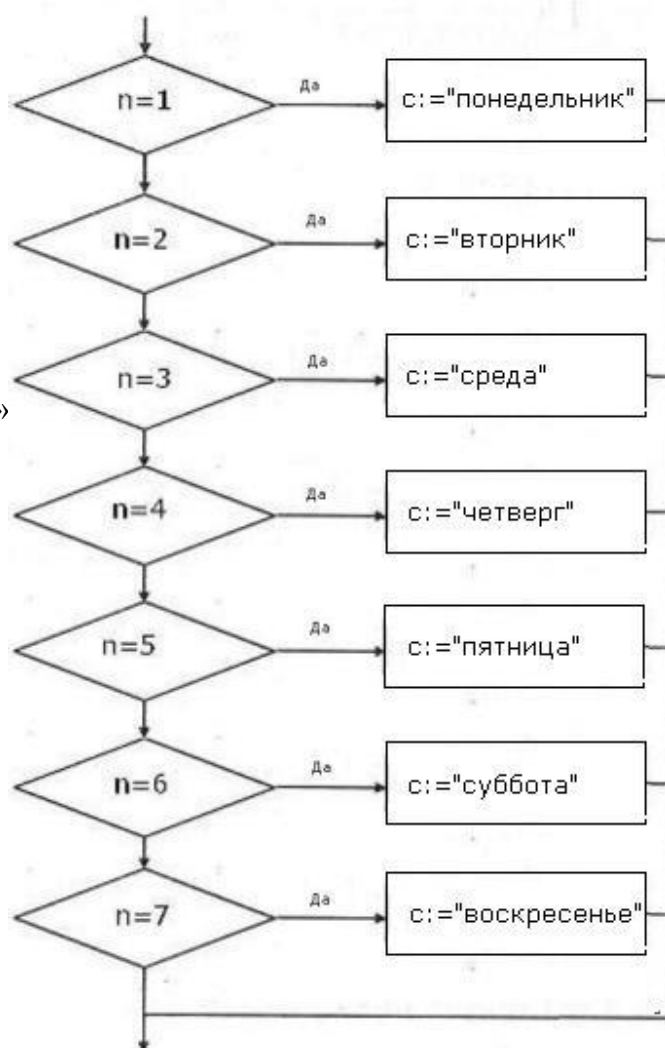


Пример 4. Дано целое число в диапазоне 1–7. Составить строку — название дня недели, соответствующее данному числу (1 — «понедельник», 2 — «вторник» и т. д.).

Алгоритмический язык

Блок-схема

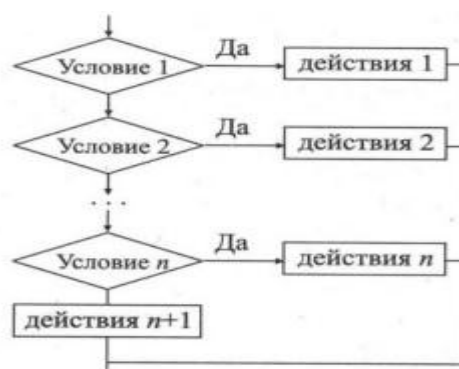
выбор
при $n=1$: $c:=\text{«понедельник»}$
при $n=2$: $c:=\text{«вторник»}$
при $n=3$: $c:=\text{«среда»}$
при $n=4$: $c:=\text{«четверг»}$
при $n=5$: $c:=\text{«пятница»}$
при $n=6$: $c:=\text{«суббота»}$
при $n=7$: $c:=\text{«воскресенье»}$
всё



Структура «**выбор-иначе**».
 Алгоритмический язык

выбор
при условие 1: действия 1
при условие 2: действия 2
 ...
при условие n : действия n
иначе действия $n+1$
всё

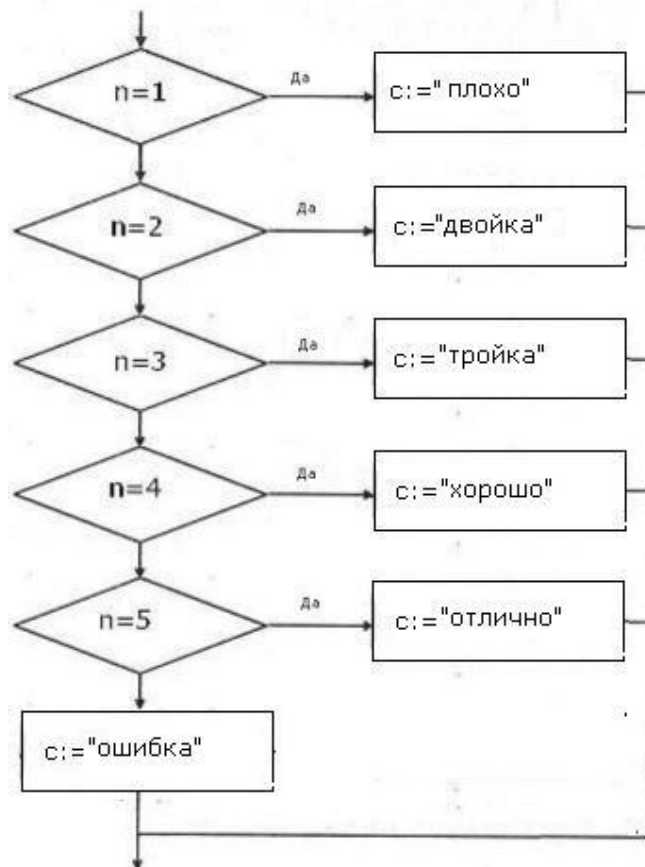
Блок-схема



Пример 4. Дано целое число n . Составить строку-описание оценки, соответствующей числу n (1 — «плохо», 2 — «двойка», 3 — «тройка», 4 — «хорошо», 5 — «отлично»). Если n не лежит в диапазоне 1–5, то вывести строку «ошибка»
 Алгоритмический языкБлок-схема

выбор

при $n=1$: $c := \text{«плохо»}$
 при $n=2$: $c := \text{«двойка»}$
 при $n=3$: $c := \text{«тройка»}$
 при $n=4$: $c := \text{«хорошо»}$
 при $n=5$: $c := \text{«отлично»}$
 иначе $c := \text{«ошибка»}$
всё



3. Структура **цикл**. Обеспечивает многократное выполнение некоторой совокупности действий, которая называется телом цикла. Циклы бывают трёх видов: с предусловием **«пока-делай»**, с постусловием **«делай-пока»**, со счётчиком **«для»**.

Цикл с предусловием (**«пока-делай»**). Предписывает выполнять тело цикла до тех пор, пока выполняется условие, записанное после слова **пока**.

Алгоритмический язык

Блок-схема

нц пока условие
 тело цикла
кц



Пример 1.

Дано целое число $N (> 0)$. Используя операции деления нацело, найти количество его цифр.

Алгоритмический язык

Блок-схема

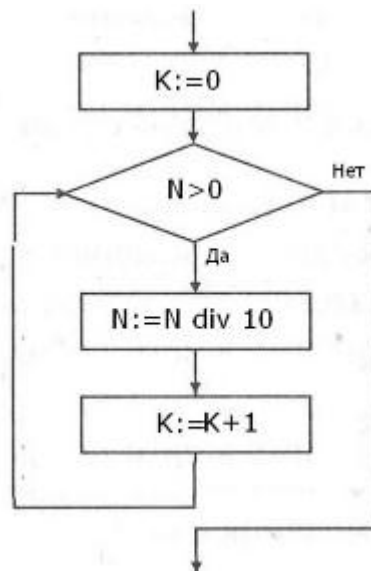
$K := 0;$

нц пока $N > 0$

$N := N \text{ div } 10$

$K := K + 1$

кц

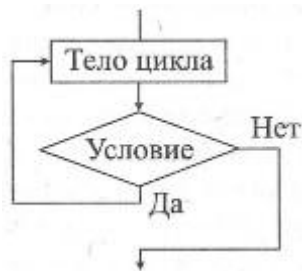


Цикл с постусловием («**делай-пока**»). Предписывает выполнять тело цикла до тех пор, пока не выполняется условие (на Паскале **until**), записанное после слова **пока**. В отличие от цикла, «**пока-делай**» тело цикла выполняется хотя бы один раз.

Алгоритмический язык

Блок-схема

нц
 тело цикла
пока условие
кц

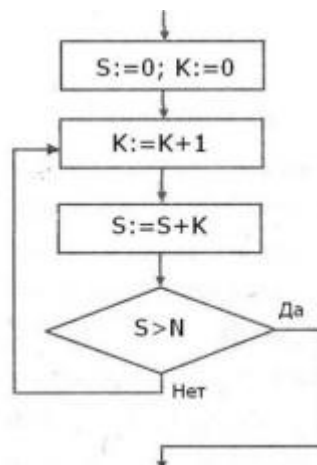


Пример 2.

Дано целое число $N (> 1)$. Определить наименьшее из целых чисел K , для которых сумма $S = 1 + 2 + \dots + K$ будет больше N .

Алгоритмический языкБлок-схема

$S:=0; K:=0$
нц
 $K:=K+1$
 $S:=S+K$
пока $S>N$
кц



Цикл со счетчиком («**для**»). Предписывает выполнять тело цикла для всех значений переменной (параметр цикла) в заданном диапазоне.

Алгоритмический языкБлок-схема

нц для i от k до m
 тело цикла
 кц



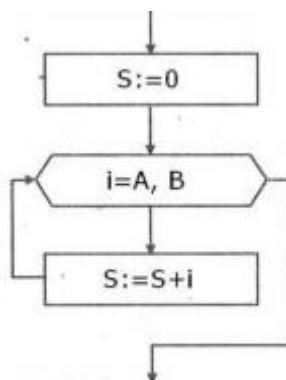
Пример 3. Даны два целых числа A и B ($A < B$). Найти сумму S всех целых чисел от A до B включительно.

Алгоритмический язык

Блок-схема

$S:=0$

нц для i от A до B
 $S:=S+i$
 кц



Пример 4. Составить блок-схему алгоритма вычисления произведения элементов заданной матрицы $A(10,10)$, расположенных на пересечении четных строк и четных столбцов.

Алгоритмический язык

Блок-схема

$i:=2; P:=1$

нц пока $i \leq 10$

$j:=2$

нц пока $j \leq 10$

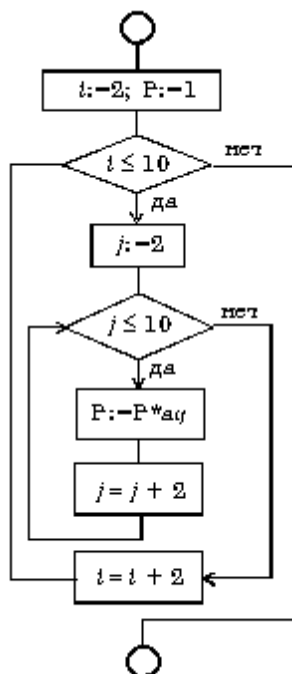
$P:=P \cdot a_{ij}$

$j:=j+2$

кц

$i:=i+2$

кц



ГЛАВА 6. Устройство компьютера.

В основу архитектуры современных персональных компьютеров положен магистрально-модульный принцип. Модульный принцип позволяет потребителю самому комплектовать нужную ему конфигурацию компьютера и производить при

необходимости ее модернизацию. Модульная организация компьютера опирается на магистральный (шинный) принцип обмена информацией между устройствами.

Магистраль (системная шина) включает в себя три многопроводные шины: *шину данных*, *шину адреса* и *шину управления*, которые представляют собой многопроводные линии. К магистральной подключаются процессор и оперативная память, а также периферийные устройства ввода, вывода и хранения информации, которые обмениваются информацией на машинном языке (последовательностями нулей и единиц в форме электрических импульсов).

Шина данных. По этой шине данные передаются между различными устройствами. Например, считанные из оперативной памяти данные могут быть переданы процессору для обработки, а затем полученные данные могут быть отправлены обратно в оперативную память для хранения. Таким образом, данные по шине данных могут передаваться от устройства к устройству в любом направлении.

Разрядность шины данных определяется разрядностью процессора, то есть количеством двоичных разрядов, которые могут обрабатываться или передаваться процессором одновременно. Разрядность процессоров постоянно увеличивается по мере развития компьютерной техники.

Шина адреса. Выбор устройства или ячейки памяти, куда пересылаются или откуда считываются данные по шине данных, производит процессор. Каждое устройство или ячейка оперативной памяти имеет свой адрес. Адрес передается по адресной шине, причем сигналы по ней передаются в одном направлении — от процессора к оперативной памяти и устройствам (однонаправленная шина).

Разрядность шины адреса определяет объем адресуемой памяти (адресное пространство), то есть количество однобайтовых ячеек оперативной памяти, которые могут иметь уникальные адреса. Количество адресуемых ячеек памяти можно рассчитать по формуле:

$N = 2^n$, где n — разрядность шины адреса. Разрядность шины адреса постоянно увеличивалась и в современных персональных компьютерах составляет 36 бит. Таким образом, максимально возможное количество адресуемых ячеек памяти равно:

$$N = 2^{36} = 68\,719\,476\,736.$$

Шина управления. По шине управления передаются сигналы, определяющие характер обмена информацией по магистральной. Сигналы управления показывают, какую операцию — считывание или запись информации из памяти — нужно производить, синхронизируют обмен информацией между устройствами и так далее.

Аппаратные средства являются базой информационных технологий, поэтому выбор компьютера и периферийного оборудования существенно влияют на эффективность информационных технологий. Различные виды профессиональной деятельности зачастую предъявляют совершенно различные требования к компьютерному оборудованию, и специалисту важно уметь оптимально подбирать компьютерную технику.

Для эффективной профессиональной деятельности важно хорошо ориентироваться в периферийном компьютерном оборудовании, уметь подобрать то, что лучше всего поможет вам организовать продуктивную работу.

ПЕЧАТАЮЩИЕ УСТРОЙСТВА

Несмотря на стремительное развитие всемирной компьютерной сети Интернет, электронную почту, прямой обмен данными и растущий электронный документооборот, значение бумажного вида документа по-прежнему велико, и в ближайшем будущем это положение едва ли изменится. Все-таки под документом принято понимать нечто осязаемое, а многие пользователи компьютера просто отказываются просматривать документ на экране монитора. Тем более, что многие ошибки лучше видны на бумаге.

Именно принтер превращает на наших глазах виртуально-мифический файл в документ с текстом, таблицами и графиками. Он позволяет выводить изображения на бумаге для дальнейшего использования.

Матричные принтеры

У большинства пользователей матричные принтеры вызывают ассоциации с чем-то очень дешевым либо морально устаревшим. В современных, даже небольших офисах, как правило, применяются лазерные принтеры или же, когда туговато с деньгами, — недорогие струйные модели. В домашнем применении также происходит постепенное вытеснение матричных принтеров струйными и лазерными.

Линейно-матричные принтеры

Принтеры построчной печати, или линейно-матричные принтеры, обеспечивают наибольшую производительность среди печатающих устройств ударного типа. Основные области применения этих принтеров — банковское дело и работа в вычислительных центрах. В отличие от обычных матричных в линейно-матричных принтерах иглы расположены не поперек, а вдоль распечатываемой строки по всей ее длине.

Струйные принтеры

Лазерные принтеры

Термосублимационные принтеры

У этих устройств есть одно уникальное свойство, которое в ряде случаев перевешивает все недостатки, присущие данному методу печати, — отменное качество отпечатков. Ни струйная, ни лазерная, ни тем более матричная печать не позволяют пока получать изображение фотореалистического качества, сегодня с этой задачей успешно справляются лишь термосублимационные принтеры.

С некоторыми оговорками можно считать, что термосублимационным принтерам нет альтернативы, когда речь идет о профессиональной работе с фотоизображениями, но для распечатки текста эти устройства далеко не самые подходящие.

Технология твердочернильной печати

Секрет технологии твердочернильной печати, реализованной фирмой Tektronix, заключается в уникальном составе твердых красителей ColorStix. Он обеспечивает практически мгновенное плавление при достижении определенной температуры и столь же быстрое затвердевание при попадании на более холодную бумагу. Кроме того, вязкостно-смачивающие характеристики расплава обеспечивают равномерное и строго ограниченное растекание красителя даже по далеко не идеальной бумажной поверхности.

Технология MicroDry

Эта новая технология позволяет выполнять печать, устойчивую к воздействию ультрафиолета и влаги и не выцветающую со временем.

Портативные принтеры

Для мобильных пользователей компьютеров существуют портативные и карманные компьютеры. Возможность печати документов в полевых условиях с таких компьютеров реализуют портативные принтеры. Их основное достоинство заключается в миниатюрности размеров, малой массе и возможности работать автономно (от батареек).

На начало XXI в. термодиффузный принтер PN-60 (CITIZEN) может считаться самым маленьким в мире. Он идеально подходит для работы «мобильного» офиса и может служить личным портативным принтером. При габаритных размерах 255х51 х46 см и массе 500 г без аккумулятора (700 г с аккумулятором) он позволяет печатать документы с разрешением 360 х 360 точек на дюйм термическим способом. В особых случаях возможна цветная печать документов.

СКАНЕРЫ

Из всех компьютерных периферийных устройств сканеры несомненно принадлежат к числу самых полезных. И хотя в скорости и удобстве использования сканеры уступают цифровым камерам, они более универсальны, существенно дешевле и к тому же обеспечивают гораздо более высокое качество изображений.

В паре с принтером сканер выполняет функции копира (вспомните, так ли уж редко вам приходится делать копии документов), а вместе с модемом способен заменить факс-аппарат. Планшетные устройства, кроме того, могут сканировать объемные предметы, например монеты или небольшие предметы.

Сканеры бывают нескольких типов, каждый из которых соответствует своей области применения: ручные, листовые, планшетные, барабанные и слайд-сканеры.

Наиболее распространены планшетные сканеры, обеспечивающие высокое разрешение. Они напоминают копировальные устройства: сканируемый материал укладывается на горизонтальную стеклянную поверхность, закрытую крышкой.

МНОГОФУНКЦИОНАЛЬНЫЕ ПЕРИФЕРИЙНЫЕ УСТРОЙСТВА

МФП-устройства сделаны по принципу «все в одном»: они объединяют в себе факс, сканер, копировальную машину и лазерный (струйный) принтер. Комбинированные устройства стремительно дешевеют при одновременном росте качества, так что сегодня такие решения представляются уже вполне разумными.

Самый популярный вариант — это сочетание принтер — копир — сканер. При выборе комбинированного устройства однозначно надо остановиться на варианте, укомплектованном лазерным принтером, так как эксплуатация такого «комбайна» на базе струйного принтера окажется весьма дорогой. Компромиссный вариант — стандартный лазерный принтер с «пристегиваемым» сканирующим модулем. Благодаря применению миниатюрного протяжного сканера такая модель будет достаточно дешевой и компактной. Она позволит копировать документы нажатием одной кнопки на передней панели, при этом работает достаточно быстро.

Главный недостаток этого гибрида — невозможность копирования книг и журналов, ведь каждая страничка должна быть протянута через сканирующий модуль.

МОДЕМ

Как известно, наиболее широко распространенным средством связи остаются пока аналоговые телефонные линии. Именно по этому основными коммуникационными устройствами являются аналоговые модемы, связывающие персональные компьютеры через обычные телефонные линии.

Модем является средством, обеспечивающим физическое подключение вашего компьютера к телефонной линии. Он подключает ваш компьютер к телефонной линии, превращая цифровую информацию в аналоговые сигналы, которые могут передаваться по телефонному каналу.

ПЛОТТЕРЫ

Плоттер — это устройство для вывода из ПК графической информации (чертежей, графиков, схем, диаграмм) на бумаге различного формата (до А0). Они подразделяются на перьевые и более современные — струйные.

ЦИФРОВЫЕ КАМЕРЫ

Цифровые камеры обеспечивают оперативность, недоступную традиционной фототехнике: снимки можно сразу же распечатать или отправить по сети. Камеры могут работать в различных режимах — от автоматических, доступных даже новичку, до полностью ручных, позволяющих реализовать свои творческие задумки. Цифровые возможности камер гармонично сочетаются с оптическими.

Современные цифровые камеры позволяют делать снимки с достаточно хорошим разрешением, а их цветовая палитра насчитывает миллионы оттенков. Вдобавок к этому фотокамеры обычно имеют полноцветный ЖК-дисплей, предназначенный для просмотра отснятых фотографий, накопитель на сменных носителях и встроенную вспышку.

Большинство цифровых фотокамер комплектуется несложными графическими редакторами, позволяющими корректировать яркость и контрастность снимков, избавляться от эффекта красных глаз. Кроме того, с цифровой фотокамерой вы будете избавлены от необходимости покупать фотопленку, а потом платить за ее проявку.

ИСТОЧНИКИ БЕСПЕРЕБОЙНОГО ПИТАНИЯ

Перепады напряжения в электрической сети являются причиной многих неполадок в работе компьютера. Скачок напряжения может вывести из строя модем, жесткий диск, принтер и любое другое внутреннее или внешнее устройство.

Для уменьшения вероятности возникновения неприятностей из-за перепадов напряжения в электросети, электропитание на компьютер и периферийные устройства лучше подавать через сетевой фильтр.

Средства многоступенчатой защиты сетевых фильтров призваны надежно охранять технику и данные от неблагоприятных факторов, будь то спонтанное отклонение сигнала, перенапряжение в сети, радиочастотные и электромагнитные помехи, разряды молний или статическое электричество.

ТЕХНИЧЕСКИЕ СРЕДСТВА ПРЕЗЕНТАЦИЙ

В качестве технических средств презентаций используются графопроекторы, слайд-проекторы и видеопроекторы.

Графопроекторы — это оптические устройства, проецирующие информацию с прозрачной пленки. Слайд-проекторы представляют собой устройства для проецирования стандартных слайдов размером 24 x 36 мм. Удобно использовать слайд-проекторы с дистанционным инфракрасным управлением.

Видеопроекторы — это сложные электронно-оптические устройства, проецирующие видеосигнал на экран. Источником данных может служить телевизор, видеокамера.

ГЛАВА 7. Программное обеспечение компьютера.

Современному компьютеру никак нельзя обойтись без программ. Ведь именно программы определяют возможности компьютера: что он будет делать — поможет свести бухгалтерский баланс или позволит побродить по всемирной сети Интернет. Большинство программ правильнее было бы называть программными продуктами, ведь зачастую на их создание требуется не меньше затрат, чем на производство самого компьютера. Практически любая программа, если это специально не оговорено, является коммерческим продуктом, который продается наравне с компьютерами.

Под программным обеспечением (ПО) информационных систем (понимается совокупность программных и документальных средств для создания и эксплуатации систем обработки данных средствами вычислительной техники. В самом общем плане программное обеспечение для вычислительной техники может быть разделено на базовое (системное) и прикладное.

Базовое (системное) ПО организует процесс обработки информации в компьютере и обеспечивает нормальную рабочую среду для прикладных программ. Базовое ПО настолько тесно связано с аппаратными средствами, что его иногда считают частью компьютера.

В состав базового ПО входят:

- операционные системы;
- сервисные программы (оболочки, утилиты, антивирусные средства);
- программы технического обслуживания (тестовые программы, программы контроля);
- инструментальное ПО (трансляторы языков программирования, компиляторы, интерпретаторы, ассемблеры)

Операционная система (ОС) — это комплекс специальных программных средств, предназначенных для управления загрузкой компьютера, запуском и выполнением других пользовательских программ, а также для планирования и управления вычислительными ресурсами персонального компьютера. Она обеспечивает управление процессом обработки информации и взаимодействие между аппаратными средствами и пользователем. Одной из важнейших функций ОС является автоматизация процессов

ввода-вывода информации, управления выполнением прикладных задач, решаемых пользователем. ОС загружает нужную программу в память ПК и следит за ходом ее выполнения; анализирует ситуации, препятствующие нормальным вычислениям, и дает указания о том, что необходимо сделать, если возникли трудности.

Сервисное программное обеспечение — это совокупность программных продуктов, предоставляющих пользователю дополнительные услуги в работе с компьютером и расширяющих возможности операционных систем.

Программные средства антивирусной защиты обеспечивают диагностику (обнаружение) и лечение (нейтрализацию) вирусов. Термином «вирус» обозначается программа, способная размножаться, внедряясь в другие программы, совершая при этом различные нежелательные действия.

Под программами технического обслуживания понимается совокупность программно-аппаратных средств для диагностики и обнаружения ошибок в процессе работы компьютера или вычислительной системы в целом.

Они включают в себя средства диагностики и тестового контроля правильности работы ПК и его отдельных частей, а также специальные программы диагностики и контроля вычислительной среды информационной системы в целом, в том числе программно-аппаратный контроль, осуществляющий автоматическую проверку работоспособности системы.

Инструментальное программное обеспечение

Система программирования — это комплекс средств, включающих в себя входной язык программирования, транслятор, машинный язык, библиотеки стандартных программ, средства отладки оттранслированных программ и компоновки их в единое целое.

Прикладное программное обеспечение предназначено для разработки и выполнения конкретных задач (приложений) пользователя.

Прикладное программное обеспечение работает под управлением базового ПО, в частности операционных систем. В состав прикладного ПО входят пакеты прикладных программ различного назначения и рабочие программы пользователя.

Пакет прикладных программ (ППП) — это комплекс программ, предназначенный для решения задач определенного класса.

Прикладное программное обеспечение общего назначения — это универсальные программные продукты, предназначенные для автоматизации разработки и эксплуатации функциональных задач пользователя и информационных систем в целом.

К этому классу ППП относятся:

- текстовые и графические редакторы;
- электронные таблицы;
- системы управления базами данных (СУБД);
- интегрированные.

Текстовые редакторы используются для обработки текстовой информации, выполняют, в основном, следующие функции: запись текста в файл; вставку, удаление, замену символов, строк и фрагментов текста; проверку орфографии; оформление текста различными шрифтами; выравнивание текста; подготовку оглавлений, разбиение текста на страницы; поиск и замену слов и выражений; включение в текст несложных иллюстраций; печать текста.

Наибольшее распространение получили текстовые редакторы Microsoft Word, Word Perfect, ChiWriter, MultiEdit, AmiPro, Lexicon.

Графические редакторы предназначены для обработки графических документов, включая диаграммы, иллюстрации, чертежи, таблицы.

Наиболее известны следующие графические-редакторы:

PaintBrush, CorelDRAW, Adobe Photoshop, Adobe Illustrator.

Издательские системы соединяют в себе возможности текстовых и графических редакторов, обладают развитыми возможностями по формированию полос с графическими материалами и последующим выводом на печать. Эти системы ориентированы на использование в издательском деле и называются системами верстки. Примером таких систем служит программа Adobe PageMaker.

Электронной таблицей называется программа для обработки числовых данных в таблицах. Данные в таблице хранятся в ячейках, находящихся на пересечении столбцов и строк. В ячейках могут храниться числа, символьные данные и формулы. Формулы задают зависимость значения одних ячеек от содержимого других ячеек. (Наиболее популярной электронной таблицей можно считать MS Excel).

Для работы с базами данных используется специальное ПО — системы управления базами данных (СУБД). База данных (БД) — это совокупность специальным образом организованных наборов данных, хранящихся на диске. Управление базой данных включает в себя ввод данных, их коррекцию и манипулирование данными, т.е. добавление, удаление, извлечение, обновление и другие операции.

ПО бухгалтерского учета (ПО БУ) интегрируется в комплексные системы автоматизации деятельности предприятия. Большинство таких пакетов работает под управлением операционной системы Windows и предназначено для эксплуатации в локальных сетях.

ПО справочно-правовых систем (ПО СПС). ПО СПС представляет собой эффективный инструмент работы с огромным объемом законодательной информации, поступающей непрерывным потоком.

В России насчитывается более десятка правовых систем. Наиболее известными и популярными можно считать справочно-правовые системы «Консультант Плюс», «Гарант», «Кодекс» и «Референт».

ГЛАВА 8. Компьютерные сети. Интернет.

Информационные ресурсы общества представляют собой ресурсы накопления в форме позволяющей их воспроизводства для общества.

Это ресурсы несущих в себе информацию: все возможные книги, архивы, документации, базы данных и т.д.

Отличительная черта большинства информационных ресурсов – их доступность для простых пользователей.

Особое место в информационных ресурсах стали занимать электронные книги, отличительной чертой которых является использования гиперссылок позволяющих переходить на разные фрагменты книг их напрямую.

Основными источниками информационных ресурсов для пользователей является сеть интернет.

Локальные компьютерные сети

При работе на персональном компьютере в автономном режиме пользователи могут обмениваться информацией (программами, документами и так далее), лишь копируя ее на дискеты. Однако перемещение дискеты между компьютерами не всегда возможно и может занимать достаточно продолжительное время.

Создание компьютерных сетей вызвано практической потребностью совместного использования информации пользователями, работающими на удаленных друг от друга компьютерах. Сети предоставляют пользователям возможность не только быстрого обмена информацией, но и совместного использования принтеров и других периферийных устройств и даже одновременной работы с документами.

Локальная сеть объединяет компьютеры, установленные в одном помещении (например, школьный компьютерный класс, состоящий из 8-12 компьютеров) или в одном

здании (например, в здании школы могут быть объединены в локальную сеть несколько десятков компьютеров, установленных в различных предметных кабинетах).

Локальная сеть объединяет несколько компьютеров и дает возможность пользователям совместно использовать ресурсы компьютеров, а также подключенных к сети периферийных устройств (принтеров, плоттеров, дисков, модемов и др.), — в небольших локальных сетях все компьютеры обычно равноправны, то есть пользователи самостоятельно решают, какие ресурсы своего компьютера (диски, каталоги, файлы) сделать общедоступными по сети. Такие сети называются *одноранговыми*.

Если к локальной сети подключено более **10** компьютеров, одноранговая сеть может оказаться недостаточно производительной. Для увеличения производительности, а также в целях обеспечения большей надежности при хранении информации в сети некоторые компьютеры специально выделяются для хранения файлов и программных приложений. Такие компьютеры называются *серверами*, а локальная сеть — *сетью на основе сервера*.

Аппаратное обеспечение сети. Каждый компьютер, подключенный к локальной сети, должен иметь специальную плату (сетевой адаптер).

Основной функцией сетевого адаптера является передача и прием информации из сети. В настоящее время наиболее часто используются сетевые адаптеры типа EtherNet, которые могут объединять в сеть компьютеры различных аппаратных и программных платформ (IBM-совместимые, Macintosh, Unix-компьютеры).

Соединение компьютеров (сетевых адаптеров) между собой производится с помощью кабелей различных типов (*коаксиального, витой пары, оптоволоконного*). Для подключения к локальной сети портативных компьютеров часто используется беспроводное подключение, при котором передача данных осуществляется с помощью электромагнитных

Важнейшей характеристикой локальных сетей, которая определяется типом используемых сетевых адаптеров и кабелей, является скорость передачи информации по сети. Скорость передачи информации по локальной сети обычно находится в диапазоне от **10** до **100** Мбит/с.

Топология сети. Общая схема соединения компьютеров в локальной сети называется *топологией сети*. Топологии сети могут быть различными.

Вариант соединения компьютеров между собой, когда кабель проходит от одного компьютера к другому, последовательно соединяя компьютеры и периферийные устройства между собой, называется линейным соединением.

Если к каждому компьютеру подходит отдельный кабель из одного центрального узла, то реализуется локальная сеть типа «звезда».

Обычно при такой схеме соединения центральным узлом является более мощный компьютер.

Преимущество локальной сети типа «звезда» перед локальной сетью типа «линейная шина» состоит в том, что при выходе из строя сетевого кабеля у одного компьютера локальная сеть в целом продолжает нормально функционировать.

Предоставление доступа к ресурсам компьютера. В операционной системе Windows пользователь любого компьютера, подключенного к сети, может предоставить доступ к своим дискам, папкам или файлам. Пользователи, работающие за другими компьютерами, могут после этого пользоваться предоставленными ресурсами.

Глобальные компьютерные сети.

Локальные сети обычно объединяют несколько десятков компьютеров, размещенных в одном здании, однако они не позволяют обеспечить совместный доступ к информации пользователям, находящимся, например, в различных частях города. На помощь приходят *региональные сети*, объединяющие компьютеры в пределах одного региона (города, страны, континента).

Многие организации, заинтересованные в защите информации от несанкционированного доступа (например, военные, банковские и пр.), создают собственные, так называемые *корпоративные сети*. Корпоративная сеть может объединять тысячи и десятки тысяч компьютеров, размещенных в различных странах и городах (в качестве примера можно привести сеть корпорации Microsoft — Microsoft Network (MSN)).

Потребности формирования единого мирового информационного пространства привели к созданию глобальной компьютерной сети Интернет. В настоящее время на более чем 150 миллионах компьютеров, подключенных к Интернету, хранится громадный объем информации (сотни миллионов файлов, документов и так далее). Глобальная сеть Интернет привлекает пользователей своими информационными ресурсами и сервисами (услугами), которыми пользуется около миллиарда человек во всех странах мира.

Интернет — это глобальная компьютерная сеть, объединяющая многие *локальные, региональные и корпоративные сети* и включающая сотни миллионов компьютеров.

В каждой такой локальной или корпоративной сети обычно имеется, по крайней мере, один компьютер, который имеет постоянное подключение к Интернету с помощью линии связи с высокой пропускной способностью (сервер Интернета). В качестве таких «магистральных» линий связи обычно используются оптоволоконные линии с пропускной способностью до 20 Гбит/с и более.

Надежность функционирования глобальной сети обеспечивает большое количество линий связи между региональными сегментами сети. Например, российский региональный сегмент Интернета имеет несколько магистральных линий связи, соединяющих его с североамериканским, европейским и японским сегментами.

Основу, «каркас» Интернета составляют более 150 миллионов серверов, постоянно подключенных к сети, из которых в России насчитывается около 400 тысяч (на начало 2002 г.).

К серверам Интернета могут подключаться с помощью локальных сетей или коммутируемых телефонных линий сотни миллионов пользователей Интернета.

ГЛАВА 9. Информационная безопасность.

Лицензионные, условно бесплатные и бесплатные программы

Программы по их юридическому статусу можно разделить на три большие группы: *лицензионные, условно бесплатные (shareware) и свободно распространяемые программы (freeware)*.

Дистрибутивы лицензионных программ (дискеты или диски CD-ROM, с которых производится установка программ на компьютеры пользователей) распространяются разработчиками на основании договоров с пользователями на платной основе, проще говоря, лицензионные программы продаются. Довольно часто разработчики предоставляют существенные скидки при покупке лицензий на использование программы на большом количестве компьютеров или на использование программы в учебных заведениях. В соответствии с лицензионным соглашением разработчики программы гарантируют ее нормальное функционирование в определенной операционной системе и несут за это ответственность.

Некоторые фирмы — разработчики программного обеспечения предлагают пользователям условно бесплатные программы в целях их рекламы и продвижения на рынок. Пользователю предоставляется версия программы с ограниченным сроком действия (после истечения указанного срока программа перестает работать, если за нее не произведена оплата) или версия программы с ограниченными функциональными возможностями (в случае оплаты пользователю сообщается код, включающий все функции).

Многие производители программного обеспечения и компьютерного оборудования

заинтересованы в широком бесплатном распространении программного обеспечения. К таким программным средствам можно отнести следующие:

- новые недоработанные (бета) версии программных продуктов (это позволяет провести их широкое тестирование);
- программные продукты, являющиеся частью принципиально новых технологий (это позволяет завоевать рынок);
- дополнения к ранее выпущенным программам, исправляющие найденные ошибки или расширяющие возможности;
- устаревшие версии программ;
- драйверы к новым устройствам или улучшенные драйверы к уже существующим.

Правовая охрана информации

Правовая охрана программ и баз данных. Правовая охрана программ для ЭВМ и баз данных впервые в полном объеме введена в Российской Федерации Законом РФ «О правовой охране программ для электронных вычислительных машин и баз данных», который вступил в силу в 1992 году.

Предоставляемая настоящим законом правовая охрана распространяется на все виды программ для ЭВМ (в том числе на операционные системы и программные комплексы), которые могут быть выражены на любом языке и в любой форме, включая исходный текст на языке программирования и машинный код. Однако правовая охрана не распространяется на идеи и принципы, лежащие в основе программы для ЭВМ, в том числе на идеи и принципы организации интерфейса и алгоритма.

Для признания и осуществления авторского права на программы для ЭВМ не требуется ее регистрация в какой-либо организации. Авторское право на программы для ЭВМ возникает автоматически при их создании.

Для оповещения о своих правах разработчик программы может, начиная с первого выпуска в свет программы, использовать знак охраны авторского права, состоящий из трех элементов:

- буквы С в окружности или круглых скобках ©;
- наименования (имени) правообладателя;
- года первого выпуска программы в свет. Например, знак охраны авторских прав

на текстовый редактор Word выглядит следующим образом:

© Корпорация Microsoft, 1993-1997.

Автору программы принадлежит исключительное право осуществлять воспроизведение и распространение программы любыми способами, а также модификацию программы.

Организация или пользователь, правомерно владеющий экземпляром программы (купивший лицензию на ее использование), вправе без получения дополнительного разрешения разработчика осуществлять любые действия, связанные с функционированием программы, в том числе ее запись и хранение в памяти ЭВМ. Запись и хранение в памяти ЭВМ допускаются в отношении одной ЭВМ или одного пользователя в сети, если другое не предусмотрено договором с разработчиком.

Необходимо знать и выполнять существующие законы, запрещающие нелегальное копирование и использование лицензионного программного обеспечения. В отношении организаций или пользователей, которые нарушают авторские права, разработчик может потребовать возмещения причиненных убытков и выплаты нарушителем компенсации в определяемой по усмотрению суда сумме от 5000-кратного до 50 000-кратного размера минимальной месячной оплаты труда.

Электронная подпись. В 2002 году был принят Закон РФ «Об электронно-цифровой подписи», который стал законодательной основой электронного документооборота в России. По этому закону электронная цифровая подпись в электронном документе признается юридически равнозначной подписи в документе на бумажном носителе.

При регистрации электронно-цифровой подписи в специализированных центрах корреспондент получает два ключа: секретный и открытый. Секретный ключ хранится на дискете или смарт-карте и должен быть известен только самому корреспонденту. Открытый ключ должен быть у всех потенциальных получателей документов и обычно рассылается по электронной почте.

Процесс электронного подписания документа состоит в обработке с помощью секретного ключа текста сообщения. Далее зашифрованное сообщение посылается по электронной почте абоненту. Для проверки подлинности сообщения и электронной подписи абонент использует открытый ключ.

Защита информации

Защита доступа к компьютеру. Для предотвращения несанкционированного доступа к данным, хранящимся на компьютере, используются пароли. Компьютер разрешает доступ к своим ресурсам только тем пользователям, которые зарегистрированы и ввели правильный пароль. Каждому конкретному пользователю может быть разрешен доступ только к определенным информационным ресурсам. При этом может производиться регистрация всех попыток несанкционированного доступа.

Защита пользовательских настроек имеется в операционной системе Windows (при загрузке системы пользователь должен ввести свой пароль), однако такая защита легко преодолима, так как пользователь может отказаться от введения пароля. Вход по паролю может быть установлен в программе **BIOS Setup**, компьютер не начнет загрузку операционной системы, если не введен правильный пароль. Преодолеть такую защиту нелегко, более того, возникнут серьезные проблемы доступа к данным, если пользователь забудет этот пароль.

В настоящее время для защиты от несанкционированного доступа к информации все более часто используются биометрические системы авторизации и идентификации пользователей. Используемые в этих системах характеристики являются неотъемлемыми качествами личности человека и поэтому не могут быть утерянными и подделанными. К биометрическим системам защиты информации относятся системы распознавания речи, системы идентификации по отпечаткам пальцев, а также системы идентификации по радужной оболочке глаза.

Защита программ от нелегального копирования и использования. Компьютерные пираты, нелегально тиражируя программное обеспечение, обесценивают труд программистов, делают разработку программ экономически невыгодным бизнесом. Кроме того, компьютерные пираты нередко предлагают пользователям недоработанные программы, программы с ошибками или их демоверсии.

Для того чтобы программное обеспечение компьютера могло функционировать, оно должно быть установлено (инсталлировано). Программное обеспечение распространяется фирмами-производителями в форме *дистрибутивов* на CD-ROM. Каждый дистрибутив имеет свой серийный номер, что препятствует незаконному копированию и установке программ.

Для предотвращения нелегального копирования программ и данных, хранящихся на CD-ROM, может использоваться специальная защита. На CD-ROM может быть размещен закодированный программный ключ, который теряется при копировании и без которого программа не может быть установлена.

Защита от нелегального использования программ может быть реализована с помощью аппаратного ключа, который присоединяется обычно к параллельному порту компьютера. Защищаемая программа обращается к параллельному порту и запрашивает секретный код; если аппаратный ключ к компьютеру не присоединен, то защищаемая программа определяет ситуацию нарушения защиты и прекращает свое выполнение.

Защита данных на дисках. Каждый диск, папка и файл локального компьютера, а также компьютера, подключенного к локальной сети, может быть защищен от несанкционированного доступа. Для них могут быть установлены определенные права

доступа (полный, только чтение, по паролю), причем права могут быть различными для различных пользователей.

Для обеспечения большей надежности хранения данных на жестких дисках используются RAID-массивы (Redundant Arrays of Independent Disks — избыточный массив независимых дисков). Несколько жестких дисков подключаются к специальному RAID-контроллеру, который рассматривает их как единый логический носитель информации. При записи информации она дублируется и сохраняется на нескольких дисках одновременно, поэтому при выходе из строя одного из дисков данные не теряются.

Защита информации в Интернете. Если компьютер подключен к Интернету, то в принципе любой пользователь, также подключенный к Интернету, может получить доступ к информационным ресурсам этого компьютера. Если сервер имеет соединение с Интернетом и одновременно служит сервером локальной сети (Интранет-сервером), то возможно несанкционированное проникновение из Интернета в локальную сеть.

Механизмы проникновения из Интернета на локальный компьютер и в локальную сеть могут быть разными:

- загружаемые в браузер Web-страницы могут содержать активные элементы ActiveX или Java-апплеты, способные выполнять деструктивные действия на локальном компьютере;
- некоторые Web-серверы размещают на локальном компьютере текстовые файлы cookie, используя которые можно получить конфиденциальную информацию о пользователе локального компьютера;
- с помощью специальных утилит можно получить доступ к дискам и файлам локального компьютера и др.

Для того чтобы этого не происходило, устанавливается программный или аппаратный барьер между Интернетом и Интранетом с помощью брандмауэра (firewall — межсетевой экран). Брандмауэр отслеживает передачу данных между сетями, осуществляет контроль текущих соединений, выявляет подозрительные действия и тем самым предотвращает несанкционированный доступ из Интернета в локальную сеть.

Первая массовая эпидемия компьютерного вируса произошла в 1986 году, когда вирус Brain «заражал» дискеты для первых массовых персональных компьютеров. В настоящее время известно несколько десятков тысяч вирусов, заражающих компьютеры с различными операционными системами и распространяющихся по компьютерным сетям.

Обязательным свойством компьютерного вируса является способность к размножению (самокопированию) и незаметному для пользователя внедрению в файлы, загрузочные секторы дисков и документы. Название «вирус» по отношению к компьютерным программам пришло из биологии именно по признаку способности к саморазмножению.

После заражения компьютера вирус может активизироваться и заставить компьютер выполнять какие-либо действия. Активизация вируса может быть связана с различными событиями (наступлением определенной даты или дня недели, запуском программы, открытием документа и так далее).

Компьютерные вирусы являются программами, которые могут «размножаться» и скрытно внедрять свои копии в файлы, загрузочные секторы дисков и документы. Активизация компьютерного вируса может вызывать уничтожение программ и данных. Разнообразны последствия действия вирусов; по величине вредных воздействий вирусы можно разделить на:

- неопасные, влияние которых ограничивается уменьшением свободной памяти на диске, графическими, звуковыми и другими внешними эффектами;
- опасные, которые могут привести к сбоям и зависаниям при работе компьютера;
- очень опасные, активизация которых может привести к потере программ и данных (изменению или удалению файлов и каталогов),

форматированию винчестера и так далее.

По «среде обитания» вирусы можно разделить на *файловые, загрузочные, макровирусы и сетевые*.

Файловые вирусы. Файловые вирусы различными способами внедряются в исполнимые файлы (программы) и обычно активизируются при их запуске. После запуска зараженной программы вирус находится в оперативной памяти компьютера и является активным (то есть может заражать другие файлы) вплоть до момента выключения компьютера или перезагрузки операционной системы.

При этом файловые вирусы не могут заразить файлы данных (например, файлы, содержащие изображение или звук).

Профилактическая защита от файловых вирусов состоит в том, что не рекомендуется запускать на выполнение файлы, полученные из сомнительного источника и предварительно не проверенные антивирусными программами.

Загрузочные вирусы. Загрузочные вирусы записывают себя в загрузочный сектор диска. При загрузке операционной системы с зараженного диска вирусы внедряются в оперативную память компьютера. В дальнейшем загрузочный вирус ведет себя так же, как файловый, то есть может заражать файлы при обращении к ним компьютера.

Профилактическая защита от таких вирусов состоит в отказе от загрузки операционной системы с гибких дисков и установке в BIOS вашего компьютера защиты загрузочного сектора от изменений.

Макровирусы. Макровирусы заражают файлы документов Word и электронных таблиц Excel. Макровирусы являются фактически макрокомандами (макросами), которые встраиваются в документ.

После загрузки зараженного документа в приложение макровирусы постоянно присутствуют в памяти компьютера и могут заражать другие документы. Угроза заражения прекращается только после закрытия приложения.

Профилактическая защита от макровирусов состоит в предотвращении запуска вируса. При открытии документа в приложениях Word и Excel сообщается о присутствии в них макросов (потенциальных вирусов) и предлагается запретить их загрузку. Выбор запрета на загрузку макросов надежно защитит ваш компьютер от заражения макровирусами, однако отключит и полезные макросы, содержащиеся в документе.

Сетевые вирусы. По компьютерной сети могут распространяться и заражать компьютеры любые обычные вирусы. Это может происходить, например, при получении зараженных файлов с серверов файловых архивов. Однако существуют и специфические сетевые вирусы, которые используют для своего распространения электронную почту и Всемирную паутину.

Интернет-черви (worm) — это вирусы, которые распространяются в компьютерной сети во вложенных в почтовое сообщение файлах. Автоматическая активизация червя и заражение компьютера могут произойти при обычном просмотре сообщения. Опасность таких вирусов состоит в том, что они по определенным датам активизируются и уничтожают файлы на дисках зараженного компьютера.

Кроме того, интернет-черви часто являются *троянами*, выполняя роль «троянского коня», внедренного в операционную систему. Такие вирусы «похищают» идентификатор и пароль пользователя для доступа в Интернет и передают их на определенный почтовый адрес. В результате злоумышленники получают возможность доступа в Интернет за деньги ничего не подозревающих пользователей.

Лавинообразная цепная реакция распространения вируса базируется на том, что вирус после заражения компьютера начинает рассылать себя по всем адресам электронной почты, которые имеются в адресной книге пользователя. Кроме того, может происходить заражение и по локальной сети, так как червь перебирает все локальные диски и сетевые диски с правом доступа и копируется туда под случайным именем.

Профилактическая защита от интернет-червей состоит в том, что не рекомендуется открывать вложенные в почтовые сообщения файлы, полученные из сомнительных источников.

Особой разновидностью вирусов являются активные элементы (программы) на языках JavaScript или VBScript, которые могут выполнять разрушительные действия, то есть являться вирусами (*скрипт-вирусами*). Такие программы передаются по Всемирной паутине в процессе загрузки Web-страниц с серверов Интернета в браузер локального компьютера.

Профилактическая защита от скрипт-вирусов состоит в том, что в браузере можно запретить получение активных элементов на локальный компьютер.

Антивирусные программы

Наиболее эффективны в борьбе с компьютерными вирусами антивирусные программы. Антивирусные программы могут использовать различные принципы для поиска и лечения зараженных файлов.

Полифаги. Самыми популярными и эффективными антивирусными программами являются антивирусные программы *полифаги* (например, Kaspersky Anti-Virus, Dr.Web). Принцип работы полифагов основан на проверке файлов, загрузочных секторов дисков и оперативной памяти и поиске в них известных и новых (неизвестных полифагу) вирусов.

Для поиска известных вирусов используются так называемые маски. Маской вируса является некоторая постоянная последовательность программного кода, специфичная для этого конкретного вируса. Если антивирусная программа обнаруживает такую последовательность в каком-либо файле, то файл считается зараженным вирусом и подлежит лечению.

Для поиска новых вирусов используются алгоритмы «эвристического сканирования», то есть анализ последовательности команд в проверяемом объекте. Если «подозрительная» последовательность команд обнаруживается, то полифаг выдает сообщение о возможном заражении объекта.

Полифаги могут обеспечивать проверку файлов в процессе их загрузки в оперативную память. Такие программы называются антивирусными *мониторами*.

К достоинствам полифагов относится их универсальность. К недостаткам можно отнести большие размеры используемых ими антивирусных баз данных, которые должны содержать информацию о максимально возможном количестве вирусов, что, в свою очередь, приводит к относительно небольшой скорости поиска вирусов.

Ревизоры. Принцип работы ревизоров (например, ADInf) основан на подсчете контрольных сумм для присутствующих на диске файлов. Эти контрольные суммы затем сохраняются в базе данных антивируса, как и некоторая другая информация: длины файлов, даты их последней модификации и пр. При последующем запуске ревизоры сверяют данные, содержащиеся в базе данных, с реально подсчитанными значениями. Если информация о файле, записанная в базе данных, не совпадает с реальными значениями, то ревизоры сигнализируют о том, что файл был изменен или заражен вирусом.

Недостаток ревизоров состоит в том, что они не могут обнаружить вирус в новых файлах (на дискетах, при распаковке файлов из архива, в электронной почте), поскольку в их базах данных отсутствует информация об этих файлах.

Блокировщики. Антивирусные *блокировщики* — это программы, перехватывающие «вирусоопасные» ситуации и сообщающие об этом пользователю. К таким ситуациям относится, например, запись в загрузочный сектор диска. Эта запись происходит при установке на компьютер новой операционной системы или при заражении загрузочным вирусом.

Наибольшее распространение получили антивирусные блокировщики в BIOS компьютера. С помощью программы BIOS Setup можно провести настройку BIOS таким

образом, что будет запрещена (заблокирована) любая запись в загрузочный сектор диска и компьютер будет защищен от заражения загрузочными вирусами.

К достоинствам блокировщиков относится их способность обнаруживать и останавливать вирус на самой ранней стадии его размножения.

Список литературы к конспекту лекций:

а) основная литература:

1. Алексеев А.П. Информатика 2015 [Электронный ресурс]: учебное пособие. - М.: СОЛОН-ПРЕСС. 2015 – режим доступа: [http:// www.studentlibrary.ru/ book/ ISBN 9785913591586.html](http://www.studentlibrary.ru/book/ISBN9785913591586.html)
2. Информатика [Электронный ресурс]: учеб. для вузов / Грошев А.С., Закляков П.В. – 2-е изд., перераб. И доп. – М.: ДМК Пресс, 2014 - режим доступа: [http:// www.studentlibrary.ru/ book/ ISBN 9785940747666.html](http://www.studentlibrary.ru/book/ISBN9785940747666.html)
3. Сборник задач по информатике. Углубленный уровень [Электронный ресурс] / Гай В.Е. – М.: БИНОМ, 2013– режим доступа: [http://www.studentlibrary.ru/ book/ ISBN 9785996311392.html](http://www.studentlibrary.ru/book/ISBN9785996311392.html)

б) дополнительная литература:

1. Кильдишов В.Д. Использование приложения MS Excel для моделирования различных задач [Электронный ресурс] - М.: СОЛОН-ПРЕСС, 2015. – режим доступа: [http://www.studentlibrary.ru/book/ ISBN9785913591456.html](http://www.studentlibrary.ru/book/ISBN9785913591456.html).
2. Начальный курс информатики. В 4 ч. Ч. 2 [Электронный ресурс]: Конспект лекций / А.М.Губарь. – М.: Издательство МГТУ им. Н.Э.Баумана, 2012. - режим доступа: http://www.studentlibrary.ru/book/bauman_0470.html
3. Омельченко В.П., Демидова А.А. Информатика. Практикум [Электронный ресурс]: / В.П. Омельченко, А.А. Демидова – М.: ГЭОТАР-Медиа, 2015 – режим доступа: [http://www.studentlibrary.ru/book/ ISBN9785970433812.html](http://www.studentlibrary.ru/book/ISBN9785970433812.html).

в) периодические издания

1. Вестник компьютерных и информационных технологий. ISSN 1810-7206.
2. Информатика и образование. ISSN 0234-0453.