

**МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РФ**  
**ТОМСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ СИСТЕМ**  
**УПРАВЛЕНИЯ И РАДИОЭЛЕКТРОНИКИ (ТУСУР)**



**Кафедра радиотехнических систем (РТС)**



**Б.Ф. Ноздреватых**

**ИНФОРМАТИКА**

**Учебное пособие**  
**для студентов направления**  
**11.03.02 «Инфокоммуникационные технологии и системы**  
**связи»**  
**профиля «Системы мобильной связи»**

**2018**

## АННОТАЦИЯ

Учебное пособие включает в себя теоретический материал, вопросы для самоконтроля и список вопросов для подготовки к экзамену по дисциплине **«Информатика»**.

Учебное пособие предназначено для подготовки студентов направления **11.03.02 «Инфокоммуникационные технологии и системы связи»** профиля **«Системы мобильной связи»**.

**СОДЕРЖАНИЕ**

<b>1. Цели и задачи дисциплины</b>	<b>5</b>
<b>2. Общие понятия</b>	<b>6</b>
<b>3. Представление информации</b>	<b>23</b>
<b>4. Основы математической логики</b>	<b>40</b>
<b>5. Программное обеспечение компьютера</b>	<b>67</b>
<b>6. Сетевое оборудование</b>	<b>85</b>
<b>Литература</b>	<b>133</b>
<b>Вопросы на самоподготовку</b>	<b>135</b>

## 1 Цель и задачи дисциплины

Информатика, компьютеры и программирование проникли во все сферы человеческой деятельности и стали элементом современной культуры. Особенно успешно процессорная техника внедряется в сферах науки и техники. В частности, с помощью компьютеров выполняется расчет, проектирование и моделирование узлов современных систем цифровой радиосвязи. Сами микропроцессорные устройства входят в состав современных радиотехнических систем, телекоммуникационных систем передачи и обработки информации. Все это делает актуальной подготовку современных специалистов в области численных методов решения прикладных задач.

Цели и задачи изучения дисциплины «Информатика» заключаются в обеспечении базовой подготовки студентов в области использования средств вычислительной техники и ознакомлении с основами проектирования и программирования.

Курс знакомит студентов с назначением и принципом действия современных персональных компьютеров, основами алгоритмизации и технологии программирования научно-технических задач, языками программирования высокого уровня, технологии обработки и отладки программ, современным программным обеспечением, методами решения типовых инженерных задач и их программной реализацией.

Изучение численных методов решения прикладных задач предполагает знакомство с одним из алгоритмических языков высокого уровня. Обычно в качестве базового языка программирования выбираются языки **C** или **Pascal**. В качестве базового языка был взят за основу язык программирования **C**. В качестве универсальной среды для научных и инженерных расчетов изучается программа **MathCAD**.

Лабораторные занятия предполагают закрепление на практике как основных конструкций языка программирования и элементов структурного программирования, так и базовых алгоритмов решения прикладных задач. В связи с этим на каждом занятии предполагается краткое напоминание конструкций языка программирования, формулировка задач, разбор вариантов

алгоритма решения, написание, отладка и тестирование программы. При разработке и написании конкретных программ особое внимание уделяется элементам структурного программирования, интерфейсу программ, операциям ввода-вывода данных с клавиатуры, чтению и записи данных с файла и в файл, а также графическому представлению результатов вычислений.

## **2. Общие понятия**

### **2.1 Введение. Общие сведения**

С появлением компьютеров сформировалась информатика – наука об общих свойствах и закономерностях информации, методах ее поиска, передачи, хранения, обработки и использования в различных сферах деятельности человека. Целью информатики является изучение структуры и общих свойств информации с выделением закономерностей процессов коммуникации. В современном понимании информатика – это область науки и техники, изучающая информационные процессы и методы их автоматизации.

Информационные технологии являются составной частью информатики. Средствами ИТ информация структурируется и формируется в виде знаний. В настоящее время идет превращение ее в ресурс, приобретающий материальный характер.

Технология при переводе с греческого (*techne*) означает искусство, мастерство, умение, а это не что иное, как процессы. Под процессом следует понимать определенную совокупность действий, направленных на достижение поставленной цели. Процесс должен определяться выбранной человеком стратегией и реализоваться с помощью совокупности различных средств и методов.

Технология – это комплекс научных и инженерных знаний, реализованных в приемах труда, наборах материальных, технических, энергетических, трудовых факторов производства, способах их соединения для создания продукта или услуги, отвечающих определенным требованиям.

Информационная технология (по определению, принятому ЮНЕСКО) – это комплекс взаимосвязанных, научных, технологических, инженерных дисциплин, изучающих методы эффективной организации труда людей, занятых обработкой и

хранением информации; вычислительную технику и методы организации и взаимодействия с людьми и производственным оборудованием, их практические приложения, а также связанные со всем этим социальные, экономические и культурные проблемы.

**Информационные технологии** (ИТ, от англ. information technology, IT) — широкий класс дисциплин и областей деятельности, относящихся к технологиям управления и обработки данных, в том числе, с применением вычислительной техники.

В последнее время под информационными технологиями чаще всего понимают компьютерные технологии. В частности, ИТ имеют дело с использованием компьютеров и программного обеспечения для хранения, преобразования, защиты, обработки, передачи и получения информации. Специалистов по компьютерной технике и программированию часто называют ИТ-специалистами.

В качестве критериев развитости информационного общества выделяют три: наличие компьютеров, уровень развития компьютерных сетей и количество населения, занятого в информационной сфере, а также использующего информационные и телекоммуникационные технологии в своей повседневной деятельности.

Информационная технология базируется на реализации базовых технологических процессов. К таким процессам можно отнести извлечение, транспортирование, обработку, хранение, представление и использование информации. Основу информационной индустрии составляют базовые информационные технологии, которые строятся на базовых технологических операциях, но кроме этого включают ряд специфических моделей и инструментальных средств.

Таким образом, информационные технологии (ИТ) рассматриваются как система, включающая базовые технологические процессы, базовые и специализированные информационные технологии, инструментальную базу.

Информационная технология – это процесс, включающий совокупность способов сбора, хранения, обработки и передачи информации на основе применения средств вычислительной техники.

Цель информационной технологии – производство информации для ее анализа человеком и принятия на его основе решения по выполнению какого – либо действия.

Информационная технология, как и любая другая, должна отвечать следующим требованиям:

- обеспечивать высокую степень разделения всего процесса обработки информации на этапы, операции, действия;
- включать весь набор элементов, необходимых для достижения поставленной цели;
- иметь регулярный характер. Этапы, действия, операции технологического процесса могут быть стандартизированы и унифицированы, что позволит более эффективно осуществлять целенаправленное управление информационными процессами.

## **2.2 Виды информации.**

Вся деятельность человека по преобразованию природы и общества сопровождается получением новой информации. Научная информация – логическая информация, адекватно отображающая объективные закономерности природы, общества и мышления. Ее делят по областям применения на следующие виды: политическая, техническая, биологическая, химическая, физическая и т. д.; по назначению – на массовую и специальную. Документальная информация – часть информации, которая занесена на бумажный носитель. Любое производство при функционировании требует перемещения документов, т.е. возникает документооборот. Техническая – информация, которая используется в сфере техники при решении производственных задач. Она сопровождает разработку новых изделий, материалов, конструкций, агрегатов, технологических процессов. Научно-техническая информация – объединение научной и технической видов информации.

Теоретической базой для информационных технологий является информатика. Целью информатики является изучение структуры и общих свойств информации с выявлением закономерностей процессов коммуникации. В современном понимании информатика – это область науки и техники, изучающая информационные процессы и методы их автоматизации. В информатике выделяют три уровня. Физический(нижний) уровень представляет собой средства вычислительной техники и техники

связи. Логический (средний) уровень составляют информационные технологии. Прикладной(верхний) уровень определяет идеологию применения информационных технологий для проектирования различных систем. Информационные технологии рассматриваются как система, включающая базовые технологические процессы, базовые и специализированные информационные технологии, инструментальную базу. Новая информационная технология – информационная технология на базе новых, компьютерных средств получения, хранения, актуализации информации (знаний). Для информационных технологий естественным является то, что они устаревают и заменяются новыми. В связи с этим при внедрении новой информационной технологии необходимо учитывать, что информационные продукты имеют чрезвычайно высокую скорость сменяемости новыми видами или версиями. Периоды сменяемости колеблются от нескольких месяцев до одного года. Поэтому для эффективного использования информационных технологий их необходимо регулярно модернизировать.

### **2.3 Классификация информационных технологий**

Информационные технологии принято классифицировать:

- по типу обрабатываемой информации (например, данные обрабатываются с помощью систем управления базами данных, а знания – с помощью экспертных систем);
- по типу пользовательского интерфейса (командный, WIMP-интерфейс, т. е. содержащий базы программ и меню действий, и SILK-интерфейс, использующий речевые команды и смысловые семантические 26 связи);
- по степени взаимодействия между собой (например, дискетное и сетевое взаимодействие).

### **2.4 Этапы эволюции информационных технологий – на практическое занятие**

Нулевое поколение, первое поколение, второе поколение, третье поколение, четвертое поколение, пятое поколение.

#### **Классификация информации**

1. По способу восприятия:

- визуальная,
- аудиальная,
- обонятельная,

- вкусовая,
  - тактильная.
2. По степени значимости:
- личная,
  - специальная,
  - общественная.

Личная – это знания, опыт, интуиция, умения, планы, прогнозы, эмоции, чувства, наследственная память конкретного человека. Специальная делится на научную, производственную, техническую, управленческую. Общественная включает в себя общественно-политическую, научно-популярную, быденную, эстетическую.

3. По форме представления:
- текстовая,
  - числовая,
  - графическая,
  - звуковая.
4. По способам (субъектам) обмена:
- социальная,
  - техническая,
  - биологическая,
  - генетическая.

Приведем другой вариант классификации информации:

- по сфере применения информации (экономическая, географическая, социологическая и пр.);
- по характеру источников информации (первичная, вторичная, обобщающая и пр.);
- по характеру носителя информации (информация, «зашифрованная» в молекулах ДНК или в длинах световых волн, информация на бумажном или магнитном носителе и пр.).

В зависимости от типа носителя различают следующие виды информации [1, с. 51]:

- документальную;
- акустическую (речевую);
- телекоммуникационную.

*Документальная* информация представляется в графическом или буквенно-цифровом виде на бумаге, а также в электронном виде на магнитных и других носителях.

*Речевая* информация возникает в ходе ведения разговоров, а также при работе систем звукоусиления и звуковоспроизведения. Носителем речевой информации являются звуковые колебания в диапазоне частот от 200...300 Гц до 4...6 кГц.

*Телекоммуникационная* информация циркулирует в технических средствах обработки и хранения информации, а также в каналах связи при ее передаче. Носителем информации при ее обработке техническими средствами и передаче по проводным каналам связи является электрический ток, а при передаче по радио- и оптическому каналам – электромагнитные волны.

Источник информации может вырабатывать непрерывное сообщение (сигнал), в этом случае информация называется *непрерывной*, или дискретной – информация называется *дискретной*.

Например, сигналы, передаваемые по радио и телевидению, а также используемые в магнитной записи, имеют форму непрерывных, быстро изменяющихся во времени зависимостей. Такие сигналы называются непрерывными, или *аналоговыми* сигналами. В противоположность этому в телеграфии и вычислительной технике сигналы имеют импульсную форму и называются *дискретными* сигналами.

Сравнивая непрерывную и дискретную формы представления информации, нетрудно заметить, что при использовании непрерывной формы для создания вычислительной машины потребуется меньшее число устройств (каждая величина представляется одним, а не несколькими сигналами), но эти устройства будут сложнее (они должны различать значительно большее число состояний сигнала).

Информация, циркулирующая в обществе, требует специальных средств и методов обработки, хранения и использования. Сформировались новые научные дисциплины – кибернетика, бионика, робототехника и другие, имеющие своей целью изучение закономерностей информационных процессов.

Существует три подхода к измерению информации:

**I подход.** – Неизмеряемость информации в быту (информация как новизна).

**II подход.** – Технический или объемный (информация как сообщения в форме знаков или сигналов, хранимые, перерабатываемые и обрабатываемые с помощью технических устройств).

В вычислительной технике применяются две стандартные единицы измерения информации: *бит* и *байт*. Поскольку компьютер предназначен для обработки больших объемов информации, то используют производные единицы – *килобайт* (Кб), *мегабайт* (Мб), *гигабайт* (Гб). Обычно приставка «кило» означает тысячу, а приставка «мега» - миллион. Но в вычислительной технике осуществляется привязка к принятой двоичной системе кодирования.

В силу этого один килобайт равен не 1000 байтов, а  $2^{10} = 1024$  байта.

Аналогично,  $1 \text{ Мб} = 2^{10} \text{ Кб} = 1024 \text{ Кб} = 2^{20} \text{ байтов} = 1\,048\,576$  байтов.

$1 \text{ Гб} = 2^{10} \text{ Мб} = 1024 \text{ Мб} = 2^{20} \text{ Кб} = 2^{30} \text{ байтов} = 1\,073\,741\,824$  байта.

**III подход.** – Вероятностный. Измерение информации в теории информации (информация как снятая неопределенность).

Получение информации (ее увеличение) означает увеличение знания, что, в свою очередь, означает уменьшение незнания или информационной неопределенности.

За единицу количества информации принимают выбор одного из двух равновероятных сообщений («да» или «нет», «1» или «0»). Она также названа *битом*.

## **2.5 Основные черты современных ИТ**

компьютерная обработка информации по заданным алгоритмам;

хранение больших объёмов информации на машинных носителях;

передача информации на любые расстояния в ограниченное время.

В широком понимании ИТ охватывает все области передачи, хранения и восприятия информации и не только компьютерные технологии. При этом ИТ часто ассоциируют именно с компьютерными технологиями, и это не случайно: появление компьютеров вывело ИТ на новый уровень. Как когда-то телевидение, а ещё ранее печатное дело. При этом основой ИТ являются технологии обработки, хранения и восприятия информации.

### Структура отрасли

Информатика

Программирование

Интернет и Всемирная паутина

Веб-разработка

Управление данными

Обработка данных

Добыча данных

Хранение данных

Базы данных

Информационная архитектура

Информационная безопасность

Криптография

Системная интеграция

Искусственный интеллект

Интеллектуальные информационные технологии

Беспроводные технологии

Программная инженерия

Информационная безопасность

Информационная система

Высокие технологии

Робототехника

Микроэлектроника

Преступления в сфере информационных технологий

## **2.6 Информационная система**

Термин информационная система (ИС) используется как в широком, так и в узком смысле.

В широком смысле информационная система есть совокупность технического, программного и организационного обеспечения, а также персонала, предназначенная для того, чтобы своевременно обеспечивать надлежащих людей надлежащей информацией[1].

Федеральный закон Российской Федерации от 27 июля 2006 г. N 149-ФЗ «Об информации, информационных технологиях и о защите информации» даёт следующее определение: «информационная система — совокупность содержащейся в базах данных информации и обеспечивающих ее обработку информационных технологий и технических средств».

По мнению одних авторов, ИС в широком смысле включает в себя персонал, её эксплуатирующий, по мнению других — нет.

В узком смысле информационной системой называют только подмножество компонентов ИС в широком смысле, включающее базы данных, СУБД и специализированные прикладные программы.

В любом случае основной задачей ИС является удовлетворение конкретных информационных потребностей в рамках конкретной предметной области. Современные ИС де-факто немислимы без использования баз данных и СУБД, поэтому термин «информационная система» на практике сливается по смыслу с термином «система баз данных».

### **Классификация информационных систем по архитектуре**

По степени распределённости отличают:

настольные (desktop), или локальные ИС, в которых все компоненты (БД, СУБД, клиентские приложения) работают на одном компьютере;

распределённые (distributed) ИС, в которых компоненты распределены по нескольким компьютерам.

### **Распределённые ИС, в свою очередь, разделяют на**

файл-серверные ИС (ИС с архитектурой «файл-сервер») - база данных находится на файловом сервере, а СУБД и клиентские приложения находятся на рабочих станциях

клиент-серверные ИС (ИС с архитектурой «клиент-сервер») - база данных и СУБД находятся на сервере, а на рабочих станциях находятся клиентские приложения

В свою очередь, клиент-серверные ИС разделяют на двухзвенные и многозвенные.

В двухзвенных (two-tier) ИС всего два типа «звеньев»: сервер баз данных, на котором находятся БД и СУБД, и рабочие станции, на которых находятся клиентские приложения. Клиентские приложения обращаются к СУБД напрямую.

В многозвенных (multi-tier) ИС добавляются промежуточные «звенья»: серверы приложений (application servers). Пользовательские клиентские приложения не обращаются к СУБД напрямую, они взаимодействуют с промежуточными звеньями.

### **Классификация информационных систем по сфере применения**

Поскольку ИС создаются для удовлетворения информационных потребностей в рамках конкретной предметной области, то каждой предметной области (сфере применения) соответствует свой тип ИС. Перечислять все эти типы не имеет смысла, так как количество предметных областей велико, но можно указать в качестве примера следующие типы ИС:

Экономическая информационная система.

Медицинская информационная система.

Географическая информационная.

### **Классификация информационных систем по охвату задач (масштабности)**

Персональная информационная система предназначена для решения некоторого круга задач одного человека.

Групповая информационная система ориентирована на коллективное использование информации членами рабочей группы или подразделения.

Корпоративная информационная система в идеале охватывает все информационные процессы целого предприятия, достигая полной согласованности, безызыточности и прозрачности

информационных процессов. Такие системы иногда называют системами комплексной автоматизации предприятия.

**Жизненный цикл информационной системы** — это процесс ее построения и развития.

Жизненный цикл информационной системы — период времени, который начинается с момента принятия решения о необходимости создания информационной системы и заканчивается в момент ее полного изъятия из эксплуатации[1].

## **2.7 Стадии и этапы создания автоматизированной системы:**

- Формирование требований к АС
- Обследование объекта и обоснование необходимости создания АС
- Формирование требований пользователя к АС
- Оформление отчета о выполнении работ и заявки на разработку АС
- Разработка концепции АС
- Изучение объекта
- Проведение необходимых научно-исследовательских работ
- Разработка вариантов концепции АС и выбор варианта концепции АС, удовлетворяющего требованиям пользователей
- Оформление отчета о проделанной работе
- Техническое задание
- Разработка и утверждение технического задания на создание АС
- Эскизный проект
- Разработка предварительных проектных решений по системе и ее частям
- Разработка документации на АС и ее части
- Технический проект
- Разработка проектных решений по системе и ее частям
- Разработка документации на АС и ее части
- Разработка и оформление документации на поставку комплектующих изделий

Разработка заданий на проектирование в смежных частях проекта

Рабочая документация

Разработка рабочей документации на АС и ее части

Разработка и адаптация программ

Ввод в действие

Подготовка объекта автоматизации

Подготовка персонала

Комплектация АС поставляемыми изделиями (программными и техническими средствами, программно-техническими комплексами, информационными изделиями)

Строительно-монтажные работы

Пусконаладочные работы

Проведение предварительных испытаний

Проведение опытной эксплуатации

Проведение приемочных испытаний

Сопровождение АС.

Выполнение работ в соответствии с гарантийными обязательствами

Послегарантийное обслуживание

### **Эскизный, технический проекты и рабочая документация**

— это последовательное построение все более точных проектных решений по всем видам обеспечения информационной системы. Допускается исключать стадию «Эскизный проект» и отдельные этапы работ на всех стадиях, объединять стадии «Технический проект» и «Рабочая документация» в «Технорабочий проект», параллельно выполнять различные этапы и работы, включать дополнительные.

## **2.8 База данных**

Базой данных является представленная в объективной форме совокупность самостоятельных материалов (статей, расчетов, нормативных актов, судебных решений и иных подобных материалов), систематизированных таким образом, чтобы эти материалы могли быть найдены и обработаны с помощью

электронной вычислительной машины (Гражданский кодекс РФ, ст. 1260).

Другие определения:

База данных — организованная в соответствии с определёнными правилами и поддерживаемая в памяти компьютера совокупность данных, характеризующая актуальное состояние некоторой предметной области и используемая для удовлетворения информационных потребностей пользователей[1].

База данных — совокупность данных, хранимых в соответствии со схемой данных, манипулирование которыми выполняют в соответствии с правилами средств моделирования данных[2].

База данных — некоторый набор перманентных (постоянно хранимых) данных, используемых прикладными программными системами какого-либо предприятия[3].

База данных — совокупность взаимосвязанных данных, совместно хранимых в одном или нескольких компьютерных файлах[4].

База данных — совместно используемый набор логически связанных данных (и описание этих данных), предназначенный для удовлетворения информационных потребностей организации[5].

**Наиболее часто используются следующие отличительные признаки:**

База данных хранится и обрабатывается в вычислительной системе. Таким образом, любые внекомпьютерные хранилища информации (архивы, библиотеки, картотеки и т. п.) базами данных не являются.

Данные в базе данных логически структурированы (систематизированы) с целью обеспечения возможности их эффективного поиска и обработки в вычислительной системе.

Структурированность подразумевает явное выделение составных частей (элементов), связей между ними, а также типизацию элементов и связей, при которой с типом элемента (связи) соотносится определённая семантика и допустимые операции[6].

База данных включает метаданные, описывающие логическую структуру БД в формальном виде (в соответствии с некоторой метамоделью).

Из перечисленных признаков только первый является строгим, а другие допускает различные трактовки и различные степени оценки. Не существует возможности строго формально определить, является ли некоторая совокупность данных на компьютере базой данных или нет. Можно лишь установить некоторую степень соответствия требованиям к БД.

**В такой ситуации не последнюю роль играет общепринятая практика. В соответствии с ней, например, не называют базами данных файловые архивы, Интернет-порталы или электронные таблицы, несмотря на то, что они в некоторой степени обладают признаками БД. Принято считать, что эта степень в большинстве случаев недостаточна (хотя могут быть исключения).**

Многие специалисты указывают на распространённую ошибку, состоящую в некорректном использовании термина база данных вместо термина система управления базами данных. Эти понятия, следовательно, необходимо различать. [7]

### **Классификации БД**

#### Классификация БД по модели данных.

Примеры:

Иерархические

Сетевые

Реляционные

Объектные

Объектно-ориентированные

Объектно-реляционные

#### Классификация БД по технологии хранения:

БД во вторичной памяти (традиционные)

БД в оперативной памяти (in-memory databases)

БД в третичной памяти (tertiary databases)

#### Классификация БД по содержанию:

Примеры:

Географические

Исторические

Научные

Мультимедийные.

Классификация БД по степени распределённости:

Централизованные (сосредоточенные)

Распределённые

Отдельное место в теории и практике занимают пространственные (англ. spatial), временные, или темпоральные (temporal) и пространственно-временные (spatial-temporal) БД.

Очень большие базы данных

Очень большая база данных (Very Large Database, VLDB) — это база данных, которая занимает чрезвычайно большой объём на устройстве физического хранения. Термин подразумевает максимально возможные объёмы БД, которые определяются последними достижениями в технологиях физического хранения данных и в технологиях программного оперирования данными.

Конкретное определение понятия «чрезвычайно большой объём» меняется во времени; в настоящее время считается, что это объём, измеряемый по меньшей мере терабайтами, а в последнее время — петабайтами.

Сверхбольшие базы и склады данных требуют особых подходов к логическому и системно-техническому проектированию, обычно выполняемому в рамках самостоятельного проекта, суть которого в том, чтобы найти такое системотехническое решение, которое попросту позволило бы хоть как-то работать с такими большими объемами. Такое решение возможно при наличии трех условий: специального решения для дисковой подсистемы, специальных версий операционной среды и специальных механизмов обращения СУБД к данным [8].

Система управления базами данных

Система управления базами данных (СУБД) — специализированная программа (чаще комплекс программ), предназначенная для организации и ведения базы данных. Для создания и управления информационной системой СУБД

необходима в той же степени, как для разработки программы на алгоритмическом языке необходим транслятор[источник не указан 280 дней].

### **Основные функции СУБД**

управление данными во внешней памяти (на дисках);  
 управление данными в оперативной памяти с использованием дискового кэша;  
 журнализация изменений, резервное копирование и восстановление базы данных после сбоев;  
 поддержка языков БД (язык определения данных, язык манипулирования данными).

**Обычно современная СУБД содержит следующие компоненты:**

ядро, которое отвечает за управление данными во внешней и оперативной памяти, и журнализацию,

процессор языка базы данных, обеспечивающий оптимизацию запросов на извлечение и изменение данных и создание, как правило, машинно-независимого исполняемого внутреннего кода,

подсистему поддержки времени исполнения, которая интерпретирует программы манипуляции данными, создающие пользовательский интерфейс с СУБД

а также сервисные программы (внешние утилиты), обеспечивающие ряд дополнительных возможностей по обслуживанию информационной системы.

### **Классификация СУБД**

#### По модели данных

Примеры:

Иерархические

Сетевые

Реляционные

Объектно-ориентированные

#### По степени распределённости

локальные СУБД (все части локальной СУБД размещаются на одном компьютере)

распределённые СУБД (части СУБД могут размещаться на двух и более компьютерах).

#### По способу доступа к БД

##### Файл-серверные

В файл-серверных СУБД файлы данных располагаются централизованно на файл-сервере. СУБД располагается на каждом клиентском компьютере (рабочей станции). Доступ СУБД к данным осуществляется через локальную сеть. Синхронизация чтений и обновлений осуществляется посредством файловых блокировок. Преимуществом этой архитектуры является низкая нагрузка на ЦП сервера. Недостатки: потенциально высокая загрузка локальной сети; затруднённая централизованная управляемость; затруднённое обеспечение таких важных характеристик как высокая надёжность, высокая доступность и высокая безопасность.

На данный момент файл-серверные СУБД считаются устаревшими.

Примеры: Microsoft Access, Paradox, dBase, FoxPro, Visual FoxPro.

##### Клиент-серверные

Клиент-серверная СУБД располагается на сервере вместе с БД и осуществляет доступ к БД непосредственно, в монопольном режиме. Все клиентские запросы на обработку данных обрабатываются клиент-серверной СУБД централизованно. Недостаток клиент-серверных СУБД состоит в повышенных требованиях к серверу. Достоинства: потенциально более низкая загрузка локальной сети; удобство централизованного управления; удобство обеспечения таких важных характеристик как высокая надёжность, высокая доступность и высокая безопасность.

Примеры: Oracle, Firebird, Interbase, IBM DB2, MS SQL Server, Sybase, PostgreSQL, MySQL, ЛИНТЕР, MDBS.

##### Встраиваемые

Встраиваемая СУБД — библиотека, которая позволяет унифицированным образом хранить большие объёмы данных на локальной машине. Доступ к данным может происходить через SQL либо через особые функции СУБД. Встраиваемые СУБД быстрее

обычных клиент-серверных и не требуют установки сервера, поэтому востребованы в локальном ПО, которое имеет дело с большими объемами данных (например, геоинформационные системы).

Примеры: OpenEdge, SQLite, BerkeleyDB, один из вариантов Firebird, MySQL, Sav Zigzag, Microsoft SQL Server Compact, ЛИНТЕР.

Вопросы для самоконтроля:

1. В чем измеряется информация?
2. Этапы развития информационных технологий
3. Виды баз данных
4. Классификация СУБД

### **3. Представление информации**

#### **3.1 Язык как способ представления информации.**

##### **Кодирование информации**

Язык - множество символов и совокупность правил, определяющих способы составления из этих символов осмысленных сообщений. Семантика - система правил и соглашений, определяющая толкование и придание смысла конструкциям языка.

*Кодирование* информации - это процесс формирования определенного представления информации. При кодировании информация представляется в виде дискретных данных. Декодирование является обратным к кодированию процессом.

В более узком смысле под термином "кодирование" часто понимают переход от одной формы представления информации к другой, более удобной для хранения, передачи или обработки. Компьютер может обрабатывать только информацию, представленную в числовой форме. Вся другая информация (например, звуки, изображения, показания приборов и т. д.) для обработки на компьютере должна быть преобразована в числовую форму. Например, чтобы перевести в числовую форму музыкальный звук, можно через небольшие промежутки времени измерять интенсивность звука на определенных частотах, представляя результаты каждого измерения в числовой форме. С помощью программ для компьютера можно выполнить преобразования

полученной информации.

Аналогичным образом на компьютере можно обрабатывать текстовую информацию. При вводе в компьютер каждая буква кодируется определенным числом, а при выводе на внешние устройства (экран или печать) для восприятия человеком по этим числам строятся изображения букв. Соответствие между набором букв и числами называется *кодировкой символов*.

Знаки или символы любой природы, из которых конструируются информационные сообщения, называют *кодами*. Полный набор кодов составляет *алфавит* кодирования. Простейшим алфавитом, достаточным для записи информации о чем-либо, является алфавит из двух символов, описывающих два его альтернативных состояния ("да" - "нет", "+" - "-", 0 или 1).

Как правило, все числа в компьютере представляются с помощью нулей и единиц (а не десяти цифр, как это привычно для людей). Иными словами, компьютеры обычно работают в *двоичной* системе счисления, поскольку при этом устройства для их обработки получают значительно более простыми. Ввод чисел в компьютер и вывод их для чтения человеком может осуществляться в привычной десятичной форме, а все необходимые преобразования выполняют программы, работающие на компьютере.

Любое информационное сообщение можно представить, не меняя его содержания, символами того или иного алфавита или, говоря иначе, получить ту или иную *форму представления*. Например, музыкальная композиция может быть сыграна на инструменте (закодирована и передана с помощью звуков), записана с помощью нот на бумаге (кодами являются ноты) или намагничена на диске (коды - электромагнитные сигналы).

Способ кодирования зависит от цели, ради которой оно осуществляется. Это может быть сокращение записи, засекречивание (шифровка) информации, или, напротив, достижение взаимопонимания. Например, система дорожных знаков, флажковая азбука на флоте, специальные научные языки и символы - химические, математические, медицинские и др., предназначены для того, чтобы люди могли общаться и понимать друг друга. От того, как представлена информация, зависит способ ее обработки, хранения, передачи и т.д.

Компьютер с точки зрения пользователя работает с информацией самой различной формы представления: числовой, графической, звуковой, текстовой и пр. Но мы уже знаем (упоминалось выше), что он оперирует только цифровой (дискретной) информацией. Значит, должны существовать способы перевода информации из внешнего вида, удобного пользователю, во внутреннее представление, удобное компьютеру, и обратно.

### 3.2. Позиционные и непозиционные системы счисления

**Система счисления** – совокупность приемов и правил наименования и обозначения чисел, позволяющих установить взаимно однозначное соответствие между любым числом и его представлением в виде конечного числа символов.

Разнообразные системы счисления, которые существовали раньше и которые используются в наше время, можно разделить на непозиционные и позиционные системы счисления. Знаки, используемые при записи чисел, называются цифрами.

В *непозиционных* системах счисления от положения цифры в записи числа не зависит величина, которую она обозначает. Примером непозиционной системы счисления является римская система, в которой в качестве цифр используются латинские буквы:

I	V	X	L	C	D	M
1	5	10	50	100	500	1000

В числе цифры записываются слева направо в порядке убывания. Величина числа определяется как сумма или разность цифр в числе. Если меньшая цифра стоит слева от большей цифры, то она вычитается, если справа - прибавляется. Например, VI = 5 + 1 = 6, а IX = 10 - 1 = 9, CCCXXVII=100+100+100+10+10+5+1+1=327.

В *позиционных* системах счисления величина, обозначаемая цифрой в записи числа, зависит от ее позиции. Количество используемых цифр называется *основанием* системы счисления. **Основание позиционной системы счисления** – количество

различных цифр, используемых для изображения чисел в данной системе счисления.

Место каждой цифры в числе называется *позицией*.

Система счисления	Основание	Алфавит
Десятичная	10	0123456789
Двоичная	2	01
Троичная	3	012
Восьмеричная	8	01234567
Шестнадцатеричная	16	0123456789ABCDEF

Первая известная нам система, основанная на позиционном принципе - шестидесятеричная вавилонская. Цифры в ней были двух видов, одним из которых обозначались единицы, другим - десятки. Следы вавилонской системы сохранились до наших дней в способах измерения и записи величин углов и промежутков времени.

Однако наибольшую ценность для нас имеет индо-арабская десятичная система. Индийцы первыми использовали ноль для указания позиционной значимости величины в строке цифр. Эта система получила название *десятичной* системы счисления, так как в ней десять цифр.

Для того чтобы лучше понять различие позиционной и непозиционной систем счисления, рассмотрим пример сравнения двух чисел. В позиционной системе счисления сравнение двух чисел происходит следующим образом: в рассматриваемых числах слева направо сравниваются цифры, стоящие в одинаковых позициях. Большая цифра соответствует большему значению числа. Например, для чисел 123 и 234, 1 меньше 2, поэтому число 234 больше, чем число 123. В непозиционной системе счисления это правило не действует. Примером этого может служить сравнение двух чисел IX и VI. Несмотря на то, что I меньше, чем V, число IX больше, чем число VI.

Далее мы будем рассматривать только позиционные системы

счисления.

Основание системы счисления, в которой записано число, обычно обозначается нижним индексом. Например,  $555_7$  - число, записанное в семеричной системе счисления. Если число записано в десятичной системе, то основание, как правило, не указывается. Основание системы - это тоже число, и его мы будем указывать в обычной десятичной системе. Вообще, число  $x$  может быть представлено в системе с основанием  $p$ , как  $x = a_n * p^n + a_{n-1} * p^{n-1} + a_1 * p^1 + a_0 * p^0$ , где  $a_n \dots a_0$  - цифры в представлении данного числа.

Так, например,  $1035_{10} = 1 * 10^3 + 0 * 10^2 + 3 * 10^1 + 5 * 10^0$ ;  
 $1010_2 = 1 * 2^3 + 0 * 2^2 + 1 * 2^1 + 0 * 2^0 = 10$ .

Наибольший интерес при работе на ЭВМ представляют системы счисления с основаниями 2, 8 и 16. Вообще говоря, этих систем счисления обычно хватает для полноценной работы, как человека, так и вычислительной машины. Однако иногда в силу различных обстоятельств приходится обращаться к другим системам счисления, например, к троичной, семеричной или системе счисления по основанию 32.

Для того чтобы нормально оперировать с числами, записанными в таких нетрадиционных системах, важно понимать, что принципиально они ничем не отличаются от привычной нам десятичной системы счисления. Сложение, вычитание, умножение в них осуществляется по одной и той же схеме.

Почему же мы не пользуемся другими системами счисления? В основном потому, что в повседневной жизни мы привыкли пользоваться десятичной системой счисления, и нам не требуется никакая другая система счисления. В вычислительных же машинах используется двоичная система счисления, так как оперировать над числами, записанными в двоичном виде, довольно просто.

Часто в информатике используют шестнадцатеричную систему, так как запись чисел в ней значительно короче записи чисел в двоичной системе. Может возникнуть вопрос: почему бы не использовать для записи очень больших чисел систему счисления, например по основанию 50? Для такой системы счисления необходимы 10 обычных цифр плюс 40 знаков, которые соответствовали бы числам от 10 до 49 и вряд ли кому-нибудь понравится работать с этими сорока знаками. Поэтому в реальной

жизни системы счисления по основанию 16, большему 16, практически не используются.

Методику представления информации в двоичной форме можно пояснить, проведя следующую игру. Нужно у собеседника получить интересующую нас информацию, задавая любые вопросы, но получая в ответ только одно из двух ДА либо НЕТ. Известным способом получения во время этого диалога двоичной формы информации является перечисление всех возможных событий. Рассмотрим простейший случай получения информации. Вы задаете только один вопрос: "Идет ли дождь?". При этом условимся, что с одинаковой вероятностью ожидаете ответ: "ДА" или "НЕТ". Легко увидеть, что любой из этих ответов несет самую малую порцию информации. Эта порция определяет единицу измерения информации, называемую битом. Благодаря введению понятия единицы информации появилась возможность определения размера любой информации числом битов. Образно говоря, если, например, объем грунта определяют в кубометрах, то объем информации - в битах. Условимся каждый положительный ответ представлять цифрой 1, а отрицательный - цифрой 0. Тогда запись всех ответов образует многозначную последовательность цифр, состоящую из нулей и единиц, например 0100.

Люди предпочитают десятичную систему, вероятно, потому, что с древних времен считали по пальцам. Но, не всегда и не везде люди пользовались десятичной системой счисления. В Китае, например, долгое время применялась пятеричная система счисления. В ЭВМ используют двоичную систему потому, что она имеет ряд преимуществ перед другими:

- для ее реализации используются технические элементы с двумя возможными состояниями (есть ток - нет тока, намагничен - ненамагничен);
- представление информации посредством только двух состояний надежно и помехоустойчиво;
- возможно применение аппарата булевой алгебры для выполнения логических преобразований информации;
- двоичная арифметика проще десятичной (двоичные таблицы сложения и умножения предельно просты).

В двоичной системе счисления всего две цифры, называемые двоичными (binary digits). Сокращение этого наименования привело к появлению термина бит, ставшего названием разряда двоичного числа. Веса разрядов в двоичной системе изменяются по степеням двойки. Поскольку вес каждого разряда умножается либо на 0, либо на 1, то в результате значение числа определяется как сумма соответствующих значений степеней двойки. Если какой-либо разряд двоичного числа равен 1, то он называется значащим разрядом. Запись числа в двоичном виде намного длиннее записи в десятичной системе счисления.

Арифметические действия, выполняемые в двоичной системе, подчиняются тем же правилам, что и в десятичной системе. Только в двоичной системе счисления перенос единиц в старший разряд возникает чаще, чем в десятичной. Вот как выглядит таблица сложения в двоичной системе:

$0 + 0$ $= 0$	$0 + 1 = 1$
$1 + 0$ $= 1$	$1 + 1 = 10$ (перенос в старший разряд)

Таблица умножения для двоичных чисел еще проще:

$0 * 0 = 0$	$1 * 0 = 0$	$0 * 1 = 0$	$1 * 1 = 1$
-------------	-------------	-------------	-------------

Рассмотрим подробнее, как происходит процесс умножения двоичных чисел. Пусть надо умножить число 1101 на 101 (оба числа в двоичной системе счисления). Машина делает это следующим образом: она берет число 1101 и, если первый элемент второго множителя равен 1, то она заносит его в сумму. Затем сдвигает число 1101 влево на одну позицию, получая тем самым 11010, и если, второй элемент второго множителя равен единице, то тоже заносит его в сумму. Если элемент второго множителя равен нулю, то сумма не изменяется.

Двоичное деление основано на методе, знакомом вам по

десятичному делению, т. е. сводится к выполнению операций умножения и вычитания. Выполнение основной процедуры - выбор числа, кратного делителю и предназначенного для уменьшения делимого, здесь проще, так как таким числом могут быть только либо 0, либо сам делитель.

Следует отметить, что большинство калькуляторов, реализованных на компьютере, позволяют осуществлять работу в системах счисления с основаниями 2, 8, 16 и, конечно, 10.

При наладке аппаратных средств компьютера или создании новой программы возникает необходимость "заглянуть внутрь" памяти машины, чтобы оценить ее текущее состояние. Но там все заполнено длинными последовательностями нулей и единиц двоичных чисел. Эти последовательности очень неудобны для восприятия человеком, привыкшим к более короткой записи десятичных чисел. Кроме того, естественные возможности человеческого мышления не позволяют оценить быстро и точно величину числа, представленного, например, комбинацией из 16 нулей и единиц.

Для облегчения восприятия двоичного числа решили разбивать его на группы разрядов, например, по три или четыре разряда. Эта идея оказалась очень удачной, так как последовательность из трех бит имеет 8 комбинаций, а последовательность из 4 бит - 16. Числа 8 и 16 являются степенями двойки, поэтому легко находить соответствие с двоичными числами. Развивая эту идею, пришли к выводу, что группы разрядов можно закодировать, сократив при этом длину последовательности знаков. Для кодировки трех битов требуется восемь цифр, поэтому взяли цифры от 0 до 7 десятичной системы. Для кодировки же четырех битов необходимо шестнадцать знаков; для этого взяли 10 цифр десятичной системы и 6 букв латинского алфавита: A, B, C, D, E, F. Полученные системы, имеющие основания 8 и 16, назвали соответственно восьмеричной и шестнадцатеричной.

В восьмеричной (octal) системе счисления используются восемь различных цифр 0, 1, 2, 3, 4, 5, 6, 7. Основание системы - 8. При записи отрицательных чисел перед последовательностью цифр ставят знак минус. Сложение, вычитание, умножение и деление чисел, представленных в восьмеричной системе, выполняются весьма просто подобно тому, как это делают в общеизвестной

десятичной системе счисления. В различных языках программирования запись восьмеричных чисел начинается с 0, например, запись 011 означает число 9.

В шестнадцатеричной (hexadecimal) системе счисления применяется десять различных цифр и шесть первых букв латинского алфавита. При записи отрицательных чисел слева от последовательности цифр ставят знак минус. Для того чтобы при написании компьютерных программ отличить числа, записанные в шестнадцатеричной системе, от других, перед числом ставят 0x. То есть 0x11 и 11 - это разные числа. В других случаях можно указать основание системы счисления нижним индексом.

Шестнадцатеричная система счисления широко используется при задании различных оттенков цвета при кодировании графической информации (модель RGB). Так, в редакторе гипертекста Netscape Composer можно задавать цвета для фона или текста как в десятичной, так и шестнадцатеричной системах счисления.

### 3.3. Перевод чисел из одной системы счисления в другую

Представление чисел в развернутой форме одновременно является способом перевода чисел в десятичную систему из любой другой позиционной системы счисления. Достаточно подсчитать результат по правилам десятичной арифметики.

Например, надо получить десятичные эквиваленты чисел:  $101,01_2$ ;  $673,2_8$ ;  $15A_{16}$ .

$$101,01_2 = 1 \cdot 2^2 + 0 \cdot 2^1 + 0 \cdot 2^0 + 1 \cdot 2^{-1} + 1 \cdot 2^{-2} = 4 + 1 + 1/4 = 5 + 0,25 = 5,25_{10};$$

Перевод десятичного числа в другую систему счисления может выполняться разными способами. При этом надо учитывать, что алгоритмы перевода целых чисел и правильных дробей будут отличаться. Для смешанного числа целая и дробная части переводятся отдельно по соответствующим алгоритмам. В итоговой записи искомого они объединяются и разделяются запятой.

Так называемый метод поэтапного деления заключается в последовательном целочисленном делении исходного числа и

получаемых неполных частных на основании той системы счисления, в которую осуществляется перевод. Остатки от деления составляют искомое число.

*Алгоритм перевода целого десятичного числа  $N_{10}$  в позиционную систему с основанием  $p$ :*

1. Разделить нацело число  $N$  на  $p$ .
2. Полученный остаток от деления дает цифру, стоящую в нулевом разряде  $p$ -ичной записи числа  $N$ .
3. Полученное частное снова разделить нацело на  $p$  и снова запомнить полученный остаток - это цифра первого разряда, и т.д.
4. Такое последовательное деление продолжается до тех пор, пока частное не станет равным 0.
5. Цифрами искомого числа являются остатки от деления, выписанные слева направо начиная с последнего полученного остатка.

Для оформления записи перевода предлагается один из возможных способов: слева от черты записываются неполные частные от целочисленного деления на основание, а справа - остатки от деления.

Например, надо перевести десятичное число 26 в двоичную, троичную и шестнадцатеричную системы счисления.

$26_{10} \rightarrow X_2$		$26_{10} \rightarrow X_3$		$26_{10} \rightarrow X_{16}$	
$q = 10, p = 2$		$q = 10, p = 3$		$q = 10, p = 16$	
26	0	26	2	26	10
13	1	8	2	1	1
6	0	2	2		
3	1				
1	1				

Результат:  $26_{10} = 11010_2$ ,  $26_{10} = 222_3$ ,  $26 = 1A_{16}$ .

*Алгоритм перевода правильной десятичной дроби  $N_{10}$  в позиционную систему с основанием  $p$ :*

1. Умножить данное число на новое основание  $p$ .

2. Целая часть полученного произведения является цифрой старшего разряда искомой дроби.

3. Дробная часть полученного произведения вновь умножается на  $p$ , и целая часть результата считается следующей цифрой искомой дроби.

4. Операции продолжать до тех пор, пока дробная часть не окажется равной нулю либо не будет достигнута требуемая точность.

Например, надо перевести десятичную дробь  $0,375$  в двоичную, троичную и шестнадцатеричную систему счисления. Перевод выполнить с точностью до третьего знака.

$0,375_{10} \rightarrow 0, X_2$	$0,375_{10} \rightarrow 0, X_3$	$0,375_{10} \rightarrow 0, X_{16}$
$q = 10, p = 2$	$q = 10, p = 3$	$q = 10, p = 16$
$0,375 \cdot 2 = 0,75$	$0,375 \cdot 3 = 1,125$	$0,375 \cdot 16 = 6,0$
$0,75 \cdot 2 = 1,5$	$0,125 \cdot 3 = 0,375$	
$0,5 \cdot 2 = 1,0$	$0,375 \cdot 3 = 1,125$	
	$0,125 \cdot 3 = 0,375$	

Результат:  $0,375_{10} = 0,011_2$ ;  $0,375_{10} = 0,101_3$ ;  $0,375_{10} = 0,6_{16}$ .

Наиболее часто встречающиеся системы счисления - это двоичная, шестнадцатеричная и десятичная. Как же связаны между собой представления числа в различных системах счисления? Рассмотрим различные способы перевода чисел из одной системы счисления в другую на конкретных примерах.

Пусть требуется перевести число  $567$  из десятичной системы счисления в двоичную систему. Сначала определим максимальную степень двойки, такую, чтобы два в этой степени было меньше или равно исходному числу. В нашем случае это  $9$ , т. к.  $2^9 = 512$ , а  $2^{10} = 1024$ , что больше начального числа. Таким образом, мы получим число разрядов результата. Оно равно  $9 + 1 = 10$ . Поэтому результат будет иметь вид  $1xxxxxxx$ , где вместо  $x$  могут стоять любые двоичные цифры. Найдем вторую цифру результата. Возведем двойку в степень  $9$  и вычтем из исходного числа:  $567 - 2^9 = 55$ . Остаток сравним с числом  $2^8 = 256$ . Так как  $55$  меньше  $256$ , то девятый разряд будет нулем, т.е. результат примет вид  $10xxxxxxx$ . Рассмотрим

восьмой разряд. Так как  $2^7=128>55$ , то и он будет нулевым.

Седьмой разряд также оказывается нулевым. Искомая двоичная запись числа принимает вид  $1000xxxxxx$ .  $2^5=32<55$ , поэтому шестой разряд равен 1 (результат  $10001xxxxx$ ). Для остатка  $55-32=23$  справедливо неравенство  $2^4=16<23$ , что означает равенство единице пятого разряда. Действуя аналогично, получаем в результате число  $1000110111$ . Мы разложили данное число по степеням двойки:

$$567=1*2^9+0*2^8+0*2^7+0*2^6+1*2^5+1*2^4+0*2^3+1*2^2+1*2^1+1*2^0$$

При другом способе перевода чисел используется операция деления в столбик. Рассмотрим то же самое число 567. Разделив его на 2, получим частное 283 и остаток 1. Проведем ту же самую операцию с числом 283. Получим частное 141, остаток 1. Опять делим полученное частное на 2, и так до тех пор, пока частное не станет меньше делителя. Теперь для того, чтобы получить число в двоичной системе счисления, достаточно записать последнее частное, то есть 1, и приписать к нему в обратном порядке все полученные в процессе деления остатки.

Результат, естественно, не изменился: 567 в двоичной системе счисления записывается как 1000110111.

Эти два способа применимы при переводе числа из десятичной системы в систему с любым основанием. Для закрепления навыков рассмотрим перевод числа 567 в систему счисления с основанием 16.

Сначала осуществим разложение данного числа по степеням основания. Искомое число будет состоять из трех цифр, т. к.  $16^2=256 < 567 < 16^3=4096$ . Определим цифру старшего разряда.  $2*16^2=512<567<3*16^2=768$ , следовательно, искомое число имеет вид  $2xx$ , где вместо  $x$  могут стоять любые шестнадцатеричные цифры. Остается распределить по следующим разрядам число 55 ( $567-512$ ).  $3*16=48<55<4*16=64$ , значит, во втором разряде находится цифра 3. Последняя цифра равна 7 ( $55-48$ ). Искомое шестнадцатеричное число равно 237.

Второй способ состоит в осуществлении последовательного деления в столбик, с единственным отличием в том, что делить надо не на 2, а на 16, и процесс деления заканчивается, когда частное становится строго меньше 16.

Конечно, не надо забывать и о том, что для записи числа в

шестнадцатеричной системе счисления, необходимо заменить 10 на А, 11 на В и так далее.

$$\begin{array}{r|l} 567 & 16 \\ \hline \bar{560} & 35 & 16 \\ 7 & \bar{32} & 2 \\ & 3 & \end{array}$$

Операция перевода в десятичную систему выглядит гораздо проще, так как любое десятичное число можно представить в виде  $x = a_0 * p^n + a_1 * p^{n-1} + \dots + a_{n-1} * p^1 + a_n * p^0$ , где  $a_0 \dots a_n$  - это цифры данного числа в системе счисления с основанием  $p$ .

Например, переведем число 4A3F в десятичную систему. По определению,  $4A3F = 4 * 16^3 + A * 16^2 + 3 * 16 + F$ . Заменяя А на 10, а F на 15, получим  $4 * 16^3 + 10 * 16^2 + 3 * 16 + 15 = 19007$ .

Пожалуй, проще всего осуществляется перевод чисел из двоичной системы в системы с основанием, равным степеням двойки (8 и 16), и наоборот. Для того чтобы целое двоичное число записать в системе счисления с основанием  $2^n$ , нужно

- данное двоичное число разбить справа налево на группы по  $n$ -цифр в каждой;
- если в последней левой группе окажется меньше  $n$  разрядов, то дополнить ее нулями до нужного числа разрядов;
- рассмотреть каждую группу, как  $n$ -разрядное двоичное число, и заменить ее соответствующей цифрой в системе счисления с основанием  $2^n$ .

Двоично-шестнадцатеричная таблица

<b>2-ная</b>	0000	0001	0010	0011	0100	0101	0110	0111
<b>16-ная</b>	0	1	2	3	4	5	6	7
<b>2-ная</b>	1000	1001	1010	1011	1100	1101	1110	1111
<b>16-ная</b>	8	9	A	B	C	D	E	F

Двоично-восьмеричная таблица

<b>2-ная</b>	000	001	010	011	100	101	110	111
<b>8-ная</b>	0	1	2	3	4	5	6	7

Например, надо перевести в восьмеричную и шестнадцатеричную системы счисления число  $1011000010,0011001_2$ .

Для этого разобьем исходное число на группы по 3 цифры, начиная от десятичной запятой, и заменим триады восьмеричными цифрами:

001 011 000 010 , 001 100 100	- 2-ичное число;
1 3 0 2 , 1 4 4	- 8-ричное число.

Разобьем число на группы по 4 цифры, начиная от десятичной запятой, и заменим тетрады шестнадцатеричными цифрами:

0010 1100 0010 , 0011 0010	- 2-ичное число;
2 C 2 , 3 2	- 16-ричное число.

Результат:  $1011000010,0011001_2 = 1302,144_8 = 2C2,32_{16}$

### 3.4. Арифметические операции в позиционных системах счисления

Арифметические операции в рассматриваемых позиционных системах счисления выполняются по законам, известным из десятичной арифметики. Двоичная система счисления имеет

#### Пример 1

$$\begin{array}{r} + 110_2 \\ 11_2 \\ \hline 1001_2 \end{array}$$

основание 2, и для записи чисел используются всего две цифры 0 и 1 в отличие от десяти цифр десятичной системы счисления.

Рассмотрим сложение одноразрядных чисел:  $0+0=0$ ,  $0+1=1$ ,  $1+0=0$ . Эти равенства справедливы

как для двоичной системы, так и для десятичной системы. Чему же равно  $1+1$ ? В десятичной системе это 2. Но в двоичной системе нет цифры 2! Известно, что при десятичном сложении  $9+1$  происходит перенос 1 в старший разряд, так как старше 9 цифры нет. То есть  $9+1=10$ . В двоичной системе старшей цифрой является 1. Следовательно, в двоичной системе  $1+1=10$ , так как при сложении двух единиц происходит переполнение разряда и производится перенос в старший разряд. Переполнение разряда наступает тогда, когда значение числа в нем становится равным или большим основания. Для двоичной системы это число равно 2 ( $10_2=2_{10}$ ).

### Пример 2

$$\begin{array}{r} - 110_2 \\ 11_2 \\ \hline 11_2 \end{array}$$

Продолжая добавлять единицы, заметим:  $10_2+1=11_2$ ,  $11_2+1=100_2$  - произошла "цепная реакция", когда перенос единицы в один разряд вызывает перенос в следующий разряд.

Сложение *многоразрядных* чисел происходит по этим же правилам с учетом возможности переносов из младших разрядов в старшие.

Вычитание *многоразрядных* двоичных чисел производится с учетом возможных *заёмов* из старших разрядов.

Действия умножения и деления чисел в двоичной арифметике можно выполнять по общепринятым для позиционных систем правилам.

### Пример 3

$$\begin{array}{r} \times 110_2 \\ 11_2 \\ + \hline 110_2 \\ 110 \\ \hline 10010_2 \end{array}$$

$$\begin{array}{r} -110_2 \overline{)111_2} \\ 11 \ 10_2 \\ \hline 0 \end{array}$$

В основе правил арифметики любой позиционной системы лежат *таблицы сложения и умножения одноразрядных чисел*.

Для двоичной системы счисления:

+	0	1
0	0	1
1	1	10

*	0	1
0	0	0
1	0	1

Аналогичные таблицы составляются для любой позиционной системы счисления. Пользуясь такими таблицами, можно выполнять действия над многозначными числами.

**Пример 4.** Выполнить действия в пятеричной системе счисления:  
 $342_5 + 23_5$ ;  $213_5 \cdot 5_5$ .

*Решение*

Составим таблицы сложения и умножения для пятеричной системы счисления:

+	0	1	2	3	4
0	0	1	2	3	4
1	1	2	3	4	10
2	2	3	4	10	11
3	3	4	10	11	12
4	4	10	11	12	13

*	1	2	3	4
1	1	2	3	4
2	2	4	11	13
3	3	11	14	22
4	4	13	22	31

$342_5$  Выполним сложение.  
 $+ 23_5$  Рассуждаем так: два плюс три равно 10 (по таблице); 0  
 $\hline 420_5$  пишем, 1 - в уме. Четыре плюс два равно 11 (по таблице),  
да еще один, 12. 2 пишем, 1 - в уме. Три да один равно 4  
(по таблице). Результат - 420.

$213_5$  Выполним умножение.  
 $\times 5_5$  Рассуждаем так: трижды три - 14 (по таблице); 4  
 $\hline 1144_5$

пишем, один - в уме. Трижды один дает 3, да плюс один, - пишем 4. Дважды три (по таблице) - 11; 1 пишем, 1 переносим влево. Окончательный результат - 1144.

Если числа, участвующие в выражении, представлены в разных системах, нужно сначала привести их к одному основанию.

**Пример 5.** Сложить два числа:  $17_8$  и  $17_{16}$ .

*Решение*

Приведем число  $17_{16}$  к основанию 8 посредством двоичной системы (пробелами условно обозначено деление на тетрады и триады):  $17_{16} = 10111_2 = 10111_2 = 27_8$ .

Выполним сложение в восьмеричной системе:

$$\begin{array}{r} + 17_8 \\ 27_8 \\ \hline 46_8 \end{array}$$

Сделаем проверку, выполнив те же действия в десятичной системе:

$$17_8 = 1 \cdot 8 + 7 = 15_{10};$$

$$17_{16} = 1 \cdot 16 + 7 = 23_{10};$$

$$15 + 23 = 38_{10};$$

$$46_8 = 4 \cdot 8 + 6 = 32 + 6 = 38_{10}.$$

**Пример 6.** Вычислить выражение , записав результат в двоичной системе счисления.

*Решение*

Приведем числа, участвующие в выражении, в единую систему счисления, например, десятичную:

$$10111_2 = 23_{10};$$

$$51_{16} = 1010001_2 = 81_{10};$$

$$33_8 = 11011_2 = 27_{10}.$$

Выполним указанные действия:  
 $23-81/27=20_{10}$ .

Запишем результат в двоичной системе счисления:  $20_{10}=10100_2$ .

Таким образом, арифметические действия в позиционных системах счисления выполняются по общим правилам. Необходимо только помнить, что перенос в следующий разряд при сложении и заем из старшего разряда при вычитании определяются величиной основания системы счисления.

Вопросы для самоконтроля:

1. Двоичная система счисления: основные методы расчета
2. Десятеричная система счисления: основные методы расчета
3. Перевод дробного числа из двоичной системы счисления в восьмеричную и наоборот

## **4. Основы математической логики**

### **4.1. Основные понятия формальной логики**

Слово *логика* означает совокупность правил, которым подчиняется процесс мышления. Сам термин "логика" происходит от древнегреческого *logos*, означающего "слово, мысль, понятие, рассуждение, закон". *Формальная логика* - наука о формах и законах мышления. Законы логики отражают в сознании человека свойства, связи и отношения объектов окружающего мира. Логика как наука позволяет строить формальные модели окружающего мира, отвлекаясь от содержательной стороны. Основными формами мышления являются *понятия, суждения и умозаключения*.

Понятие - это форма мышления, которая выделяет существенные признаки предмета или класса предметов, отличающие его от других. Например, компьютер, человек, ученики.

Суждения - это форма мышления, в которой утверждается или отрицается связь между предметом и его признаком, отношения между предметами или факт существования предмета и которая может быть либо истинной, либо ложной. Языковой формой выражения суждения является повествовательное предложение. Вопросительные и побудительные предложения суждениями не являются.

Суждения рассматриваются не с точки зрения их смысла и содержания, а только с точки зрения их истинности или ложности. Истинным будет суждение, в котором связь понятий правильно отражает свойства и отношения реальных объектов. "Дважды два равно четырем" - истинное суждение, а вот "Процессор предназначен для печати" - ложное. Суждения могут быть простыми и сложными. "Весна наступила, и грачи прилетели" - сложное суждение, состоящее из двух простых. Простые суждения (высказывания) выражают связь двух понятий. Сложные - состоят из нескольких простых суждений.

Умозаключение - прием мышления, позволяющий на основе одного или нескольких суждений-посылок получить новое суждение (знание или вывод).

Примерами умозаключений являются доказательства теорем в геометрии. Посылками умозаключения по правилам формальной логики могут быть только истинные суждения. Тогда и умозаключение будет истинным. Иначе можно прийти к ложному умозаключению.

*Математическая логика* изучает вопросы применения математических методов для решения логических задач и построения логических схем, которые лежат в основе работы любого компьютера. Суждения в математической логике называют *высказываниями* или *логическими выражениями*. Подобно тому, как для описания действий над переменными был разработан раздел математики алгебра, так и для обработки логических выражений в математической логике была создана *алгебра высказываний*, или *алгебра логики*.

## 4.2. Логические выражения и логические операции

*Логическое выражение* - это символическая запись, состоящая из логических величин (констант или переменных), объединенных логическими операциями (связками).

В булевой алгебре простым высказываниям ставятся в соответствие *логические переменные*, значение которых равно 1, если высказывание истинно, и 0, если высказывание ложно. Обозначаются логические переменные буквами латинского

алфавита.

Существуют разные варианты обозначения истинности и ложности переменных:

Истина	И	True	T	1
Ложь	Л	False	F	0

Связки "НЕ", "И", "ИЛИ" заменяются логическими операциями *инверсия*, *конъюнкция*, *дизъюнкция*. Это основные логические операции, при помощи которых можно записать любое логическое выражение.

**Логическое отрицание (инверсия).**

В обыденной речи мы часто пользуемся словом "НЕ", или словами "НЕВЕРНО, ЧТО", когда хотим что-то отрицать. Пусть, например, кто-то сказал: "Тоска зеленая." (Обозначим это высказывание **A**). Если Вы не согласны, Вы скажете: "Тоска НЕ зеленая." Или: "Неверно, что тоска зеленая." (Ваше высказывание обозначим **B**). Нетрудно заметить, что значения истинности высказываний **A** и **B** находятся в определенной связи: если **A** истинно, то **B** ложно, и наоборот. Операция, с помощью которой из высказывания **A** получается высказывание **B**, называется логическим отрицанием и само высказывание **B** называется отрицанием высказывания **A** и обозначается  $\neg A$ .

Таким образом, отрицанием  $\neg A$  некоторого высказывания **A** называется такое высказывание, которое истинно, когда **A** ложно, и ложно, когда **A** истинно. Отрицание высказывания **A** обозначим  $\neg A$ . Определение отрицания может быть записано с помощью так называемой таблицы истинности:

A	$\neg A$
И	Л
Л	И

В ней указано, какие значения истинности (Истина, Ложь) принимает отрицание  $\neg A$  в зависимости от значений истинности исходного высказывания  $A$ .

*Логическое умножение (конъюнкция)* от латинского conjunctio - союз, связь.

Если два высказывания соединены союзом "И", то полученное сложное высказывание обычно считается истинным тогда и только тогда, когда истинны оба составляющие его высказывания. Если хотя бы одно из составляющих высказываний ложно, то и полученное из них с помощью союза "И" сложное высказывание также считается ложным. Например, возьмем два высказывания: "У кота есть хвост" (**A**), "У зайца есть хвост" (**B**). Сложное высказывание "У кота есть хвост и у зайца есть хвост" истинно, т.к. истинны оба высказывания **A** и **B**. Но если взять другие высказывания: "У кота длинный хвост" (**C**), "У зайца длинный хвост" (**D**), то сложное высказывание "У кота длинный хвост и у зайца длинный хвост" будет ложным, т.к. ложно высказывание (**D**). Таким образом, исходя из обычного смысла союза "И", приходим к определению соответствующей логической операции - конъюнкции.

Таким образом, конъюнкцией двух высказываний  $A$  и  $B$  называется такое высказывание, которое истинно тогда и только тогда, когда истинны оба высказывания  $A$  и  $B$ .

Конъюнкцию высказываний  $A$  и  $B$  мы обозначим:  $A \& B$ . Знак  $\&$  - амперсент - читается как английское "and" (помните Procter & Gamble или Wash & Go?). Часто встречается обозначение  $A \wedge B$ . Иногда, для краткости, пишут просто **AB**.

Определение конъюнкции может быть записано в виде таблицы истинности, в которой для каждого из четырех возможных наборов значений исходных высказываний  $A$  и  $B$  задается соответствующее значение конъюнкции  $A \& B$ :

A	B	A&B
и	и	и
и	л	л

Л	И	Л
Л	Л	Л

Определение конъюнкции двух высказываний естественным образом распространяется на любое конечное число составляющих: конъюнкция  $A_1 \& A_2 \& A_3 \& \dots \& A_N$  истинна тогда и только тогда, когда истинны все высказывания  $A_1, A_2, A_3, \dots, A_N$  (а, следовательно, ложна, когда ложно хотя бы одно из этих высказываний).

*Логическое сложение (дизъюнкция)* от латинского disjunctio - разобшение, различие.

Если два высказывания соединены союзом "ИЛИ", то полученное сложное высказывание обычно считается истинным, когда истинно хотя бы одно из составляющих высказываний. Например, возьмем два высказывания: "Мел черный." (**A**), "Доска черная." (**B**). Высказывание "Мел черный или доска черная" будет истинным, т.к. одно из исходных высказываний (**B**) истинно.

Таким образом, дизъюнкцией двух высказываний называется такое новое высказывание, которое истинно тогда и только тогда, когда истинно хотя бы одно из этих высказываний.

Дизъюнкцию высказываний **A** и **B** мы обозначим символом **A V B** и будем читать: **A** или **B**. Определение дизъюнкции может быть записано в виде таблицы истинности:

A	B	AVB
И	И	И
И	Л	И
Л	И	И
Л	Л	Л

Определение дизъюнкции двух высказываний естественным образом распространяется на любое конечное число составляющих: дизъюнкция  $A_1 V A_2 V A_3 V \dots V A_N$  истинна тогда и только тогда, когда истинно хотя бы одно из высказываний  $A_1, A_2, A_3, \dots, A_N$  (а следовательно, ложна, когда ложны все эти высказывания).

*Логическое следование (импликация)* от латинского implico - тесно

связываю.

В наших рассуждениях, особенно в математических доказательствах, мы часто пользуемся сложными высказываниями, образованными с помощью слов "если..., то...". Здесь высказывание, расположенное после слова "если", называется основанием или посылкой, а высказывание, расположенное после слова "то", называется следствием или заключением.

Рассмотрим пример: из арифметики. Вам должно быть известно, что утверждение "если каждое слагаемое делится на 3, то и сумма делится на 3" истинно, т.е. из высказывания "каждое слагаемое делится на 3" следует высказывание "сумма делится на 3". Посмотрим, какие наборы значений истинности посылки и заключения возможны, когда истинно все утверждение. Возьмем, например, в качестве слагаемых числа 6 и 9. В этом случае истинны и посылка, и заключение, и все утверждение. Если же взять числа 4 и 5, то посылка будет ложной, а заключение истинным. Для чисел 4 и 7 и посылка и заключение ложны. (Если Вы сомневаетесь в истинности высказывания для последнего случая попробуйте произнести его в сослагательном наклонении: если бы числа 4 и 7 делились бы на 3, то и их сумма делилась бы на 3). Очевидно, что только один случай невозможен: мы не найдем таких двух слагаемых, чтобы каждое из них делилось на 3, а их сумма не делилась на 3, т.е. чтобы посылка была истинной, а заключение ложным. Из истины не может следовать ложь, иначе логика теряет смысл. Высказывание "Если А, то В" с логической точки зрения имеет тот же смысл, что и высказывание "неверно, что А истинно и В ложно". Это означает, что функцию импликации можно заменить комбинацией двух функций (отрицания и конъюнкции). Обычно, когда мы хотим установить ложность высказывания "Если А, то В", мы стараемся показать, что возможен случай, когда А истинно, а В ложно (доказательство "от противного"). Обозначим импликацию символом  $\Rightarrow$  и запись "А  $\Rightarrow$  В" будем читать: "Из А следует В".

Таким образом, импликацией  $A \Rightarrow B$  называется высказывание, которое ложно тогда и только тогда, когда А истинно и В ложно.

Запишем это определение в виде таблицы истинности:

A	B	A $\Rightarrow$ B
---	---	-------------------

И	И	И
И	Л	Л
Л	И	И
Л	Л	И

*Логическое тождество (эквиваленция).*

Интуитивно можно догадаться, что высказывания эквивалентны (равносильными), когда их значения истинности одинаковы. Например, эквивалентны высказывания: "железо тяжелое" и "пух легкий", так же как и высказывания: "железо легкое" и "пух тяжелый". Обозначим эквиваленцию символом  $\Leftrightarrow$  и запись "**A**  $\Leftrightarrow$  **B**" будем читать "**A** эквивалентно **B**", или "**A** равносильно **B**", или "**A**, если и только если **B**".

Таким образом, эквиваленцией двух высказываний **A** и **B** называется такое высказывание, которое истинно тогда и только тогда, когда оба эти высказывания **A** и **B** истинны или оба ложны.

Отметим, что высказывание типа "**A**, если и только если **B**" можно заменить высказыванием "Если **A**, то **B** и, если **B**, то **A**" (обдумайте это на досуге и обратите внимание на символ  $\Leftrightarrow$ ). Следовательно, функцию эквиваленции можно заменить комбинацией функций импликации и конъюнкции. Запишем таблицу истинности для эквиваленции:

<b>A</b>	<b>B</b>	<b>A</b> $\Leftrightarrow$ <b>B</b>
И	И	И
И	Л	Л
Л	И	Л
Л	Л	И

Приведем примеры записи сложных высказываний с помощью обозначения логических связок:

"Быть или не быть - вот в чем вопрос." (В. Шекспир)  $A \vee \neg A \Leftrightarrow B$

"Если хочешь быть красивым, поступи в гусары." (К. Прутков)  $A \Rightarrow B$

### 4.3. Построение таблиц истинности для логических функций

*Логическая функция* - это функция, в которой переменные принимают только два значения: логическая единица или логический ноль. Истинность или ложность сложных суждений представляет собой функцию истинности или ложности простых. Эту функцию называют булевой функцией суждений  $f(a, b)$ .

Любая логическая функция может быть задана с помощью таблицы истинности, в левой части которой записывается набор аргументов, а в правой части - соответствующие значения логической функции.

При построении таблицы истинности необходимо учитывать порядок выполнения логических операций. Операции в логическом выражении выполняются слева направо с учетом скобок в следующем порядке:

1. инверсия;
2. конъюнкция;
3. дизъюнкция;
4. импликация и эквивалентность.

Для изменения указанного порядка выполнения логических операций используются круглые скобки.

Предлагается следующий *алгоритм построения таблицы истинности*.

1. Определить *количество наборов входных переменных* - всевозможных сочетаний значений переменных, входящих в выражения, по формуле:  $Q=2^n$ , где  $n$  - количество входных переменных. Оно определяет количество строк таблицы.
2. Внести в таблицу все наборы входных переменных.
3. Определить количество логических операций и последовательность их выполнения.
4. Заполнить столбцы результатами выполнения логических операций в обозначенной последовательности.

$$\text{Количество столбцов} = \left\{ \begin{array}{l} \text{количество переменных} + \\ + \text{ количество операций.} \end{array} \right\}$$

Чтобы не повторить или не пропустить ни одного возможного сочетания значений входных переменных, следует пользоваться одним из предлагаемых ниже способов заполнения таблицы.

**Способ 1.** Каждый набор значений исходных переменных есть код числа в двоичной системе счисления, причем количество разрядов числа равно количеству входных переменных. Первый набор - число 0. Прибавляя к текущему числу каждый раз по 1, получаем очередной набор. Последний набор - максимальное значение двоичного числа для данной длины кода.

Например, для функции от трех переменных последовательность наборов состоит из чисел:

000
001
010
011
100
101
110
111

**Способ 2.** Для функции от трех переменных последовательность данных можно получить следующим путем:

а) разделить колонку значений первой переменной пополам и заполнить верхнюю половину нулями, нижнюю половину единицами;

б) в следующей колонке для второй переменной половинку снова разделить пополам и заполнить группами нулей и единиц; аналогично заполнить вторую половинку;

в) так делать до тех пор, пока группы нулей и единиц не будут

состоять из одного символа.

**Способ 3.** Воспользоваться известной таблицей истинности для двух аргументов. Добавляя третий аргумент, сначала записать первые 4 строки таблицы, сочетая их со значением третьего аргумента, равным 0, а затем еще раз записать эти же 4 строки, но теперь уже со значением третьего аргумента, равным 1. В результате в таблице для трех аргументов окажется 8 строк:

000
010
100
110
001
011
101
111

Например, построим таблицу истинности для логической функции:

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

Количество входных переменных в заданном выражении равно трем ( $A, B, C$ ). Значит, количество входных наборов  $Q = 2^3 = 8$ .

Столбцы таблицы истинности соответствуют значениям исходных выражений  $A, B, C$ , промежуточных результатов  $\bar{A}$  и  $(B \vee C)$ , а также искомого окончательного значения сложного арифметического выражения  $\bar{A} \& (B \vee C)$ :

A	B	C	$\bar{A}$	$B \vee C$	$\bar{A} \& (B \vee C)$
---	---	---	-----------	------------	-------------------------

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

#### 4.4 Логические функции и их преобразования. Законы логики

Для операций конъюнкции, дизъюнкции и инверсии определены законы булевой алгебры, позволяющие производить *тождественные (равносильные) преобразования логических выражений.*

##### Законы логики

1.  $\neg\neg A \Leftrightarrow A$  закон двойного отрицания;
2.  $A \& B \Leftrightarrow B \& A$  коммутативность конъюнкции;
3.  $A \vee B \Leftrightarrow B \vee A$  коммутативность дизъюнкции;
4.  $A \& (B \& C) \Leftrightarrow (A \& B) \& C$  ассоциативность конъюнкции;
5.  $A \vee (B \vee C) \Leftrightarrow (A \vee B) \vee C$  ассоциативность дизъюнкции;
6.  $A \& (B \vee C) \Leftrightarrow (A \& B) \vee (A \& C)$  дистрибутивность конъюнкции относительно дизъюнкции;
7.  $A \vee (B \& C) \Leftrightarrow (A \vee B) \& (A \vee C)$  дистрибутивность дизъюнкции относительно конъюнкции;
8.  $A \& A \Leftrightarrow A$
9.  $A \vee A \Leftrightarrow A$
10.  $A \vee \neg A \Leftrightarrow И$  закон исключенного третьего;
11.  $A \& \neg A \Leftrightarrow Л$  закон непротиворечия;
12.  $A \& И \Leftrightarrow A$
13.  $A \vee И \Leftrightarrow И$
14.  $A \& Л \Leftrightarrow Л$

15.  $\overline{A \vee B} \Leftrightarrow \overline{A} \wedge \overline{B}$   
 16.  $\overline{(A \wedge B)} \Leftrightarrow \overline{A} \vee \overline{B}$  законы де Моргана;  
 17.  $\overline{(A \vee B)} \Leftrightarrow \overline{A} \wedge \overline{B}$   
 18.  $A \Rightarrow B \Leftrightarrow \overline{A} \vee B$  замена импликации.

Основываясь на законах, можно выполнять упрощение сложных логических выражений. Такой процесс замены сложной логической функции более простой, но равносильной ей, называется минимизацией функции.

**Пример 1.** Упростить выражения  $\overline{AB + B}$ ,  $\overline{\overline{BC} + C}$ ,  $\overline{\overline{AC} + B\overline{C}}$  так, чтобы в полученных формулах не содержалось отрицания сложных высказываний.

*Решение*

$$X = \overline{AB + B} = \overline{A + B + B} = \overline{A + B};$$

$$Y = \overline{\overline{BC} + C} = \overline{(B + \overline{C})\overline{C}} = \overline{C};$$

$$\overline{\overline{AC} + B\overline{C}} = \overline{A + \overline{C} + B\overline{C}} = \overline{A + \overline{C}}.$$

**Пример 2.** Минимизировать функцию

$$F(x, y, z) = (x \vee y)z \vee xyz(\overline{z} \vee xy) \vee \overline{z}xy \vee \overline{x}y\overline{z}.$$

*Решение*

$$\begin{aligned} F(x, y, z) &= (x \vee y)z \vee xyz(\overline{z} \vee xy) \vee \overline{z}xy \vee \overline{x}y\overline{z} = \\ &= xz \vee yz \vee 0 \vee xyz \vee y\overline{z}(x \vee \overline{x}) = xz \vee yz. \end{aligned}$$

При упрощении выражения использовались формулы поглощения и склеивания.

**Пример 3.** Найти отрицание следующего высказывания: "Если урок будет интересным, то никто из учеников (Миша, Вика, Света) не будет смотреть в окно".

*Решение*

Обозначим

высказывания:

$Y$	-	-	"Урок	интересный";
$M$	-	"Миша	смотрит	в окно";
$B$	-	"Вика	смотрит	в окно";
$C$	-	"Света	смотрит	в окно".

$$\begin{aligned} \overline{Y \Rightarrow M \& B \& C} &= \overline{\overline{Y} \vee M \& B \& C} = \\ &= \overline{\overline{Y} \& \overline{M \& B \& C}} = Y \& (M \vee B \vee C) \end{aligned}$$

При упрощении выражения использовались формула замены операций и закон де Моргана.

**Пример 4.** Определить участника преступления, исходя из двух посылок:

- 1) "Если Иванов не участвовал или Петров участвовал, то Сидоров участвовал";
- 2) "Если Иванов не участвовал, то Сидоров не участвовал".

*Решение*

Составим выражения:

$I$	-	"Иванов участвовал	в	преступлении";
$P$	-	"Петров участвовал	в	преступлении";
$S$	-	"Сидоров участвовал	в	преступлении".

Запишем посылки в виде формул:

$$\overline{I} \vee P \Rightarrow S \quad \text{И} \quad \overline{I} \Rightarrow \overline{S}.$$

Тогда

$$\begin{aligned} F(I, P, S) &= (\overline{I} \vee P \Rightarrow S) \& (\overline{I} \Rightarrow \overline{S}) = \\ &= \left( \overline{(\overline{I} \vee P)} \vee S \right) \& (I \vee \overline{S}) = (I \overline{P} \vee S) \& (I \vee \overline{S}) = \\ &= I \overline{P} \vee IS \vee I \overline{P} \overline{S} \vee 0 = I \overline{P} \vee IS = I (\overline{P} \vee S). \end{aligned}$$

Проверим результат, используя таблицу истинности:

$I$	$P$	$S$	$\bar{I} \vee P$	$\bar{I} \vee P \Rightarrow S$	$\bar{I} \Rightarrow \bar{S}$	$(\bar{I} \vee P \Rightarrow S)(\bar{I} \Rightarrow \bar{S})$
0	0	0	1	0	1	0
0	0	1	1	1	0	0
0	1	0	1	0	1	0
0	1	1	1	1	0	0
1	0	0	0	1	1	1
1	0	1	0	1	1	1
1	1	0	1	0	1	0
1	1	1	1	1	1	1

*Ответ:* Иванов участвовал в преступлении.

*Построение логической функции по ее таблице истинности*

Мы научились составлять таблицу истинности для логической функции. Попробуем решить обратную задачу. Пусть дана таблица истинности для некоторой логической функции  $Z(X,Y)$ :

X	Y	Z
0	0	1
0	1	0
1	0	1
1	1	0

Рассмотрим строки, где значение истинности функции  $Z$  истинно ( $Z=1$ ). Функцию для этой таблицы истинности можно составить следующим образом:  $Z(X,Y) = (\neg X \& \neg Y) \vee (X \& \neg Y)$ .

Каждой строке, где функция истинна (равна 1), соответствует скобка, представляющая собой конъюнкцию аргументов, причем если значение аргумента 0, то мы берем его с отрицанием. Все скобки соединены между собой операцией дизъюнкции.

Полученную формулу можно упростить, применив законы логики:

$$\begin{aligned} Z(X,Y) \Leftrightarrow ((\neg X \& \neg Y) \vee X) \& ((\neg X \& Y) \vee \neg Y) \Leftrightarrow (X \vee (\neg X \& \neg Y)) \\ \& (\neg Y \vee (\neg X \& \neg Y)) \Leftrightarrow ((X \vee \neg X) \& (X \vee \neg Y)) \& ((Y \vee \neg Y) \& (\neg Y \vee \neg Y)) \\ \Leftrightarrow (1 \& (X \vee \neg Y)) \& ((\neg Y \vee \neg X) \& \neg Y) \Leftrightarrow (X \vee \neg Y) \& ((\neg Y \vee \neg X) \& \neg Y). \end{aligned}$$

Проверьте полученную формулу: составьте таблицу истинности для функции  $Z(X,Y)$ .

Запишите правила конструирования логической функции по ее таблице истинности:

1. Выделить в таблице истинности те строки, в которых значение функции равно 1.

2. Выписать искомую формулу в виде дизъюнкции нескольких логических элементов. Число этих элементов равно числу выделенных строк.

3. Каждый логический элемент в этой дизъюнкции записать в виде конъюнкции аргументов функции.

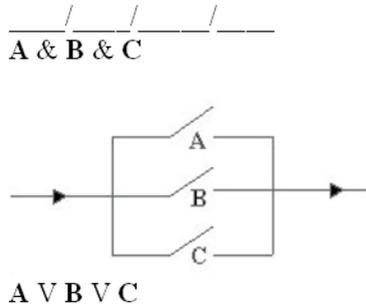
4. Если значение какого-либо аргумента функции в соответствующей строке таблицы равно 0, то этот аргумент мы берем с отрицанием.

#### 4.5 Построение логических схем

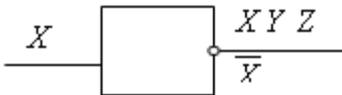
Знания из области математической логики можно использовать для конструирования электронных устройств. Нам известно, что 0 и 1 в логике не просто цифры, а обозначение состояний какого-то предмета нашего мира, условно называемых "ложь" и "истина". Таким предметом, имеющим два фиксированных состояния, может быть электрический ток. Устройства, фиксирующие два устойчивых состояния, называются бистабильными (например, выключатель, реле). Если вы помните, первые вычислительные машины были релейными. Позднее были созданы новые устройства управления электричеством - электронные схемы, состоящие из набора полупроводниковых элементов. Такие электронные схемы, которые преобразовывают сигналы только двух фиксированных напряжений электрического тока (бистабильные), стали называть *логическими элементами*.

На элементарном уровне конъюнкцию можно представить себе в

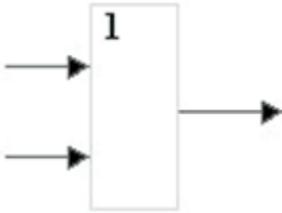
виде последовательно соединенных выключателей, а дизъюнкцию - в виде параллельно соединенных выключателей:



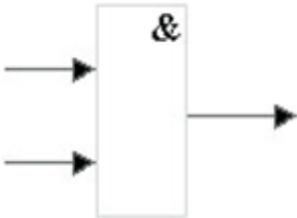
Логические элементы имеют один или несколько входов и один выход, через которые проходят электрические сигналы, обозначаемые условно 0, если "отсутствует" электрический сигнал, и 1, если "имеется" электрический сигнал. Простейшим логическим элементом является *инвертор*, выполняющий функцию отрицания. Если на вход поступает сигнал, соответствующий 1, то на выходе будет 0. И наоборот. У этого элемента один вход и один выход. На функциональных схемах он обозначается:



Логический элемент, выполняющий логическое сложение, называется *дизъюнктор*. Он имеет, как минимум, два входа. На функциональных схемах он обозначается:

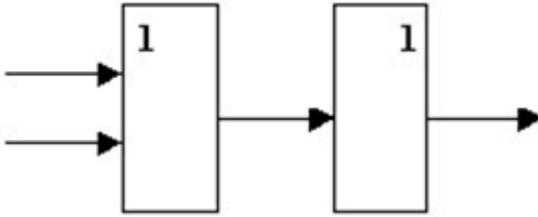


Логический элемент, выполняющий логическое умножение, называется конъюнктор. Он имеет, как минимум, два входа. На функциональных схемах он обозначается:



Специальных логических элементов для импликации и эквивалентности нет, т.к.  $A \Rightarrow B$  можно заменить на  $\neg A \vee B$ ;  $A \Leftrightarrow B$  можно заменить на  $(A \& B) \vee (\neg A \& \neg B)$ .

Другие логические элементы построены из этих трех простейших и выполняют более сложные логические преобразования информации. Сигнал, выработанный одним логическим элементом, можно подавать на вход другого элемента, это дает возможность образовывать цепочки из отдельных логических элементов. Например:

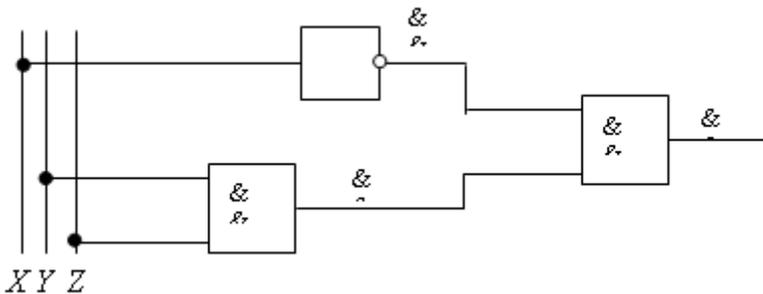


Эта схема соответствует сложной логической функции  $F(A,B) = \neg(A \vee B)$ .

Попробуйте проследить изменения электрического сигнала в этой схеме. Например, какое значение электрического сигнала (0 или 1) будет на выходе, если на входе:  $A=1$  и  $B=0$ .

Такие цепи из логических элементов называются *логическими устройствами*. Логические устройства же, соединяясь, в свою очередь образуют *функциональные схемы* (их еще называют структурными или *логическими схемами*). По заданной функциональной схеме можно определить логическую формулу, по которой эта схема работает, и наоборот.

**Пример 1.** Логическая схема для функции  $F(X,Y,Z) = \bar{X} \& (Y \vee Z)$  будет выглядеть следующим образом:



Правила составления электронных логических схем по заданным таблицам истинности остаются такими же, как для контактных схем.

**Пример 2.** Составить логическую схему для тайного голосования

трех персон А, В, С, условия которого определяются следующей таблицей истинности:

A	0	0	0	0	1	1	1	1
B	0	0	1	1	0	0	1	1
C	0	1	0	1	0	1	0	1
F	0	0	0	1	0	1	1	1

*Решение*

По таблице построим СДНФ логической функции и упростим ее:

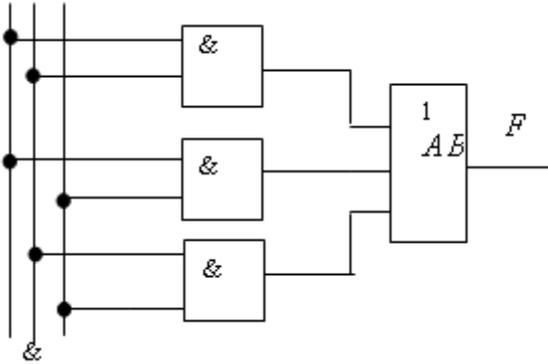
$$\begin{aligned}
 F &= \bar{A}BC \vee A\bar{B}C \vee AB\bar{C} \vee ABC = BC(\bar{A} \vee A) \vee A(\bar{B}C \vee B\bar{C}) = \\
 &= BC \vee A\bar{B}C \vee AB\bar{C} = C(B \vee A\bar{B}) \vee AB\bar{C} = C(B \vee A) \vee AB\bar{C} = \\
 &= BC \vee AC \vee AB\bar{C} = BC \vee AC \vee AB.
 \end{aligned}$$

Правильность полученной формулы можно проверить, составив для нее таблицу истинности:

A	B	C	AB	AC	BC	F
0	0	0	0	0	0	0
0	0	1	0	0	0	0
0	1	0	0	0	0	0
0	1	1	0	0	1	1
1	0	0	0	0	0	0
1	0	1	0	1	0	1
1	1	0	1	0	0	1
1	1	1	1	1	1	1

Значение полученной функции совпадает с исходным, что можно заметить, сравнивая таблицы.

Логическая схема полученной функции имеет вид:



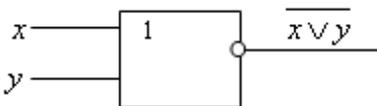
Рассмотрим еще два логических элемента, которые играют роль базовых при создании более сложных элементов и схем.



Логический элемент И-НЕ состоит из конъюнктора и инвертора:

Выходная функция выражается формулой  $F(x, y) = \overline{x \cdot y}$ .

Логический элемент ИЛИ-НЕ состоит из дизъюнктора и инвертора:



Выходная функция выражается формулой  $F(x, y) = \overline{x \vee y}$ .

#### 4.6 Логическая реализация типовых устройств компьютера

Обработка любой информации на компьютере сводится к выполнению процессором различных арифметических и логических операций. Для этого в состав процессора входит так называемое арифметико-логическое устройство (АЛУ). Оно состоит из ряда устройств, построенных на рассмотренных выше логических элементах. Важнейшими из таких устройств являются *триггеры, полусумматоры, сумматоры, шифраторы, дешифраторы, счетчики, регистры.*

##### **Этапы конструирования логического устройства.**

Конструирование логического устройства состоит из следующих этапов:

1. Построение таблицы истинности по заданным условиям работы проектируемого узла (т.е. по соответствию его входных и выходных сигналов).

2. Конструирование логической функции данного узла по таблице истинности, ее преобразование (упрощение), если это возможно и необходимо.

3. Составление функциональной схемы проектируемого узла по формуле логической функции.

После этого остается только реализовать полученную схему.

Попробуем, действуя по этому плану, сконструировать устройство для сложения двух двоичных чисел (одноразрядный полусумматор).

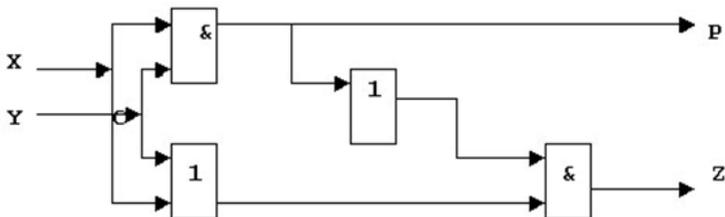
Пусть нам необходимо сложить двоичные числа  $X$  и  $Y$ . Через  $P$  и  $Z$  обозначим первую и вторую цифру суммы:  $X + Y = PZ$ . Вспомните таблицу сложения двоичных чисел.

1. Таблица истинности, определяющая результат сложения, имеет вид:

X	Y	P	Z
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

2. Сконструируем функции  $P(X,Y)$  и  $Z(X,Y)$  по этой таблице:  
 $P(X,Y)=X \ \& \ Y$ ;  $Z(X,Y)=(\neg X \& Y) \vee (X \& \neg Y)$ .  
 Преобразуем вторую формулу, пользуясь законами логики.  
 $(\neg X \& Y) \vee (X \& \neg Y) \Leftrightarrow ((\neg X \& Y) \vee X) \& ((\neg X \& Y) \vee \neg Y) \Leftrightarrow$   
 $(X \vee (\neg X \& Y)) \& (\neg Y \vee (Y \& \neg X)) \Leftrightarrow ((X \vee \neg X) \& (X \vee Y)) \& ((\neg Y \vee Y) \& (\neg Y \vee \neg X)) \Leftrightarrow$   
 $(1 \& (X \vee Y)) \& ((\neg Y \vee \neg X)) \Leftrightarrow (X \vee Y) \& \neg(X \& Y)$ .

3. Теперь можно построить функциональную схему одноразрядного полусумматора:

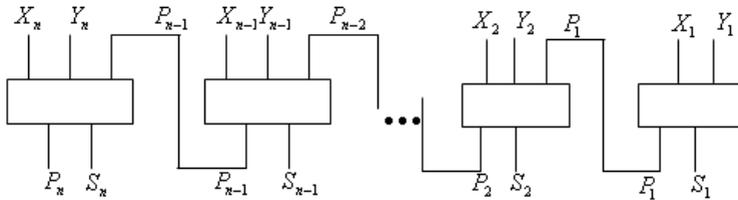


Одноразрядный двоичный сумматор на три входа и два выхода называется *полным одноразрядным сумматором*.

*Сумматор* - это электронная логическая схема, выполняющая суммирование двоичных чисел поразрядным сложением. Сумматор является центральным узлом арифметико-логического устройства процессора. Находит он применение и в других устройствах компьютера. Сумматор выполняет сложение *многозначных двоичных чисел*. Он представляет собой последовательное соединение

одноразрядных двоичных сумматоров, каждый из которых осуществляет сложение в одном разряде. Если при этом возникает переполнение разряда, то перенос суммируется с содержимым старшего соседнего разряда.

Общая схема сумматора:



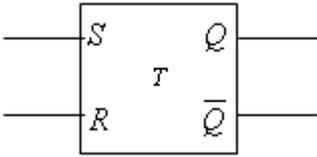
Триггер - электронная схема, применяемая для хранения значения одноразрядного двоичного кода.

Воздействуя на входы триггера, его переводят в одно из двух возможных состояний (0 или 1). С поступлением сигналов на входы триггера в зависимости от его состояния либо происходит переключение, либо исходное состояние сохраняется. При отсутствии входных сигналов триггер сохраняет свое состояние сколь угодно долго.

Термин *триггер* происходит от английского слова *trigger* - защёлка, спусковой крючок. Для обозначения этой схемы в английском языке чаще употребляется термин *flip-flop*, что в переводе означает "хлопанье". Это звукоподражательное название электронной схемы указывает на её способность почти мгновенно переходить ("перебрасываться") из одного электрического состояния в другое.

Существуют разные варианты исполнения триггеров в зависимости от элементной базы (И-НЕ, ИЛИ-НЕ) и функциональных связей между сигналами на входах и выходах (*RS*, *JK*, *T*, *D* и другие).

Самый распространённый тип триггера - это *RS*-триггер (*S* и *R* соответственно от английских *set* - установка, и *reset* - сброс). Условное обозначение *RS*-триггера:



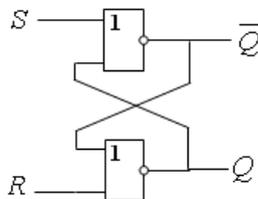
Триггер имеет два симметричных входа  $S$  и  $R$ , которые используются для установки в единичное состояние и сброса, - в нулевое. Еще у него есть два симметричных выхода  $Q$  и  $\bar{Q}$ , причем выходной сигнал  $Q$  является логическим отрицанием сигнала  $\bar{Q}$ .

За единичное состояние триггера условились принимать такое, при котором  $Q=1$ .

На каждый из входов  $S$  и  $R$  могут подаваться входные сигналы в виде кратковременных импульсов  $\underline{\quad 1 \quad}$ . Наличие импульса на входе считается единицей, а его отсутствие - нулем.

Ниже показана схема реализации триггера с помощью элементов ИЛИ-НЕ и соответствующая таблица истинности.

$S$	$R$	$Q$	$\bar{Q}$	Режим триггера
1	0	1	0	Установка 1
0	1	0	1	Установка 0
0	0	Последние значения		Хранение информации
1	1	Запрещено		



Два одинаковых двухвходовых логических элемента ИЛИ-НЕ соединены симметричным образом. Сигнал, поданный на один из входов каждого элемента, снимается с выхода другого. Наличие такого соединения и дает триггеру возможность сохранять свое состояние после прекращения действия сигналов (никакой другой логический элемент не в состоянии поддерживать сигнал на выходе после прекращения действия входного напряжения).

Проанализируем возможные комбинации значений входов  $R$  и  $S$  триггера, используя его схему и таблицу истинности схемы.

1. Пусть поданы сигналы  $S=1$ ,  $R=0$ . Независимо от состояния другого входа на выходе верхнего элемента появится 0. Этот нулевой сигнал подается на вход нижнего элемента, где уже есть  $R=0$ . Выход нижнего элемента станет равным 1. Эта единица возвращается на вход первого элемента. Теперь состояние другого входа ( $S$ ) этого элемента роли не играет: если даже убрать входной сигнал  $S$ , состояние триггера останется без изменения. Поскольку  $Q=1$ , триггер перешел в единичное состояние, устойчивое, пока не придут новые внешние сигналы.

2. При  $S=0$  и  $R=1$  вследствие симметричности схемы все происходит аналогично, но теперь на выходе  $Q$  будет 0. То есть при подаче сигнала на вход  $R$  триггер сбрасывается в устойчивое нулевое состояние.

3. При окончании действия обоих сигналов ( $R=0$  и  $S=0$ ) триггер сохраняет на выходе  $Q$  тот сигнал, который был установлен входным импульсом  $S$  или  $R$ .

4. Подача импульсов одновременно на входы  $R$  и  $S$  может привести к неоднозначному результату, поэтому эта комбинация входных сигналов ( $R=1$  и  $S=1$ ) запрещена.

Один триггер хранит один бит информации. Для хранения одного байта информации необходимо 8 триггеров. Современные микросхемы памяти содержат миллионы триггеров.

По технологии изготовления память делится на статическую и динамическую. На триггерах основана статическая память, а динамическая устроена по принципу конденсатора: заряженный конденсатор соответствует единице, а незаряженный - нулю.

Динамическая память проще по устройству, имеет больший объем и дешевле. В силу этих преимуществ в настоящее время

основной объем оперативного запоминающего устройства компьютера является динамическим.

Однако статическая память имеет более высокое быстродействие. Кэш-память имеет статическую природу, что позволяет согласовать высокое быстродействие процессора и низкую скорость работы динамической памяти.

Конденсаторы динамической памяти постепенно разряжаются через внешние цепи, и потому требуют периодической подзарядки, чтобы не потерять информацию. Этот процесс называется регенерацией памяти, его наличие усложняет подключение микросхем динамической памяти. Микросхема статической памяти сильнее нагревается при работе, так как использует активные элементы - транзисторы.

Некоторое количество триггеров, объединенных вместе общей системой управления, называется *регистром*. Регистры содержатся в различных вычислительных узлах компьютера - процессоре, периферийных устройствах и т.д. Регистр - это устройство, предназначенное для хранения многоразрядного двоичного числового кода, которым можно представлять и адрес, и команду, и данные.

Упрощенно регистр можно представить как совокупность ячеек, в каждой из которых может быть записано одно из двух значений: 0 или 1, то есть один разряд двоичного числа.

Существует несколько типов регистров, отличающихся видом выполняемых операций. Некоторые важные регистры имеют свои названия, например:

*сдвиговый регистр* - предназначен для выполнения операции сдвига;

*счетчики* - схемы, способные считать поступающие на вход импульсы. К ним относятся *T*-триггеры (название от англ. *tumble* - опрокидываться). Этот триггер имеет один счетный вход и два выхода. Под действием сигналов триггер меняет свое состояние с нулевого на единичное и наоборот. Число перебрасываний соответствует числу поступивших сигналов;

*счетчик команд* - регистр устройства управления процессора (УУ), содержимое которого соответствует адресу очередной выполняемой команды; служит для автоматической выборки

программы из последовательных ячеек памяти; *регистр команд* - регистр УУ для хранения кода команды на период времени, необходимый для ее выполнения. Часть его разрядов используется для хранения кода операции, остальные - для хранения кодов адресов операндов.

Техническая сторона логики компьютера основана на технологии транзистора, что позволяет получать одну из двух возможных единиц информации (0 и 1), оперируя с передачей или отсутствием передачи тока. На следующем уровне вступает система носителей (переносчиков) информации - это нули и единицы (0 и 1), которые отражают в себе реальную информацию путем применения, как систем счисления, так и системы команд микропроцессора.

Логика микропроцессора (МП). МП имеет (может иметь) встроенную логику, основанную на формальной логике человека. Таким образом, имеется возможность обрабатывать информацию. Так, например, получая группу информационных потоков (два потока), МП может, используя формальную логику, выдать один искомый поток.

Логика операционной системы. Хотя технически возможна работа на компьютере без использования операционной системы (ОС), так как в ПЗУ находятся (могут находиться) для этого специальные программы, целесообразность использования ОС никем не подвергается сомнению. ОС представляет собой программу (группу программ), которая обеспечивает полный системный интерфейс компьютера.

И, наконец, логика прикладных программ. В данном случае все зависит от фантазии и профессионализма программиста или пользователя.

Проиллюстрируем уровни логики ЭВМ. Получив на два регистра по единице и команду реализовать функцию AND, МП, применив заложенную в нем формальную логику, выдает на результирующий регистр единицу. Эта единица некоторым образом интерпретируется операционной системой, например, как истинность выполненной операции, а затем передается (может передаваться) прикладной программе, которая в свою очередь так же интерпретирует полученную информацию и производит в соответствии с этим некоторые действия.

## Вопросы для самоконтроля

1. Основные логические операции: конъюнкция, дизъюнкция (оба вида), отрицание, импликация, эквивалентность. Примеры логических выражений.
2. Таблица истинности. Примеры.  $A$  and not  $A$ ;  $A$  or not  $A$
3. Основные законы математической логики: перестановочное, сочетательное и распределительное
4. Законы де Моргана (закон отрицания).
5. (Совершенная) дизъюнктивная нормальная форма. Примеры.

## 5. Программное обеспечение компьютера

### 5.1 Программное обеспечение компьютера. Классификация

Бурное развитие новой информационной технологии и расширение сферы ее применения привели к интенсивному развитию программного обеспечения (ПО). Достаточно отметить, что в 1996 г. мировым сообществом на программное обеспечение затрачено свыше 110 млрд долларов. Причем тенденции развития ПО показывают, что динамика затрат имеет устойчивую тенденцию к росту, примерно 20% в год.

Под программным обеспечением информационных систем понимается совокупность программных и документальных средств для создания и эксплуатации систем обработки данных средствами вычислительной техники.

В зависимости от функций, выполняемых программным обеспечением, его можно разделить на:

- системные программы (иногда называют базовым программным обеспечением);
- прикладные программы;
- среды программирования.

К *системным* относятся прежде всего операционные системы и программы, входящие в состав операционной системы (например, драйвера для различных устройств компьютера (от английского слова "drive" - управлять), т.е. программы, управляющие работой устройств: драйвера для сканера, принтера и т.д.). Кроме

операционных систем еще относятся обслуживающее программное обеспечение (их ещё называют сервисные или утилиты, от английского слова "utilize" - использовать) для обслуживания дисков, архиваторы, антивирусные программы и т.д.

К *прикладным* относятся программы, предназначенные для решения задач в различных сферах деятельности человека (бухгалтерские программы, текстовые и графические редакторы, базы данных, экспертные системы, переводчики, энциклопедии, обучающие, тестовые и игровые программы и т.д.).

К *средам программирования* относятся инструментальные средства для создания новых программ (ЛОГО, QuickBASIC, Pascal, Delphi и т.д.)

## **5.2 Системное программное обеспечение. Операционные системы**

Операционная система - комплекс программ, постоянно (псевдопостоянно) находящихся в памяти ЭВМ, организующий управление устройствами машины и ее взаимодействие с пользователем (интерфейс). В операционную систему обычно входят следующие программы: стартовая программа, диспетчер (монитор или супервизор) (очередность исполнения программ), редакторы, загрузчики (для ввода программ в ОЗУ), файловая система, административная система (учет ресурсов), а так же базовое программное обеспечение.

Классификацию операционных систем можно проводить:

- по принципу организации пользовательского интерфейса
- по количеству одновременно обрабатываемых задач
- по количеству одновременно работающих пользователей

Нормальная работа операционной системы, а значит и компьютера, зависит от двух основных факторов:

1. Неприкосновенность системных файлов.
2. Оптимальное состояние памяти - наличие как свободного пространства на диске, так и нормальное (нефрагментированное) расположение файлов на диске.

Для достижения оптимального состояния памяти пользователю достаточно знать следующее. Во-первых, во время своей работы большинство программ создают временные файлы, которые самостоятельно не всегда (по различным причинам) удаляют с "винчестера". Такие временные файлы имеют расширение ".tmp". Пользователь может и должен удалять их "вручную", произведя поиск tmp-файлов. Большинство tmp-файлов можно обнаружить в каталогах TMP и TEMP. Во-вторых, время работы с дисками происходят постоянные операции записи и удаления файлов, в результате чего информация на магнитном носителе приобретает фрагментарный характер, что тормозит процесс ее поиска операционной системой. Для устранения проблем фрагментарности диски необходимо периодически дефрагментировать, для чего специалистами разработана масса программ. Операционная система Windows имеет и свою программу дефрагментации "Defrag".

Операционная система (ОС) - комплекс программных средств, который загружается при включении компьютера и обеспечивает:

- загрузку в оперативную память и выполнение всех программ;
- управление ресурсами компьютера (оперативной памятью, процессорным временем, файловой системой, внешними устройствами);
- диалог пользователя с компьютером, предоставляя удобный способ взаимодействия (*интерфейс*).

***Интерфейс*** - это совокупность средств и правил, которые обеспечивают взаимодействие устройств, программ и человека.

В зависимости от объектов взаимодействия интерфейс определяют как *пользовательский, аппаратный, программный*. Например, интерфейс между пользователем и программно-аппаратными средствами компьютера называют пользовательским интерфейсом, а между аппаратным и программным обеспечением - аппаратно-программный интерфейс.

### **5.3 Файлы и файловая система**

Логическая структура дисков состоит из:

- загрузочной области,
- FAT-области,
- области каталогов,
- области данных.

Основное назначение файлов - хранить информацию. Они также предназначены для передачи данных от программы к программе и от системы к системе. Другими словами, файл - это хранилище стабильных и мобильных данных. Но, файл - это нечто большее, чем просто хранилище данных. Обычно файл имеет имя, атрибуты, время модификации и время создания.

Понятие файла менялось с течением времени. Операционные системы первых больших ЭВМ представляли файл, как хранилище для базы данных и, поэтому файл являлся набором записей. Обычно все записи в файле были одного размера, часто по 80 символов каждая. При этом много времени уходило на поиск и запись данных в большой файл.

В конце 60-х годов наметилась тенденция к упрощению операционных систем, что позволило использовать их на менее мощных компьютерах. Это нашло свое отражение и в развитии операционной системы Unix. В Unix под файлом понималась последовательность байтов. Стало легче хранить данные на диске, так как не надо было запоминать размер записи.

Unix оказал очень большое влияние на другие операционные системы персональных компьютеров. Почти все они поддерживают идею Unix о том, что файл - это просто последовательность байтов. Файлы, представляющие собой поток данных, стали использоваться при обмене информацией между компьютерными системами. Если используется более сложная структура файла (как в операционных системах OS/2 и Macintosh), она всегда может быть преобразована в поток байтов, передана и на другом конце канала связи воссоздана в исходном виде.

Итак, мы можем считать, что файл - это поименованная последовательность байтов, имеющая собственное имя и хранящаяся на внешнем носителе.

*Файловая структура* представляет собой систему хранения файлов на запоминающем устройстве, например, диске. Файлы

организованы в каталоги (иногда называемые директориями или папками). Любой каталог может содержать произвольное число подкаталогов, в каждом из которых могут храниться файлы и другие каталоги.

Способ, которым данные организованы в байты, называется *форматом* *файла*.

Для того чтобы прочесть файл, например, электронной таблицы, необходимо знать, каким образом байты представляют числа (формулы, текст) в каждой ячейке; чтобы прочесть файл текстового редактора, надо знать, какие байты представляют символы, а какие шрифты или поля, а также другую информацию.

Программы могут хранить данные в файле таким способом, какой выберет программист. Зачастую предполагается, однако, что файлы будут использоваться различными программами. По этой причине многие прикладные программы поддерживают некоторые наиболее распространенные форматы, так что другие программы могут понять данные в файле. Компании по производству программного обеспечения (которые хотят, чтобы их программы стали "стандартами"), часто публикуют информацию относительно форматов, которые они создали, чтобы их можно было бы использовать в других приложениях.

Все файлы условно можно разделить на две части - текстовые и двоичные.

*Текстовые* файлы - наиболее распространенный тип данных во всем компьютерном мире. Для хранения каждого символа чаще всего отводится один байт, а кодирование текстовых файлов выполняют с помощью специальных таблиц, в которых каждому символу соответствует определенное число, не превышающее 255. Файл, для кодировки которого используется только 127 первых чисел, называется ASCII-файлом (сокращение от American Standard Code for Information Intercange - американский стандартный код для обмена информацией), но в таком файле не могут быть представлены буквы, отличные от латиницы (в том числе и русские). Большинство национальных алфавитов можно закодировать с помощью восьмибитной таблицы. Для русского языка наиболее популярны на данный момент три кодировки: Koi8-R, Windows-1251 и, так называемая, альтернативная (alt) кодировка. Подробнее о

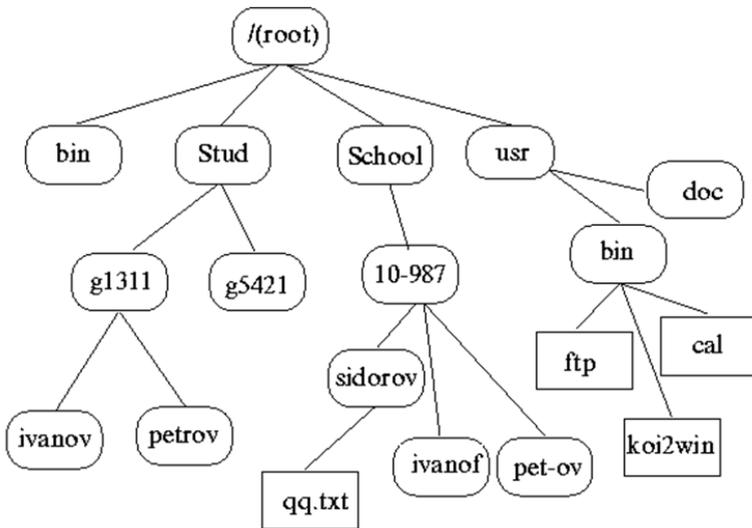
кодировании русского текста рассказано в главе "Обработка документов".

Такие языки, как китайский, содержат значительно больше 256 символов, поэтому для кодирования каждого из них используют несколько байтов. Для экономии места зачастую применяется следующий прием: некоторые символы кодируются с помощью одного байта, в то время как для других используются два или более байтов. Одной из попыток обобщения такого подхода является стандарт *Unicode*, в котором для кодирования символов используется диапазон чисел от нуля до 65 536. Такой широкий диапазон позволяет представлять в численном виде символы языка людей из любого уголка планеты.

Но чисто текстовые файлы встречаются все реже. Люди хотят, чтобы документы содержали рисунки и диаграммы и использовали различные шрифты. В результате появляются форматы, представляющие собой различные комбинации текстовых, графических и других форм данных.

*Двоичные* файлы, в отличие от текстовых, не так просто просмотреть и в них, обычно, нет знакомых нам слов - лишь множество непонятных символов. Эти файлы не предназначены непосредственно для чтения человеком. Примерами двоичных файлов являются исполняемые программы и файлы с графическими изображениями.

Файловую систему можно представить в виде дерева, только корень (root) этого дерева находится вверху, а ветви растут вниз. Листья этого дерева и есть файлы, а разветвления - каталоги (директории). На следующем рисунке для изображения файлов используются прямоугольники, а для каталогов - овалы.



В каждом каталоге (разветвлении) может быть много различных файлов и каталогов. В этом смысле каталог выполняет две важные функции: в каталоге хранятся файлы и каталог соединяется с другими каталогами, как ветвь дерева соединяется с другими ветвями. Каждый каталог может содержать множество каталогов, но сам должен быть потомком только одного каталога.

При входе в систему пользователь оказывается в своей домашней директории, в которой он может создавать файлы, удалять их и копировать. Имя, которое дается каталогу или файлу при его создании, не является полным. Полным именем является его путьевое имя. В ОС Linux путьевое имя состоит из имен всех каталогов, образующих путь к файлу от вершины дерева (/). Эти имена отделяются друг от друга символами "косая черта" (/, "слеш"). Путьевые имена даются и файлам. Фактически имя, которым система обозначает файл, представляет собой комбинацию имени файла с именами каталогов, образующих путь от корневого каталога до каталога данного файла.

Путь может быть абсолютным и относительным. *Абсолютный путь* - это полное имя файла или каталога, начинающееся символом корневого каталога. *Относительный путь* начинается символом

текущего (рабочего) каталога и представляет собой обозначение пути к файлу относительно рабочего каталога.

Информация, содержащаяся на диске, становится доступной для просмотра, загрузки и редактирования при использовании специальных программ - "операционных оболочек" (Norton Comander, Диска командир, Проводник и др.). Наиболее часто встречающейся проблемой при работе с выше названными программами является ошибочное удаление или редактирование программных модулей (файлов с расширением .exe, .com, .dll, .sys, .vxd, .386, .drv ) , - поэтому не следует обращаться к файлам с приведенными выше расширениями через "операционные оболочки", так как работа с данными файлами - это прерогатива исключительно программистов.

Для просмотра файлов с иными расширениями существуют специальные программы, которые правильно отображают формат файла:

Наиболее часто встречающиеся расширения:

<b>Архивные файлы</b>	.arj, .arc, .zip, .zoo, .pak, .lzh, .rar, .cab
<b>Графические файлы</b>	.bmp, .wmf, .pic, .wpg, .wpb, .tif, .gif, .icp, .pcx, .grf, .drw, .clp, .msp, .psd, .pcd
<b>Файлы редакторов документов</b>	.txt, .doc, .wps, .vw4, .wri, .sam, .jw, .ws7, .pdf, (.html)
<b>Файлы табличных процессов</b>	.wks, .wki, .wk3, .wkt, .wxq, .wrk, .xls, .wqi, .vpp, .nod

Логическая структура диска (как жесткого, так и съемного) предусматривает наличие области каталогов и области файлов. Каталоги в операционных системах семейства WINDOWS представлены папками. Папки (каталоги) могут содержать как следующий уровень каталогов (папок), так и непосредственно файлы. От умелого размещения пользователем данных на диске в системе каталогов зависит качество работы на ПЭВМ, поэтому прежде, чем начинать создавать сложные системы документооборота необходимо продумать структуру каталогов. Как правило, бывает достаточно использовать предметную

каталогизацию, однако при значительном увеличении объемов документации возникает необходимость вводить объектное разбиение каталогов. Таким образом, если заранее известно, что количество документов по предметному (тематическому) разбиению сделает работу оператора ПЭВМ неудобной, целесообразным может стать ввести разбиение по объектам, а уже внутри данной системы использовать предметную каталогизацию. Как показывает практика, работа оператора тем продуктивнее, чем продуманнее создана им структура каталогов (папок). Более того, при объектном использовании системы каталогов становится возможным более продуктивно использовать механизм разграничения допуска к информации, хранимой на диске.

Проводник отображает структуру каталогов - в левой части экрана и содержание выбранной папки - в правой части экрана. Для того чтобы отобразить содержание конкретной папки необходимо просто щелкнуть по ней два раза. Выбранные файлами можно:

а) переименовывать - однократный щелчок мышью по имени файла;

б) перемещать и копировать - с помощью приема перетащить (скопировать - удерживая ) мышью; используя возможности меню "Правка"; или воспользовавшись специальными кнопками на панели инструментов.

г) создать папку или ярлык (ссылку на программу или документ);

д) найти и перейти к конкретной папке или файлу ("сервис");

е) изменить параметры отображения данных в окне ("сервис" > "параметры");

ж) установить или изменить "трактовку расширений" ("сервис" > "параметры").

В обычном состоянии двойной щелчок по имени файла не приводит ни к каким действиям, только наличие "трактовки расширений" (когда операционной системе сообщено какие расширения соответствуют каким программам) позволяет реализовать данный механизм загрузки документов (программ). Если трактовка расширений не установлена или требуется загрузить документ не в стандартную для него программу, то необходимо загрузить программу двойным щелчком на ее имени, а уже затем из меню программы загружать требуемый документ.

Интерфейс "Проводника" имеет много общего с интерфейсом обычного каталога (отсутствует только меню "Сервис"), поэтому если требуется оперативный доступ к каталогам можно использовать простые папки, которые через "Другие папки" позволяют производить обзор содержимого дисков.

#### 5.4 Служебные программы

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

По функциональным возможностям сервисные средства можно подразделить на средства:

- улучшающие пользовательский интерфейс;
- защищающие данные от разрушения и несанкционированного доступа;
- восстанавливающие данные;
- ускоряющие обмен данными между диском и ОЗУ;
- средства архивации и разархивации;
- антивирусные средства.

По способу организации и реализации сервисные средства могут быть представлены: оболочками, утилитами и автономными программами. Разница между оболочками и утилитами зачастую выражается лишь в универсальности первых и специализации вторых.

Оболочки, являющиеся надстройками над операционными системами (ОС), называются операционными оболочками. Оболочки являются как бы настройками над операционной системой. Утилиты и автономные программы имеют узкоспециализированное назначение и выполняют каждая свою функцию. Но утилиты, в отличие от автономных программ, выполняются в среде соответствующих оболочек. При этом они конкурируют в своих функциях с программами ОС и другими утилитами. Поэтому классификация сервисных средств по их функциям и способам

реализации является достаточно размытой и весьма условной. Оболочки предоставляют пользователю качественно новый интерфейс и освобождают его от детального знания операции и команд ОС.

Функции большинства оболочек, например семейства MS-DOS, направлены на работу с файлами и каталогами и обеспечивают быстрый поиск файлов; создание, просмотр и редактирование текстовых файлов; выдачу сведений о размещении файлов на дисках, о степени занятости дискового пространства и ОЗУ. Все оболочки обеспечивают ту или иную степень защиты от ошибок пользователя, что уменьшает вероятность случайного уничтожения слайдов.

Среди имеющихся оболочек для семейства MS-DOS наиболее популярна оболочка Norton Commander. Утилиты предоставляют пользователю дополнительные услуги (не требующие разработки специальных программ) в основном по обслуживанию дисков и файловой системы. Эти утилиты чаще всего позволяют выполнять следующие функции:

- обслуживание дисков (форматирование, обеспечение сохранности информации, возможности ее восстановления в случае сбоя и т. д.);
- обслуживание файлов и каталогов (аналогично оболочкам);
- создание и обновление архивов;
- предоставление информации о ресурсах компьютера, о дисковом пространстве, о распределении ОЗУ между программами;
- печать текстовых и других файлов в различных режимах и форматах;
- защита от компьютерных вирусов.

Из утилит, получивших наибольшую известность, можно назвать многофункциональный комплекс Norton Utilities. 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 ЭВМ и ее отдельных частей, в том числе автоматического поиска ошибок и неисправностей с определенной локализацией их в ЭВМ;
- специальные программы диагностики и контроля вычислительной среды информационной системы в целом, в том числе программно-аппаратный контроль, осуществляющий автоматическую проверку работоспособности системы обработки данных перед началом работы вычислительной системы в очередную производственную смену.

### 5.5 Компьютерные вирусы. Антивирусные программы

Как уже отмечалось, компьютер работает исключительно под управлением программ (программного обеспечения). Это делает его по-настоящему универсальным устройством, которое может выполнять роль музыкального центра, телевизора, пишущей машинки и т. д. Программы пишут программисты и у некоторых из них появляется желание придумать что-то эдакое. Иногда это невинные шалости, в других случаях они имеют явную зловещую направленность. До тех пор, пока человек, сидящий за компьютером, мог контролировать работу всех программ и знал, что и зачем он запустил, все было нормально. Но потом появились программы, которые, не спрашивая ничего разрешения, запускались, копировались в разные места диска и "заражали" другие программы (заменяли часть полезного кода рабочей программы своим или изменяли его). С этого момента и нужно начинать разговор о "компьютерных вирусах".

Отдельно хочется подчеркнуть, что практически все вирусы функционируют в операционных системах семейства MS Windows и в MS DOS.

*Компьютерным вирусом* называется программа (некоторая совокупность выполняемого кода), которая способна создавать свои копии (не обязательно полностью совпадающие с оригиналом) и внедрять их в различные объекты, ресурсы компьютерных систем и сетей без ведома пользователя. При этом копии сохраняют

способность дальнейшего распространения.

Компьютерные вирусы, как и биологические вирусы, ставят перед собой три задачи - заразить, выполнить, размножиться. Заражается компьютер "снаружи", когда человек запускает на исполнение некую программу, которая либо заражена вирусом (т. е. при ее выполнении запускается и вирус), либо сама является вирусом.

Поведение вирусов разнообразно. Некоторые вирусы просто "осыпали" буквы с экрана монитора или рисовали безобидные рисунки. Такие вирусы считаются наиболее безвредными. Другие могут переименовывать файлы на диске, стирать их. Эти, без сомнения, гораздо опаснее. А вирус "Win95.CIH" может испортить микросхему BIOS компьютера. Трудно сказать, что хуже - потеря информации или выход из строя компьютера.

И, наконец, вирус размножается, то есть дописывает себя везде, где он имеет шанс выполниться. Есть вирусы, которые достаточно один раз запустить, после чего они постоянно при загрузке компьютера активно включаются в работу и начинают заражать все исполняемые файлы.

Появились вирусы, использующие возможности внутреннего языка программ серии Microsoft Office. Они содержатся в файлах, подготовленных в редакторе Word или в электронных таблицах Excel. Для заражения компьютера достаточно открыть такой документ. Так как все больше людей использует Интернет, то последний все чаще становится рассадником заразы. Теперь достаточно зайти на некий сайт и нажать на кнопку формы, чтобы заполнить какой-нибудь вирус.

В последнее время широко распространился вид почтовых вирусов, играющих на любопытстве людей. Например, вам приходит письмо с признанием в любви и приложенными фотографиями. Первое движение - посмотреть содержимое письма. И как результат, - все фотографии и музыка на вашей машине пропали, а вместо них злобный вирус "I Love You" (или подобный ему). Кроме того, он еще и пошлет себя всем, кто записан в вашей адресной книге.

Троянские программы отличаются от вирусов тем, что они вместо разрушительных действий собирают и отправляют по известным им адресам пароли и другую секретную информацию пользователя. Такая программа может давать злоумышленнику полный доступ к

вашим программам и данным.

К сожалению, единственный действенный метод не "заразить" компьютер - не включать компьютер вовсе. Можно еще посоветовать ничего не устанавливать и ничего не запускать. Только тогда какой смысл иметь компьютер?

Поэтому широко используются *антивирусы* - программы, призванные обнаруживать и удалять известные им "нехорошие программы". Наиболее представительными являются DrWeb, Antiviral Toolkit Pro (AVP), ADInf. При использовании таких программ главное - постоянное обновление антивирусных баз.

И все-таки очень важно не запускать неизвестно что. Или установить *антивирусный монитор* (который отличается от *антивирусного сканера*, занимающегося тотальной проверкой файлов). Когда вы запускаете тот же DrWeb на проверку дисков - это антивирусный сканер. А в комплекте с ним идет некий Spider - вот это антивирусный монитор.

Однако при борьбе с вирусами не стоит впадать в крайность и стирать все подряд. При этом вы можете случайно удалить важные системные файлы, что приведет к невозможности работы на компьютере. На этом построено действие "психологических" вирусов, рассчитанных именно на то, что пользователь своими руками разрушит систему.

Основные признаки появления в системе вируса:

- замедление работы некоторых программ;
- увеличение размеров файлов (особенно выполняемых), хотя это достаточно сложно заметить (попробуйте Adinf);
- появление не существовавших ранее "странных" файлов, особенно в каталоге Windows или корневом;
- уменьшение объема доступной оперативной памяти;
- внезапно возникающие разнообразные видео и звуковые эффекты;
- заметное снижение скорости работы в Интернете (вирус могут передавать информацию по сети);
- жалобы от друзей (или провайдера) о том, что к ним приходят непонятные письма - вирусы любят рассылать себя по почте.

В операционной системе Linux вирусы в были выявлены только в лабораторных условиях. Несмотря на то, что некоторые образцы Linux-вирусов действительно обладали всеми необходимыми способностями к размножению и автономной жизни, ни один из них так и не был зафиксирован в "диком" виде. Использование ОС Linux защищает от вирусов гораздо лучше, чем любые антивирусные программы в MS Windows.

## 5.6 Системы программирования

Транслятором языка программирования называется программа, осуществляющая перевод текста программы с языка программирования в (как правило) машинный код.

Комплекс средств, включающих в себя входной язык программирования, транслятор, машинный язык, библиотеки стандартных программ, средства отладки оттранслированных программ и компоновки их в единое целое, называется системой программирования. В системе программирования транслятор переводит программу, написанную на входном языке программирования, на язык машинных команд конкретной ЭВМ. В зависимости от способа перевода с входного языка (языка программирования) трансляторы подразделяются на компиляторы и интерпретаторы. В компиляции процессы трансляции и выполнения программы разделены во времени. Сначала компилируемая программа преобразуется в набор объектных модулей на машинном языке, которые затем собираются (компонуются) в единую машинную программу, готовую к выполнению и сохраняемую в виде файла на магнитном диске. Эта программа может быть выполнена многократно без повторной трансляции.

Интерпретатор осуществляет пошаговую трансляцию и немедленное выполнение операторов исходной программы: каждый оператор входного языка программирования транслируется в одну или несколько команд машинного языка, которые тут же выполняются без сохранения на диске. Таким образом, при интерпретации программа на машинном языке не сохраняется и поэтому при каждом запуске исходной программы на выполнение ее нужно (пошагово) транслировать заново. Главным достоинством

интерпретатора по сравнению с компилятором является простота.

Входной язык программирования называется языком высокого уровня по отношению к машинному языку, называемому языком низкого уровня.

Особое место в системе программирования занимают ассемблеры, представляющие собой комплекс, состоящий из входного языка программирования ассемблера и ассемблер-компилятора. Ассемблер представляет собой мнемоническую (условную) запись машинных команд и позволяет получить высокоэффективные программы на машинном языке. Однако его использование требует высокой квалификации программиста и больших затрат времени на составление и отладку программ.

Наиболее распространенными языками программирования являются: Pascal, Basic, C++, Fortran и др. Тенденции развития - появление языков четвертого поколения типа Visual Basic.

## 5.7 Архивация

Архиватор - это программа, которая сжимает файл или группу файлов в один *архивный* файл с целью уменьшения их размера. При этом не теряется ни бита информации, и любой файл можно из архива извлечь. Что дает архивация? Во-первых, экономия места на диске, во-вторых, на дискете можно перенести большой объем информации, в-третьих, есть возможность пересылать большие файлы по электронной почте.

Наиболее известные архиваторы - это архиваторы ZIP, ARJ, RAR, GZIP, LHA, HA, ACE. Архив, созданный тем или иным архиватором, имеет расширение, соответствующее названию архиватора. Например, расширение файла archive.rar говорит о том, что он был создан с помощью архиватора RAR.

Архиваторы различаются возможностями и качеством сжатия, которое зависит также и от типа сжимаемых данных. Некоторые архиваторы лучше работают с одними типами данных, но плохо показывают себя с другими. Создать универсальный архиватор невозможно. К важным функциям архиваторов относят создание многотомных архивов и самораспаковывающихся архивов.

Многотомные архивы - это архивы, разбитые на несколько

отдельных файлов. Их применяют, когда необходимо перенести большой объём информации на дискетах: на каждую дискету помещается отдельный том архива. При извлечении данных из многотомного архива архиватор будет последовательно обрабатывать том за томом, и запрашивать смену дискеты.

Самораспаковывающиеся архивы используются в тех случаях, когда необходимо перенести информацию на другой компьютер, но неизвестно, установлен ли там соответствующий архиватор. Самораспаковывающийся архив представляет собой исполняемый (.EXE) файл, который включает в себя заархивированные данные и программу для их распаковки.

Как иллюстрацию сжатия информации рассмотрим пример одного из способов сжатия. Допустим, есть такая строчка символов: "АААААААБББББВВВВВВВВВВВВВГГГГГГ". Длина такой строчки - 28 символов. Но в ней очень много повторяющихся одинаковых символов, поэтому можно просто записать количество повторений и сам повторяющийся символ, в результате получится следующее: "7А5Б10В6Г". Строчка стала занимать всего 9 символов, т.е. сократилась более чем в три раза!

Разумеется, такой метод сжатия будет работать не для всех данных, например, возьмём такую строчку: "АБВВГАБГВАВГБАГ". Её длина - 15 символов. Если мы применим к ней тот же метод, то её длина увеличится вдвое: "1А1Б1В1Б1Г1А1Б1Г1В1А1В1Г1Б1А1Г". Отсюда следует важный вывод: один и тот же алгоритм сжатия для одних исходных данных сокращает их размер, а для других может и увеличить.

Алгоритмов сжатия данных существует великое множество. Некоторые алгоритмы подходят для одних типов данных, другие - для других. Хорошо сжимаются простые растровые изображения, не содержащие большого количества деталей (.BMP, .PSD и др.). Коэффициент сжатия во многом зависит от используемого алгоритма и сложности изображения. Например, чёрно-белое изображение чертёжного качества может быть сжато даже в сотню раз, цветные рисунки - в пять-десять раз, а цветные изображения высокого фотографического качества сжимаются менее чем в два раза. Хорошо сжимаются текстовые файлы (.TXT, .DOC, .PAS и др.). Если файл содержит текст, написанный на естественном языке,

например, повесть или рассказ, то коэффициент сжатия будет равен двум-трём, а если в файле записан текст программы, то коэффициент сжатия может достигать пяти и выше. Коэффициент сжатия исполняемых файлов (.EXE, и др.) также сильно колеблется, однако, в среднем он равен примерно трём.

В отношении звука (.WAV, .AU), использование универсальных методов сжатия редко даёт хорошие результаты - звуковой файл сокращается всего на 20-40%. То же самое касается и высококачественных изображений, имеющих много деталей. Поэтому для этих типов изображений используются специальные методы сжатия, особо хорошие результаты дают так называемые *алгоритмы сжатия с потерями*. Одна из идей этих алгоритмов состоит в том, что человеческий глаз и ухо не очень восприимчивы к некоторым мелким деталям изображения или звука, поэтому лишнюю информацию можно просто отбросить. Разумеется, кроме отбрасывания лишней информации, используются и другие алгоритмы, в результате чего достигается несравнимый коэффициент сжатия при минимальных потерях качества (с точки зрения восприятия человека). Такой подход используется в файлах формата JPEG, предназначенного для хранения статичных изображений и MPEG, предназначенного для хранения видео- и аудиоинформации. Сегодня особенно популярны такие форматы, как MPEG-3 (или MP3), предназначенный для хранения аудиоинформации и позволяющий достигнуть десятикратного сжатия почти без потери качества, и MPEG-4, используемый для хранения видеофильмов.

Плохо архивируются, либо вообще увеличиваются в размере файлы, данные в которых уже сжаты, в том числе архивы (.RAR, .ZIP, .ARJ и др.), графические файлы тех форматов, которые имеют собственное сжатие (.GIF, .JPG, .PNG и др.), аудиофайлы (.MP3), видеофайлы (.MPG, .AVI и др.), самораспаковывающиеся архивы (.EXE).

## Вопросы для самоконтроля

1. Понятие "программное обеспечение". Виды программного обеспечения.

2. Понятие "операционная система". Развитие операционных систем.
3. Виды операционных систем. Состав операционных систем.
4. Особенности ОС Windows. Основные объекты и приемы управления в ОС Windows.
5. Работа с файловой системой в ОС Windows.
6. Компоненты (структура) DOS. Этапы загрузки DOS.
7. Работа с файловой системой в DOS.
8. Операционные оболочки. Работа с файлами и каталогами в операционной оболочке Norton Commander.
9. Программное обеспечение общего назначения. Вирусы и средства антивирусной защиты.
10. Программное обеспечение общего назначения. Архивация данных. (Winrar)

## **6 Сетевое оборудование**

### **6.1 Компьютерные сети. Виды, классификация**

С развитием компьютерной техники одновременно развивались и средства обмена информацией *между* несколькими компьютерами. В начале 70-х годов был разработан прообраз современной высокоскоростной технологии Ethernet. В то же время несколько университетов США стали обмениваться данными с помощью модемов, работающих на небольшой скорости. Эта технология положила начало системе телеконференций Usenet. Разнообразные подходы и идеи, используемые при построении сети, в то время еще не обеспечивали совместимости различных систем. Вскоре была разработана концепция объединения сетей различных компаний и организаций с помощью специально разработанных для этих целей *шлюзов (gateways)*. Шлюзы используют общий протокол для обмена данными, получивший название *Internet Protocol (IP)*.

Протокол IP стал использоваться в качестве стандарта и для локальных сетей. Все большая стандартизация и расширение предоставляемых возможностей способствовали тому, что количество компьютеров, имеющих доступ к этим услугам, неуклонно росло. Со временем пользователи стали воспринимать такое свободное объединение компьютеров как нечто единое - **Internet**. Далее мы будем использовать русскую транскрипцию этого

термина

**Интернет.**

Так что же это такое Интернет? Это - совокупность государственных, региональных, корпоративных и других компьютерных сетей, а также отдельных компьютеров, объединенных между собой разнообразными каналами передачи данных и унификацией применяемых технологий. На сегодняшний день сеть Интернет охватывает практически все страны мира, ее услугами пользуется уже более 50 миллионов человек, и их число продолжает неуклонно расти.

Интернет часто называют "Сетью сетей". По своей структуре это полностью децентрализованная сеть, состоящая из множества других сетей. В Интернет нет никакого единого центра управления или единого руководства. С самого начала в структуру будущей сети Интернет были заложены такие качества, как надежность передачи информации и высокая отказоустойчивость.

Три ключевых понятия составляют основу Интернет - узловой компьютер, канал передачи данных и протокол TCP/IP.

*Узловой компьютер*, называемый также хостом, обеспечивает передачу информации в сеть от абонентов, подключенных к нему, и прием информации для своих абонентов. Вторая, не менее важная функция хоста - регулирование и, при необходимости, перенаправление по другому пути потоков данных от других хостов. Обычно хост-компьютеры устанавливаются в организациях, предоставляющих доступ к сети.

Если хост-компьютеры являются как бы узелками сети, то *каналы передачи данных* - это нити, связывающие воедино все такие узелки. Наиболее часто для этой цели используются обычные коммутируемые и выделенные телефонные линии. Несколько реже используются оптоволоконные и спутниковые каналы передачи данных, их широкое распространение сдерживается дороговизной установки и подключения.

И, наконец, *протокол передачи данных TCP/IP* (Transmission Control Protocol/Internet Protocol) - стандартный сетевой протокол связи, используемый для соединения компьютерных систем через Интернет, обеспечивает надежную пересылку информации в масштабах всей сети. Программное обеспечение для работы с протоколом TCP/IP устанавливается на хост-компьютерах и

машинах абонентов сети. В отличие от других протоколов передачи данных TCP/IP был разработан специально для работы в Интернет, поэтому в него изначально заложены такие необходимые качества, как гарантированная доставка информации без потерь до места назначения, изменение пути следования информации при отказе одного из сегментов сети, гибкость и расширяемость.

Современные электронно-вычислительные комплексы - это открытые системы ЭВМ, т. е. имеющие в своей основе возможность подключения различных систем,- как программного, так и аппаратного характера.

### **Компьютерные**

### **коммуникации**

*Интернет (Internet)* - глобальная сеть компьютеров, связанных между собой с помощью базового протокола, например TCP/IP. (Интранет - внутренняя (закрытая) сеть, использующая технологии Интернет - самая эффективная клиент-серверная технология) Локальные и глобальные компьютерные информационные сети.

*Локальная сеть* - сеть предприятия, организации и т.д (школьная сеть)

*Глобальная сеть* - Интернет - сеть в мировых или региональных масштабах.

Глобальные сети развиваются в рамках открытых информационных систем. (Fido, Goldnet, AT50)

*Модемы,* *каналы* *связи.*

*Канал связи (КС)* - техническое средство для передачи сигналов между устройствами, находящимися на расстоянии друг от друга. Информация, передаваемая с помощью одиночных или последовательных сигналов, называется сообщением. КС состоит из трех главных частей: передатчика (модуляция), приемника (демодуляция) и линии связи (физическая среда).

*Затухание сигнала* - рассеивание частиц сигнала.

*Пропускная способность (быстродействие канала)* - количество бит передаваемых в единицу времени (бит/с, бод) Каналы связи: коммутируемые (назначаемые в момент набора номера) и выделенные (закрепленные) Модем - средство межкомпьютерного соединения посредством телефонных каналов связи (компьютер может связываться с помощью модема с факсом).

## 6.2 Сетевое программное обеспечение и сетевой протокол

Функционирование аппаратной части сети должно быть поддержано соответствующими программами. Сетевые программы позволяют определять адреса компьютеров, делать доступными программные и аппаратные ресурсы для клиентов сети, назначать различные права доступа пользователям, защищать информацию.

Эти программы входят в состав сетевых операционных систем, к которым относятся Windows 95, Windows 98, Windows NT, Windows 2000, Windows XP, NetWare и UNIX, причем первые две из перечисленных систем позволяют организовать только одноранговую сеть.

Сетевое программное обеспечение можно разделить на два вида: *программы-серверы*, которые размещаются на сервере сети, и *программы-клиенты*, размещенные на компьютере пользователя и пользующиеся услугами сервера. Сетевые протоколы, о которых речь пойдет ниже, являются частью сетевого программного обеспечения.

Если вы - пользователь локальной сети, как практически будет организовано ваше пребывание в этой сети? Эти условия определяются топологией сети, используемой операционной системой и *администратором вашей сети*. Администратор сети (системный администратор) - человек, отвечающий за организацию и работу сети.

Допустим, вы собрались работать в сети с выделенным сервером. Администратор добавит вас как нового пользователя сети. Это означает следующее:

- вам будет присвоен *логин* - уникальное имя пользователя;
- вы заведете себе *пароль* - секретную последовательность символов, подтверждающую, что именно вы являетесь владельцем этого логина;
- администратор определит ваши права доступа к информационным и аппаратным ресурсам сети.

В дальнейшем при входе в систему вы каждый раз будете вводить для *авторизации доступа* свои логин и пароль. Авторизацией

называют процесс проверки имеющихся у пользователя прав и разрешений на доступ к ресурсам в домене (сети).

Вам может быть выделено место на диске сервера для хранения ваших личных файлов и определено место, где хранятся программы общего пользования, которые вы сможете запускать со своего компьютера. Вам могут быть доступны сетевой принтер, сетевой CD-ROM и другие устройства в сети.

Добраться до доступных вам ресурсов поможет папка Сетевое окружение на Рабочем столе. В ней вы найдете имена всех компьютеров сети. Зная заранее, на каком компьютере находится необходимое вам обеспечение, вы быстрее доберетесь до нужного ресурса. Иначе вам придется отыскивать его путем перебора самостоятельно. Администратор сети может автоматически внести ссылки на доступные вам ресурсы в папку Мой компьютер на вашем Рабочем столе. Это значительно облегчит поиски.

Для вас выделяется также уникальное рабочее пространство на локальном компьютере, с которого вы заходите в сеть. Сюда относятся настраиваемые и изменяемые объекты Windows (Главное меню, Рабочий стол, папки Избранное и Мои документы, и др.). Кроме администратора и вас никто не может проникнуть в это пространство, что обеспечивает защиту информации от чужого вмешательства.

Для передачи по сети файл разбивается на части - *пакеты*. Каждый пакет снабжен служебным уведомлением и передается независимо от остальных пакетов. На конечном пункте в компьютере все пакеты собираются в один файл. Так как пакеты передаются независимо, то каждый пакет может дойти до конечного компьютера по своему пути.

Каждый компьютер в сети имеет свой *уникальный адрес*. Правила адресации в сети должны быть *одинаковыми*, хотя компьютеры и операционные системы, входящие в сеть, могут быть разнородными. Пакеты данных перемещаются по сети к компьютеру с нужным адресом. Вначале проверяется кратчайший путь, если он занят или разрушен, то проверяется следующий наиболее короткий путь и т.д. На конечном компьютере проверяется наличие всех пакетов, составляющих файл. Если какого-либо пакета не хватает, компьютер-адресат делает запрос компьютеру-отправителю и

сообщает, какой пакет отсутствует. Нужный пакет заново посылается адресату. Все правила кодирования и пересылки файлов записываются в *сетевом протоколе*.

**Сетевой протокол** или **протокол обмена** - это общее соглашение, определяющее единые правила передачи информации в сети.

Протокол определяет тип используемых данных, стандарты связи, правила обработки ошибок. Существует множество сетевых протоколов. Протокол, позволяющий делить файлы на пакеты и передавать пакеты от узла к узлу, называется IP (Internet Protocol). Для объединения сетей, работающих по протоколу IP и сетей, работающих по другим протоколам, был создан специальный *межсетевой протокол*, и назван он был TCP (Transmission Control Protocol) - протокол управления передачей. Протокол TCP обеспечивает устойчивое соединение между компьютером-отправителем и компьютером-адресатом и отвечает за разбиение передаваемых данных на пакеты, за сбор отдельных пакетов в форму исходных данных, за досылку потерянных пакетов.

Так как протоколы IP и TCP работают совместно, их объединение называют *протоколом TCP/IP*.

Протокол - система соглашений, поддерживаемых программным обеспечением и оборудованием (периферией) ЭВМ.

Протокол TCP/IP (Transmission Control Protocol/Internet Protocol) - сетевой протокол, позволяющий компьютерам осуществлять соединение по внутренним сетям или через Интернет. Каждый компьютер в Интернет использует TCP/IP.

IP-адрес (*айпи-адрес*, сокращение от [англ. Internet Protocol Address](#)) — уникальный [сетевой адрес](#) узла в [компьютерной сети](#), построенной по [протоколу IP](#). При связи через сеть [Интернет](#) требуется глобальная уникальность адреса, в случае работы в [локальной сети](#) требуется уникальность адреса в пределах сети. Имеет длину 4 байта. Обычно первый и второй байты определяют адрес сети, третий байт определяет адрес подсети, а четвертый - адрес компьютера в подсети

### 6.3 Локальные компьютерные сети

Принципы функционирования различных электронных сетей примерно одинаковы. Все они представляют собой систему, складывающуюся из компьютеров, каналов связи и некоторого соглашения, которое позволяет компьютеру-адресату преобразовывать воспринимаемый сигнал.

Каждый компьютер в составе ЛВС должен иметь следующие компоненты:

- *сетевой адаптер* (сетевую карту);
- *канал связи* (линия связи);
- *сетевую операционную систему* (сетевые программы).

*Сетевой адаптер* - устройство, отвечающее за сопряжение компьютера и канала связи; он принимает и передает сигналы, распространяемые по каналу.

Адаптер вставляют в гнездо материнской платы одного компьютера и соединяют с сетевым адаптером другого компьютера. Каждому компьютеру назначается свой адрес в сети, который фиксируется на сетевой карте. В соответствии с этим адресом компьютер из всей информации, передаваемой по сети, выбирает предназначенную именно для него.

Сетевой кабель обеспечивает канал связи компьютера с остальными машинами сети. Используют различные виды сетевых кабелей. Рассмотрим их свойства.

*Витая пара.* Кабель содержит две или более пары проводов, скрученных один с другим по всей длине кабеля. Скручивание позволяет повысить помехоустойчивость кабеля и снизить влияние каждой пары на все остальные. Различают неэкранированные и экранированные витые пары. Максимальное расстояние, на котором могут быть расположены компьютеры, соединенные неэкранированной витой парой, достигает 300 м. Скорость передачи информации - от 10 до 155 Мбит/с. Экранированная витая пара обладает лучшей по сравнению с неэкранированной помехозащищенностью, и скорость передачи по этому кабелю - 16 Мбит/с на расстоянии до 90 м.

*Коаксиальный кабель.* Состоит из центрального проводника (сплошного или многожильного), покрытого слоем полимерного изолятора, поверх которого расположен другой проводник (экран). Экран представляет собой оплетку из медного провода вокруг изолятора или обернутую вокруг изолятора фольгу. В высококачественных кабелях присутствуют и оплетка и фольга. Коаксиальный кабель обеспечивает более высокую помехоустойчивость по сравнению с витой парой, позволяет передавать информацию на расстояние до 2000 м со скоростью до 44 Мбит/с, но он дороже, и возникают проблемы с наращиванием кабеля.

*Оптический кабель.* Состоит из одного или нескольких кварцевых волокон (иногда полимерных), покрытых защитной оболочкой. Оболочка, как правило, состоит из нескольких слоев для обеспечения лучшей защиты волокон. Позволяет передавать информацию на дальнее расстояние со скоростью до 10 Гбит/с.

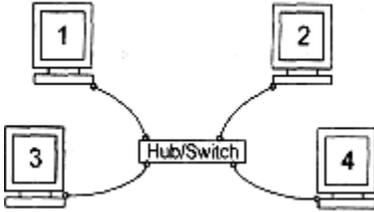
**Топология** - это конфигурация локальной сети, которая описывает схему физического соединения компьютеров, тип оборудования, методы управления обменом, надежность работы и возможность расширения сети.

Выделяют два способа соединения - *последовательное* и *соединение звездой*. При последовательном соединении компьютеры могут быть соединены кольцом или общей шиной.

*Соединение кольцом.* При этом соединении данные передаются последовательно от компьютера к компьютеру, причем, в двух направлениях, что повышает устойчивость к неполадкам сети. Один разрыв не выводит сеть из строя, но два разрыва делают сеть нерабочей. Кольцевая сеть достаточно широко применяется из-за высокой скорости передачи данных и надежности. Однако стоимость такой сети достаточно высока за счет расходов на адаптеры, кабели и дополнительные приспособления.

*Соединение по общей шине.* При таком соединении обмен может производиться непосредственно между любыми компьютерами сети, независимо от остальных. При повреждении связи одного компьютера с общей шиной этот компьютер отключается от сети, но вся сеть работает. В этом смысле сеть достаточно устойчива, но если повреждается шина, то вся сеть выходит из строя.

*Соединение звездой.* В данной топологии каждый компьютер подключается к специальному *концентратору (коммутатору, хабу)*. Такая сеть очень устойчива к повреждениям. При повреждении одного из соединений от сети отключается только один компьютер. Кроме того, эта схема соединения позволяет создавать сложные разветвленные сети.



Все указанные схемы могут, в свою очередь, быть *одноранговыми* или с *выделенным сервером*, в зависимости от способа организации.

В *одноранговой* сети все компьютеры равноправны. С каждого компьютера есть доступ на все остальные компьютеры сети.

Сеть с выделенным сервером имеет центральный компьютер - *сервер*, который управляет работой сети. Это наиболее мощный компьютер, с большим объемом оперативной и дисковой памяти. Сервер распределяет доступ пользователей к компьютерам сети и к общим сетевым ресурсам. Общим является жесткий диск сервера, на нем находятся программы, которые могут все запускать, и через него пользователи обмениваются информацией. Общие аппаратные ресурсы могут быть распределены по сети. Например, один компьютер в сети имеет CD-ROM, а другой - принтер, а на третьем установлен модем. Сеть делает доступными все эти устройства для совместного использования.

*Сервер* - это главный компьютер сети, предоставляющий доступ к общей базе данных, совместное использование устройств ввода-вывода, обеспечение взаимодействия пользователей.

Остальные компьютеры называются *рабочими станциями* или *клиентами*. Каждому клиенту выделяется регистрационное имя и пароль.

*Клиент* - компьютер в сети, имеющий доступ к информационным

ресурсам и устройствам сервера.

Клиенты, в свою очередь, могут быть серверами для других компьютеров. Например, одна и та же машина, будучи клиентом главного сервера сети, может предоставлять свои информационные ресурсы (файлы, программы, тексты) другим компьютерам.

#### 6.4 Глобальные компьютерные сети

В отличие от локальных сетей в глобальных сетях нет какого-либо единого центра управления. Глобальная сеть строится на основе нескольких мощных компьютеров-серверов, соединенных между собой. К этим серверам обычно подключены региональные серверы со своими сетями, корпоративные и локальные сети. А к локальным сетям - пользователи отдельных компьютеров. Хотя для подключения отдельного компьютера к глобальной сети вовсе не обязательно подключать его к сетям промежуточных уровней.

Подключиться к глобальной сети можно, выбрав один из способов:

- *удаленный* доступ по коммутируемой телефонной линии;
- *прямой* доступ по выделенному каналу.

*Удаленный* доступ по коммутируемой телефонной линии. В этом случае в распоряжении пользователя вместо сетевого адаптера должен быть *модем*. Модем подключает компьютер к телефонной линии. Он предназначен для преобразования (модуляции) цифровых сигналов на выходе компьютера в аналоговые сигналы, которые могут передаваться по телефонной сети, и обратном преобразовании (демодуляции) при приеме информации на компьютер. На другом конце телефонной линии также должен быть подключен модем.

*Прямой* доступ по выделенному каналу. Этот, более дорогой способ подключения, чаще используют те или иные организации. В качестве выделенных каналов могут быть коаксиальные и оптоволоконные кабели, радиорелейные линии, спутниковая связь. По видам каналов различают и модемы: радиомодемы, телефонные модемы, волоконно-оптические модемы и др.

Скорость, с которой будет производиться обмен информацией,

определяется *скоростью передачи данных модема и пропускной способностью* линии связи.

Скорость передачи данных модема определяет, какое количество информации (бит) модем может передавать или принимать за единицу времени (секунду). Таким образом, единица измерения скорости передачи данных - *бит/с* или *бод*.

Чем выше скорость передачи данных, тем меньше времени потребуется модему на передачу или прием информации. Например, при пересылке файла размером 500 Кб по модему со скоростью передачи данных 2400 бит/с понадобится около 28 минут, модему на 9600 бит/с - около 7 минут, модему на 14 400 бит/с - около 5 минут.

На скорость передачи данных влияет также *выбор линий связи*. Так, при использовании коммутируемых, обычных телефонных линий, когда модем подключен параллельно к телефонному аппарату, скорость передачи данных составляет всего несколько тысяч бод.

При выделенных линиях связи скорость передачи данных увеличивается до нескольких десятков тысяч бод. Такие линии связи постоянно подключены и выгодны при передаче больших объемов информации или срочной передаче данных. Например, система резервирования и продажи авиабилетов действует при выделенных линиях связи.

Применение спутниковой связи и высокоскоростных оптоволоконных линий связи повышает пропускную способность каналов до сотен миллионов килобод. Благодаря таким каналам, оказывается возможным объединение всех компьютерных сетей в глобальные сети, связывающие пользователей независимо от их географического расположения.

## **6.5 Интернет. Сервисы сети Интернет**

*Электронная почта*, или *e-mail*, является аналогом обычной бумажной почты. Электронный адрес позволяет совершенно однозначно идентифицировать пользователя этой услуги среди миллионов других пользователей сети. С помощью специальных программ для пересылки электронной почты, зная адрес другого человека, можно отправить ему текстовое сообщение, программы,

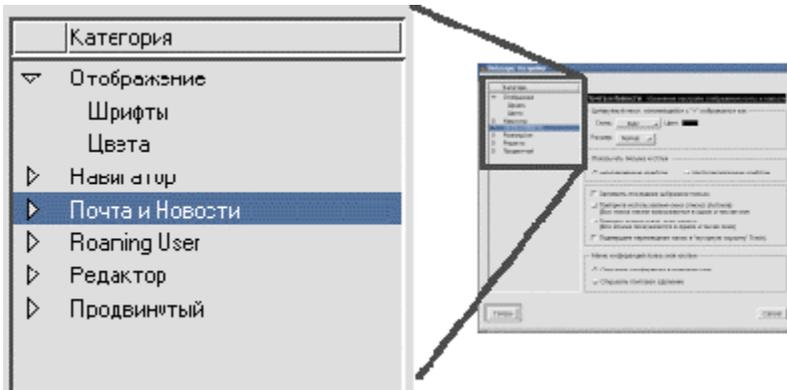
изображения, словом, любую информацию, способную храниться в электронном виде на компьютере, и она будет доставлена через несколько минут даже на другой конец Земли. Точно так же любой другой человек, зная ваш электронный адрес, сможет отправить электронное послание вам.

*Электронная почта* (E-mail) представляет собой сервис, предназначенный для пересылки сообщений между пользователями Интернета и локальных сетей. Основная идея состоит в следующем: каждый пользователь имеет уникальный почтовый адрес, как правило образуемый из его регистрационного (входного) имени и имени сервера, где он зарегистрирован. Таким образом, сохраняется некоторая аналогия с бумажной почтой, где адрес состоит из двух частей: "Куда" и "Кому". "Куда" - на сервер, "Кому" - пользователю. Имя пользователя и имя сервера разделяются символом "@" (называемым "коммерческое ат", а русскоязычные пользователи часто используют термин "собачка"). Никакие пробелы в адресе не допускаются. Адрес зарегистрированного в компьютерной сети МГИУ пользователя имеет вид: logname@mail.msiu.ru, где logname заменяется на входное имя пользователя.

Для работы с электронной почтой существует много различных программ. Далее рассматривается вопрос отправки и приема почты с помощью программы Netscape Messenger, входящей в состав браузера Netscape.

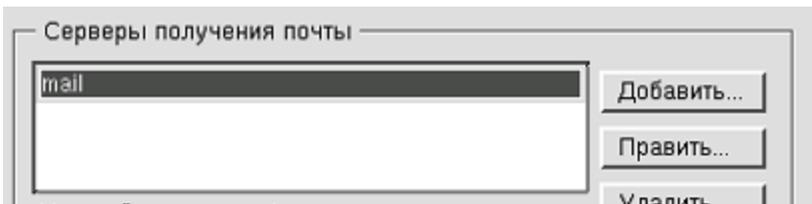
### **Настройка почты в Netscape**

Для того чтобы Netscape смог работать с электронной почтой, нужно его настроить, сообщив ему необходимую информацию. Для этого выберите пункт **Правка** (Edit) в верхнем меню браузера Netscape, и в появившемся списке выберите пункт **Настройка** (Preferences)



После этого на экране появится окно настроек. Оно состоит из двух главных частей: слева расположены категории настроек, а справа - сами настройки, их содержимое.

Обратите внимание, категории бывают двух видов: есть собственно категории и подкатегории, т. е. категории, входящие в состав первых. У названия категорий первого уровня слева есть такой треугольничек: "▶". Если он направлен вправо (как на картинке), категория не раскрыта на подкатегории, а если он направлен вниз; - раскрыта. Для того чтобы перевести треугольничек из одного состояния в другое, нажмите на него мышью. На рисунке категория **Отображение** (Appearance) раскрыта, а категории **Навигатор** (Navigator), **Roaming User**, **Почта и Новости** (Mail & Newsgroups), **Редактор** (Composer) и **Продвинутый** (Advanced) - нет.



Для настройки почты в Netscape следует выполнить следующие действия.

1. Раскрыть категорию **Почта и Новости** (Mail & Newsgroups).
2. Выбрать подкатегорию **Идентификация** (Identify); эта подкатегория содержит информацию о пользователе; необходимо вписать свой почтовый адрес в поле **Адрес e-mail** (Email address). Если там уже указано ваше входное имя, то нужно будет дописать к нему символы **@mail.msiu.ru**. Содержимое остальных полей изменять не обязательно.
3. Выбрать подкатегорию **Почтовый сервер** (Mail servers).  
В появившемся окне в верхнем поле следует проверить имя почтового сервера и, если оно отличается от **mail**, то выделить его и нажать кнопку **Править...** (Edit) справа. После этого появится окно, в котором надо будет вписать в поле **Имя сервера** (Server Name) название сервера **mail**, а в поле **Ваше имя** (User Name) - свое входное имя. Для того чтобы пароль не запрашивался каждый раз при попытке получить почту, установите флажок **Запомнить пароль** (Remember password). После это надо нажать кнопку **Готово** (OK), чтобы опять появилось окно основных настроек.
4. Продолжая работать с категорией Почтовый сервер, введите в поле Сервер для отправки почты: (Outgoing mail (SMTP) server) название **mail**.  
После этого настройку почты можно считать законченной. Для выхода из окна настроек нажмите кнопку **Готово**.

### **Отправка почты**

Для входа в режим работы с почтой нажмите либо кнопку с изображением письма, находящуюся в правом нижнем углу браузера Netscape, либо выберите пункт **Почтовый ящик** (Messenger) из меню **Коммуникатор** (Communicator).

Не забудьте установить правильную кодировку - **Cyrillic (KOI8-R)**. Для того чтобы написать письмо, нужно нажать на кнопку **Написать** (New Msg) вверху окна. Появится окно с формой для отправки почты. В нем в верхнем поле **То** вводится электронный адрес получателя.

Среднее поле **Тема** (Subject) предназначено для записи предмета письма, его темы или названия. Это нужно для того, чтобы получатель мог сразу понять, о чем письмо, либо быстро найти давно полученное письмо среди других.

Сам текст письма пишется в большом поле внизу. При написании письма можно использовать форматирование текста: выделение цветом, размером и начертанием шрифта и т. д. Для создания текста автоматически подключается программа Netscape Composer, позволяющая включать в письма все элементы, доступные в гипертекстовом документе: ссылки, таблицы, графику и т. д. Эта программа подробно рассматривается в разделе "Обработка текста на ЭВМ".

Для отправки написанного письма необходимо нажать кнопку **Отослать** (Send). Если письмо содержало элементы форматирования, то появится окно, в котором нужно будет также нажать **Send**.

Иногда требуется послать одно и то же письмо нескольким людям. Для этого в поле ввода электронного адреса ввести сразу несколько адресов, разделяя их запятой или нажимая клавишу Enter после каждого адреса.

### Получение

### почты

Проверка того, пришла ли почта, осуществляется нажатием кнопки **Получить** (Get Msg). Если новых сообщений нет, в самом низу окна появится сообщение об этом (There are no new messages on the server). Если же имеются новые сообщения, то покажется окошко загрузки почты и в списке сообщений справа появится новое, отмеченное жирным шрифтом. Чтобы прочитать полученное сообщение, нажмите на него мышкой.

Накапливающиеся сообщения сортируются по разным полям: по предмету письма (**Subject**), имени адресата (**Sender**), дате отправки (**Date**) и приоритету (**Priority**). Щелчок левой кнопкой мыши по наименованию поля сортирует почту по данному признаку (в порядке возрастания), повторный щелчок изменяет порядок сортировки на обратный.

### Ответ

### на

### послание

Для того чтобы ответить на пришедшее письмо, нужно нажать на кнопку **Ответить** (Reply). Появится окно посылки почты, в котором уже заполнены поля адреса и предмета письма, а также процитировано письмо, на которое вы пишете ответ. Цитату при

желании можно удалить (полностью или частично).

Элементы	почтового			ящика
Почтовый ящик	состоит	из	нескольких	частей:
1. <b>Inbox</b>	-	пришедшие	письма;	
2. <b>Unsent Messages</b>	-	неотосланные	письма;	
3. <b>Drafts</b>	-	недописанные	письма (черновики);	
4. <b>Templates</b>	-		шаблоны;	
5. <b>Sent</b>	-	отосланные	письма;	
6. <b>Trash</b>	-	удаленные	письма (корзина).	

Переключаться между этими элементами можно путем выбора их слева в окне или в раскрывающемся списке вверху. Письма можно переносить с помощью мыши из одной части почтового ящика в другую. Удалить письмо можно, если нажать на него мышью и выбрать кнопку **Удалить** (Delete), либо нажать комбинацию клавиш Alt+d.

### **Адресная книга**

Для удобства работы с почтой в Netscape имеется адресная книга - аналог телефонной книги. В адресную книгу можно записывать адреса ваших корреспондентов, чтобы не вспоминать и не вводить адрес при каждой отправке почты.

Для входа в адресную книгу выберите пункт **Адресная книга** (Address Book) из меню **Коммуникатор** (Communicator). Для добавления новой записи в появившемся окне выберите пункт **Новая карточка** (New Card) из меню **Файлы** (File), либо нажмите кнопку Новая карта (New Card). В появившейся форме на вкладке Name вводится следующая информация об адресате:

- **First Name** - имя адресата;
- **Last Name** - фамилия адресата;
- **Display Name** - полное имя адресата (автоматически добавляется к письму);
- **Email Address** - электронный адрес.

Поскольку адресную книгу можно использовать не только как список электронных адресов, но и как записную книгу, то

в ней предусмотрены поля для хранения дополнительной информации об адресатах:

- **Nickname** - псевдоним;
- **Title** - заголовок или название;
- **Organization** - организация, которую представляет или к которой имеет отношение адресат;
- **Department** - отдел в этой организации;
- **Notes** - некоторые дополнительные заметки;
- **Prefers to receive HTML** - предпочитает получать не простой текст, а оформленный средствами HTML.

Ввод информации и добавление новой записи в адресную книгу осуществляется при нажатии **Готово** (OK).

Чтобы послать сообщение человеку, адрес которого хранится в адресной книге, нужно либо, выбрав его адрес, мышкой нажать кнопку **Написать** (New Msg), либо при открытом окне создания нового письма перетащить мышкой адрес из адресной книги в поле **То** окна создания письма, либо в окне создания нового письма нажать кнопку **Адрес** (Address), после чего появится окно с адресами, в котором, после выбора адресата, нажмите кнопку **То**.

### **Присоединение файлов (Attach)**

Вместе с письмом можно пересылать различные файлы, например, картинки, архивы, музыкальные файлы, программы и т. д. Получается как бы уже не письмо, а посылка. Для того чтобы присоединить к письму файл, нужно нажать в окне создания письма кнопку **Прицепить** (Attach) и в появившемся списке выбрать **Файл** (File), если подключается файл общего вида, и **Web-страница** (Web Page), если подключается гипертекстовый документ. После этого появляется окно поиска файла, в котором и надо выбрать требуемый файл. Можно присоединять несколько разных файлов, нажимая на **Прицепить** требуемое число раз.

### **Переотправка почты (Forward)**

Если полученной вами почтой хочется поделиться еще с кем-то, можно это письмо переслать по другому адресу. Для этого надо

нажать в окне чтения почты кнопку **Переслать** (Forward) и вписать новый адрес.

*Телеконференции*, или как их еще называют, группы новостей, похожи на электронную почту. Различие состоит в том, что телеконференции - это как бы огромный, непрерывно обновляющийся электронный журнал, разбитый на множество разделов по интересам, на которые можно подписаться и получать сообщения только из интересующих вас разделов, а неинтересные попросту игнорировать. Вы можете не только читать сообщения, поступающие в телеконференции, но и посылать туда свои вопросы, предложения и высказывать мнения, которые прочтут все люди, подписанные на эти разделы.

Ну и наконец *World-Wide-Web* (WWW). WWW - это гипертекстовая среда, содержащая огромное количество различных документов, таких как информационно-справочные базы данных, правительственные документы, каталоги библиотек и многое, многое другое. Вы можете просматривать такие документы в реальном времени, переходя от одного документа к другому простым нажатием на кнопку мыши, наведенной на нужную вам ссылку. Основное достоинство WWW состоит в том, что документы могут иметь ссылки не только в пределах одной базы или компьютера, но могут ссылаться и на другие документы, хранящиеся на удаленном компьютере. В результате мы получаем как бы единое гипертекстовое пространство, по которому можно перемещаться в любом необходимом направлении в поисках нужной информации.

Электронная почта и доски объявлений - самый дешевый вид межрегиональной связи - это система межкомпьютерной связи, при которой один ПК, используя специальный протокол, оставляет сообщение на специальном компьютере (сервере), физическое местонахождение которого не имеет значения, - если данное сообщение ориентировано на использование одним пользователем - это электронная почта; если же сообщение предназначено широкому кругу пользователей - это доска объявлений.

Система телеконференций и чат-серверов - это система межкомпьютерной связи в реальном времени.

*World Wide Web* ("Всемирная паутина") - система доступа к данным, используемая в Интернет. Пользователь получает доступ к

страницам (*pages*) информации, содержащим текст, графику и ссылки на другие страницы информации. Графические программы просмотра (*graphical browser programs*, **браузеры**) позволяют переходить на другую страницу, содержащую нужную вам информацию, при помощи щелчка мыши на ссылке.

Система WWW состоит из большого числа программ-серверов, выполняющихся на машинах сети Интернет. Совместно серверы WWW образуют распределенную базу данных сетевого гипертекста. Сервер наполняется информацией на какую-либо тему, включая образы фотографий и картин, музыки, шумов и речи. Далее пользователь через сеть Интернет с помощью одной из программ просмотра у себя на компьютере обращается к этому серверу по его адресу в сети. Пользователь видит текст, в котором, выделены некоторые участки текста - "ссылки". Достаточно щелкнуть по такому участку мышкой и раскроется содержание нового документа. При этом ссылка может практически мгновенно (за несколько секунд) привести пользователя на сервер WWW, установленный на другом конце света.

Технология World Wide Web базируется на трех важных стандартах. Первый из них - **URL** (*Universal, или Uniform Resource Locator, универсальный адрес ресурса*) - предоставляет стандартный способ задания местоположения данных, доступных в глобальной компьютерной сети Интернет.

Второй - протокол **HTTP** (*Hyper Text Transfer Protocol, протокол передачи гипертекста*) - предоставляет доступ к информации и позволяет передавать гипертекстовые документы по сети.

Наконец, **HTML** (*Hyper-Text Markup Language, язык разметки гипертекста*) позволяет создавать текстовые документы, включающие ссылки на URL других данных. Зачастую эти ссылки указывают на другие документы HTML, которые, в свою очередь, доступны с помощью HTTP. В результате перед пользователем расстилается огромная паутина взаимосвязанной информации.

Следует отметить, что HTML не позволяет точно задавать внешний вид документа. Можно лишь предлагать свой вариант оформления. Различные программы просмотра могут интерпретировать ваши предложения по-своему. Авторы, ориентирующиеся на возможности только одной из программ

просмотра, ограничивают круг возможной аудитории.

**URL** - специальная форма адреса информации в сети Интернет, содержащая данные об имени сервера, на котором хранится документ, путь к каталогу файла и собственно имя файла. URL-адрес состоит из двух частей. Сначала указывается *способ связи*, при помощи которого будет осуществляться доступ к данным. От этого зависит, какая дополнительная информация потребуется. Затем помещается информация о том, где эти данные *расположены*. Разделяются эти части двоеточием, например:

`http://имя_сервера/путь/файл`

Рассмотрим наиболее распространенные способы доступа к данным в сети Интернет.

**HTTP** (протокол передачи гипертекста) был разработан специально для World Wide Web. При использовании этого протокола необходимо указать имя машины, а также сообщить дополнительную информацию, которую компьютер сможет использовать для поиска и создания необходимых данных. Эти дополнительные данные обычно представляют собой имя файла и информацию о каталоге. Отчасти из-за того, что первые разработки были созданы на системах Unix, для разделения имен каталогов и файла в URL используется прямая наклонная черта "/". Например, "`http://www.ctc.msiu.ru`".

**FTP** (*File Transfer Protocol*, протокол передачи файлов) - давно используемый метод доступа, разработанный для передачи больших объемов информации по Интернет. Этот метод получил широкое распространение уже достаточно давно. Чтобы получить доступ к файлу или каталогу FTP, необходимо указать имя машины и имя файла или каталога этой машины. Например, "`ftp://prep.ai.mit.edu`".

Появление в ноябре 1996 года сервиса ICQ (можно прочесть как *I Seek You*, то есть "Я ищу тебя", а русскоязычные пользователи ласково окрестили эту программу "аськой"), предоставило еще одну, поистине революционную, возможность общения пользователям Интернет. Название отражает самое важное отличие этой программы от большинства других программ общения в Интернете. ICQ на самом деле позволяет "видеть" всех ваших друзей и знакомых, как только они появляются в сети Интернет в режиме онлайн. ICQ предоставляет возможность, забыв о расстояниях, мгновенно



*Выбор**состояния*

Пользователь имеет возможность выбрать для себя желаемое состояние (в сети, отошел, недоступен, занят, не беспокоить, свободен для разговора, отключен, режим полной невидимости).

*ChatDirect*

Обеспечивает диалог (чат) с неограниченным количеством избранных ICQ-пользователей.

*MessageDirect*

Позволяет обмениваться сообщениями, а также передавать сообщение всем, кто в этот момент не доступен. Все сообщения, поступившие, когда пользователь был недоступен, сохраняются до тех пор, пока он не получит возможность их прочесть. Сохраняется полный диалог с каждым пользователем.

*Запись**диалога*

Все входящие и исходящие сообщения, а также прочие события автоматически сохраняются. Диалоги с каждым могут быть восстановлены в связной форме, даже если в это время вы разговаривали с десятком различных ICQ-пользователей. На протяжении, а также в конце чата, пользователь может сохранить беседу в текстовом файле.

*Конфиденциальность*

Находясь в активном режиме, пользователь может "скрыться", позволяя видеть себя только некоторым, или даже сохранять полную анонимность.

*Секретность*

Система предоставляет возможность защитить доступ в ICQ личным паролем.

*"Черный**список"*

Формирование списка людей, сообщения которых будут автоматически игнорироваться.

*E-mail-совместимость*

Система поддерживает все приложения электронной почты и позволяет быстро проверять содержание почтового ящика.

*UIN-номер*

Каждому присваивается персональный номер, который идентифицирует пользователя, и может быть указан в любых документах для установления мгновенных контактов.

## 6.6 Поиск информации в Интернет

Современная Сеть действительно в состоянии предложить своему пользователю массу информации самого разного профиля. Здесь можно познакомиться с новостями, интересно провести время, получить доступ к разнообразной справочной, энциклопедической и учебной информации. Интернет можно эффективно использовать для решения самых разных задач на работе и дома.

Самая главная проблема, возникающая при работе с Сетью, - быстро найти нужную информацию и разобраться в ней, оценить информационную ценность того или иного ресурса для своих целей.

Путь к огромному информационному багажу человечества, хранящемуся в библиотеках, фонотеках, фильмотеках, лежит через карточки каталогов. В Интернете существуют аналогичные механизмы для нахождения требуемой информации. Речь идет о поисковых серверах, служащих отправной точкой для пользователей Сети. С содержательной точки зрения о них можно говорить как о специальной службе Интернета, хотя они используют механизмы Всемирной Паутины и с технической точки зрения не выходят за ее рамки.

Поисковые сервера достаточно многочисленны и разнообразны. Принято различать поисковые индексы и каталоги. Сервера-индексы работают следующим образом: регулярно прочитывают содержание большинства веб-страниц Сети ("индексируют" их), и помещают их полностью или частично в общую базу данных. Пользователи поискового сервера имеют возможность осуществлять полнотекстовый поиск по этой базе данных, используя ключевые слова, относящиеся к интересующей их теме. Выдача результатов поиска обычно состоит из выдержек рекомендуемых вниманию пользователя страниц и их адресов (URL), оформленных в виде гиперссылок. Работать с поисковыми серверами этого типа удобно, когда хорошо представляешь себе, что именно хочешь найти.

Каталоги выросли из списков интересных ссылок, закладок (bookmarks). По сути дела они представляют собой многоуровневую смысловую классификацию ссылок, построенную по принципу "от общего к частному". Иногда ссылки сопровождаются кратким описанием ресурса. Как правило, возможен поиск в названиях

рубрик (категориях) и описаниях ресурсов по ключевым словам. Каталогами пользуются тогда, когда не вполне четко знают, что именно ищут. Переходя от самых общих категорий к более частным, можно определить, с каким именно ресурсом Сети следует ознакомиться. Поисковые каталоги уместно сравнивать с тематическими библиотечными каталогами, словарями-тезаурусами или биологическими классификациями животных и растений. Ведение поисковых каталогов частично автоматизировано, но до сих пор классификация ресурсов осуществляется главным образом вручную.

Поисковые каталоги бывают общего назначения и специализированные. Поисковые каталоги общего назначения включают в себя ресурсы самого разного профиля. Специализированные каталоги объединяют только ресурсы, посвященные определенной тематике. Им часто удается достичь лучшего охвата ресурсов из своей области и построить более адекватную рубрикацию.

История поисковых служб начинается в середине 90-х годов. В 1994 году два аспиранта Стэндфордского Университета, Дэвид Фило и Джерри Янг, начали работу над созданием каталога Yahoo (<http://www.yahoo.com/>; англ. "yahoo" - "йеху" из "Путешествий Гулливера" Джонатана Свифта). Они преследовали простую цель: организовать собственные ссылки. Чтобы сделать это эффективным образом, им пришлось построить специальную систему с использованием базы данных. Она могла одновременно выдерживать обращения тысяч пользователей. И эти обращения не замедлили последовать. В начале 1995 года Марк Андрессен, один из сооснователей корпорации Netscape Communications, предложил создателям уже завоевавшего огромную популярность Yahoo перенести систему с кампуса Стэндфордского Университета на сервера Netscape. Огромная нагрузка с университетской сети была снята, а Yahoo стал коммерческим проектом. Сегодня его создатели - мультимиллионеры; над ведением каталога трудятся тысячи специалистов в самых разных областях знания.

Один из первых индексирующих поисковых серверов, AltaVista ("вид с высоты") корпорации Digital (теперь Compaq), появился в 1995 году. Нескольким служащим корпорации-производителя

суперкомпьютеров пришла в голову мысль использовать новейшие сервера для считывания содержимого Всемирной Паутины в базу данных и осуществления поиска по ней. Сервер AltaVista, расположенный по адресу <http://altavista.digital.com/>, - один из крупнейших поисковых серверов сегодняшней Сети.

В последнее время поисковые каталоги общего назначения и индексирующие поисковые сервера интенсивно интегрируются. Yahoo уже не только каталог, но и поисковый сервер. AltaVista, как и многие другие поисковые сервера, исходно предлагавшие исключительно поиск по базе данных, сегодня включает в выдачу результатов поискового запроса еще и список рубрик, соответствующих теме запроса. Поисковые технологии не стоят на месте. Традиционные индексирующие сервера искали в базе данных документы, содержащие ключевые слова из поискового запроса. При таком подходе очень сложно оценить значение и качество ресурса, выдаваемого пользователю. Альтернативный подход - искать такие веб-страницы, на которые ссылаются другие ресурсы по данной тематике. Чем больше ссылок на страницу существует в Сети - тем больше шансов, что вы ее найдете. Такой своеобразный мета-поиск осуществляет поисковый сервер Google (<http://www.google.com/>), появившийся совсем недавно, но уже отлично себя зарекомендовавший.

Кроме поисковых серверов, заносящих в свои базы веб-страницы по всей Сети, есть поисковые сервера, ориентированные более узко в географическом и языковом отношении. Так, существует много русских поисковых серверов. Их краткий список вы найдете в следующем разделе.

В мире существует огромное количество WWW серверов самого разного назначения. Без специальных средств ориентировка в этом гигантском объеме информации просто невозможна. Решают эту проблему поисковые серверы, которые хранят миллионы ссылок на разные темы и производят поиск нужных документов по запросу пользователя.

Для того чтобы облегчить поиск документов были созданы каталоги WEB-серверов и поисковые машины. В большинстве случаев каталог представляет собой тематические подборки ссылок на Web-ресурсы (медицина, политика, программирование и т. д.).

Поисковые же машины позволяют попасть на страничку, текст которой содержит заданный набор слов. Каждая поисковая машина обладает своими специфическими возможностями, достоинствами и недостатками. Следует отметить, что наполнение сети Интернет русскоязычной информацией, хотя и происходит быстрыми темпами, все еще значительно отстает от уровня англоязычной информацией. Английский язык продолжает оставаться основным языком общения пользователей Интернет.

Познакомимся с возможностями некоторых поисковых серверов.

**AltaVista** - <http://www.av.com>

Это одна из первых наиболее мощных поисковых машин. Она имеет встроенный переводчик страниц с английского языка на французский, немецкий, итальянский, испанский и т. д. и обратно. Правда, на русский язык она переводить не умеет. Схема использования стандартна: в строке для ввода пишутся ключевые слова, а затем нажимается кнопка **Search** (поиск). Далее выдается список ссылок на страницы с краткой аннотацией.

**Yahoo!** - <http://www.yahoo.com>

Это не только поисковая машина, но и крупнейший классификатор ресурсов сети. Здесь выделено несколько разделов верхнего уровня: искусство, бизнес, компьютеры, образование, развлечения, правительство и другие. Каждый из разделов помимо ссылок содержит подразделы, которые в свою очередь тоже содержат подразделы и т. д.

Искать интересующую информацию можно как с помощью строки запроса аналогично AltaVista, так и перемещаясь по разделам, но в последнем случае необходимо точно знать, к какой именно категории относится предмет поиска.

**Rambler** - <http://www.rambler.ru>

Помимо известной поисковой системы, сервер "Rambler" включает классификационный каталог "Rambler Top 1000", состоящий из категорий, которые охватывают все основные тематические направления российских Web-серверов.

В разделах списки серверов оформлены в виде таблицы. На одном экране выводится 20 ссылок. По умолчанию сортировка идет по текущей посещаемости серверов в каждой категории. Для каждого ресурса приводится дополнительная информация о числе посещений

и динамике популярности сервера. Таким образом, сервер помогает получить верное представление о популярности отечественных Web-серверов. Российская часть Интернет представлена здесь достаточно полно.

**Yandex** - <http://www.yandex.ru>

Занимает одно из ведущих мест среди русскоязычных поисковых серверов. Активно и динамично развивается.

Стратегии, применяемые для поиска информации в Интернет, постоянно совершенствуются. Так **Google (www.google.com)**, некоторое время назад считавшийся экспериментальной поисковой системой, сейчас пользуются пятнадцать миллионов человек ежемесячно. Секрет Google в удачном алгоритме отбора документов, отвечающих поисковому запросу. Предшественники Google пытались определить "ценность" документа только на основании анализа его текста: как часто и как близко к началу документа встречаются там слова из запроса, есть ли они в заголовке и т.д.

Создатели Google решили оценивать соответствие документа запросу с помощью специального числа, похожего по смыслу на индекс цитирования: чем больше ссылок на данный документ, тем это число, называемое PageRank, больше. Если же сервер не находит запрашиваемой комбинации в своем каталоге, то пользуется тематическим каталогом Yahoo. Результаты оказались столь хороши, что Google стал "народной" поисковой системой. Google постоянно совершенствует свой сервис: проиндексированы все документы в формате pdf, организован поиск картинок в Интернет ([images.google.com](http://images.google.com)).

Одним из недостатков Google можно считать отсутствие диалога с пользователем во время подбора документов. Этот недостаток стараются исправить конкуренты Google, классифицируя найденные документы. Так поисковая система **Teoma (www.teoma.com)** автоматически классифицирует найденные ссылки на документы. Результаты при выводе делятся на три части: собственно классификация, занимающая верхнюю часть окна, документы, лучше всего соответствующие запросу, размещаются слева, а справа выводятся документы, в которых много ссылок на страницы, стоящие слева.

Еще одним претендентом на роль заменителя Google является

метапоисковая система **Vivisimo** ([www.vivisimo.com](http://www.vivisimo.com)), разработанная в университете Карнеги. Как и Google в прошлом, Vivisimo - экспериментальный некоммерческий проект. Основной особенностью этой системы является представление в виде дерева (иерархическая кластеризация) документов, найденных другими поисковыми системами, такими как Altavista и **Fast** ([www.allthemes.com](http://www.allthemes.com)). В сущности Vivisimo даже не метапоисковая система, а приставка, способная кластеризовать любые данные. Алгоритм кластеризации Vivisimo присоединяется к любой поисковой системе. В окошке правее окна запросов можно вместо "Search the Web" выбрать например "Альтависту". Vivisimo перехватит ссылки, выданные "Альтавистой", отсортирует их и разместит их на ветках дерева в левой части своего окна.

### **Работа с поисковыми серверами**

Работа с поисковыми серверами не составляет большого труда. Вы заходите на свой любимый поисковый сервер, в строке запроса набираете на нужном языке ключевые слова или фразу, соответствующие ресурсу или ресурсам Сети, которые вы хотите найти. Затем нажимаете мышью на кнопку с английской надписью "Search" или русской надписью "Поиск", и через несколько секунд в рабочем окне браузера появляются результаты поиска.

Пусть вас не смущает яркая надпись на прямоугольной картинке в заголовке страницы. Это реклама, так называемый "баннер". При нажатии на баннер вы попадете на веб-сайт рекламодателя. Ниже, под баннером, идут ссылки на страницы, которые поисковый сервер рекомендует вам посетить, их адреса и краткие описания.

Yandex: образование Москвы (971618) - Microsoft Internet Explorer

Файл Правка Вид Избранное Сервис Справка

Назад Поиск Избранное

Адрес: <http://www.yandex.ru/yandsearch?pt=rad&text=%EE%E1%F0%E0%E7%EE%E2%E0%ED%E8%E5+%CC%EE%>

Почта Помощь Настроить поиск

**Yandex**  
Найдётся всё

образование Москвы

в найденном  в регионе: Москва [расширенный поиск](#)

Везде [Новости](#) [Маркет](#) [Адреса](#) [Словари](#) [Блоги](#) [Картинки](#) [Все службы...](#)

Результат поиска: страниц — 976 168, сайтов — не менее 1 703, в каталоге — 11  
Запросов за месяц: образование — 201 413, москвы — 3 446 169. [Купить эти слова.](#)

**Адреса: образование** в Москве и Московской области

1. [Департамент образования г. Москвы](#)  
© 2006, Департамент образования г. Москвы  
[www.educom.ru](http://www.educom.ru) · 9 КБ  
[Сохраненная копия](#) · [Еще с сайта](#) 2345

[Yandex Директ](#)  
[Образовательные программы](#)  
Проекты, рекомендации, новости **образования**  
[www.school-net.ru](http://www.school-net.ru)

[Информационный по образованию](#)  
Новости проектов,

Окно браузера с результатами запроса к поисковому серверу.

Обычно поисковый сервер выдает результаты поиска небольшими порциями, например, по 10 на одну страницу выдачи. Поэтому часто они занимают больше одной страницы. Тогда под списком рекомендуемых ссылок будет находиться ссылка, предлагающая перейти к следующей "порции" результатов поиска.

В идеальном случае тот ресурс, который вы ищете, поисковый сервер поместит на первую страницу результатов поиска, и вы сразу распознаете нужную ссылку по краткому описанию. Однако часто приходится просмотреть несколько ресурсов, прежде чем обнаруживается подходящий. Как правило, пользователь просматривает их в новых окнах браузера, не закрывая окно браузера с результатами поиска. Иногда поиск и просмотр найденных ресурсов ведется в одном и том же окне браузера. Если ресурс не удовлетворяет ожиданий пользователя, то пользователь возвращается к результатам поиска, используя кнопку "Назад" ("Back") в меню браузера.

Ключевые слова, составляющие поисковый запрос, обычно просто разделяются пробелами. Различные поисковые сервера по-разному интерпретируют это. Некоторые из них отбирают по такому запросу только документы, содержащие все ключевые слова, то есть воспринимают пробел в запросе как логическую связку "и". Некоторые интерпретируют пробел как логическое "или" и ищут

документы, содержащие хотя бы одно из ключевых слов. При формировании поискового запроса большинство серверов позволяют в явном виде указать логические связки, объединяющие ключевые слова, и задать некоторые другие параметры поиска. Логические связки обычно обозначаются с помощью английских слов "and", "or", "not". На разных поисковых серверах при формировании расширенного поискового запроса используется разный синтаксис. Обычно на титульной странице поискового сервера присутствует ссылка с названием, похожим на "помощь" (англ. "help"). Возможно, именно она указывает на документ, объясняющий правила формирования расширенного поискового запроса. Прочитайте такой документ на поисковом сервере, которым станете пользоваться чаще всего, и освоите все богатство возможностей, предоставляемых этим сервером. Это позволит вам составлять более точные запросы, а грамотное построение поискового запроса играет решающую роль в процессе поиска.

С первого раза удачно задать вопрос поисковому серверу получается не всегда. Если запрос короткий и в нем присутствуют только часто употребляемые слова, может быть найдено очень много документов, сотни тысяч и миллионы. Наоборот, если ваш запрос окажется слишком детализированным или в нем будут использованы очень редкие слова, вы увидите сообщение о том, что ресурсов, отвечающих вашему запросу, в базе сервера не найдено.

Постепенное сужение или расширение фокуса поиска через увеличение или уменьшение списка ключевых слов, замена неудачных поисковых терминов на более удачные помогут вам улучшить результаты поиска.

Еще один очень важный момент - это выбор подходящего для ваших задач поискового сервера. Как мы уже говорили выше, работать с индексирующими поисковыми серверами хорошо, когда ясно, что именно требуется найти. Каталогами пользуются в тех случаях, когда не вполне четко знают, что именно ищут. Когда предметная область очерчена, но что именно в ней вас интересует в данный момент, вы не вполне понимаете, очень полезным может оказаться использование специализированного поискового каталога. Приведем список некоторых наиболее известных поисковых серверов общего назначения. Все эти сервера в настоящее время

предлагают и полнотекстовый поиск, и поиск по категориям, таким образом, сочетая в себе достоинства индексирующего сервера и каталога.

<http://www.yahoo.com/>                      <http://altavista.digital.com/>

<http://www.hotbot.com/>                      <http://www.infoseek.com/>

<http://www.northernlight.com/>

Попробуйте также поработать с поисковым сервером Google, расположенным по адресу <http://www.google.com/>. Не упустите эту возможность использовать последние достижения в области поисковых технологий Интернета.

Рост числа пользователей Интернета в России сопровождается активным наполнением российской части сети собственным содержанием. По оценкам журнала "Эксперт", в 1996 году обращения пользователей российской Сети к российским серверам составляли всего 10% от общего числа обращений, в то время как в 1999 году - уже свыше 60%. Происходит быстрое накопление разнообразных ресурсов как на столичных, так и на региональных серверах. Российский Web уже накопил информационную массу, необходимую для полноценной работы русскоязычных пользователей.

Если веб-страница, которую вы ищете, расположена в русской части Сети, возможно, стоит воспользоваться русскими поисковыми серверами. Они лучше работают с русскоязычными поисковыми запросами, снабжены интерфейсом на русском языке. Индексирующие сервера российского Интернета:

- <http://www.rambler.ru/>
- <http://www.aport.ru/>
- <http://www.yandex.ru/>

Российские каталоги общего назначения:

- <http://www.ru/>
- <http://www.atrus.ru/>
- <http://www.stars.ru/>
- <http://www.list.ru/>

В российской части Сети есть и специализированные каталоги. Типичный пример российского специализированного каталога - АгроРус, каталог российских Интернет-ресурсов по сельскому хозяйству. АгроРус был создан Фондом Исследования Аграрного Развития при поддержке Фонда Евразия.

WWW представляет собой место, где пересекаются информационные потоки всего Интернета. Для доступа к одной из самых интересных служб Сети, службе новостей (news), также известной как USENET, помимо традиционных специализированных программ (news-клиентов), можно использовать веб-интерфейсы, такие как Dejanews(<http://www.dejanews.com/>). Помимо чтения и отправки сообщений эти сервера позволяют осуществлять поиск нужного сообщения в огромном архиве. Это намного увеличивает эффективность работы с USENETом.

Если вы ищете файл с определенным названием, но не знаете, на каком FTP-сервере его найти или с какого FTP-сервера окажется быстрее его загрузить, вам поможет FTP Search (<http://ftpsearch.lycos.com/> - изобретенная в Норвегии служба поиска по ftp-серверам, расположенная на WWW. Это еще один яркий пример удачной интеграции различных служб Сети на основе Web. Существует российский аналог FTP Search, его адрес - <http://www.filesearch.ru/>.

Еще один яркий пример удачной интеграции различных служб Сети на основе Web - архивы программного обеспечения, такие как <http://www.tucows.com/>, <http://www.windows95.com/>, <http://www.freeware.ru/>. Они организованы по сходному с поисковыми каталогами принципу: сначала пользователь выбирает зеркало (копию) архива, с которым он может обмениваться данными на большой скорости, затем свою операционную систему, потом интересующую его категорию программ и конкретную программу, которая поможет решить его насущные проблемы. Архивы организованы по-разному; так, не на всех нужно выбирать зеркало

архива. Но в большинстве своем они имеют удобный и понятный пользователю интерфейс, с которым просто разобраться.

В этой главе мы хотели рассказать о том, как получить доступ к информации, находящейся в Сети. О том, что вы найдете в Интернете, рассказать невозможно. Каждый человек обнаружит здесь для себя что-то особенное. Существенно одно: вас ждет много открытий.

**Справочная карта по составлению запросов в поисковой системе Altavista <http://www.altavista.com>**

Действие	Реализация	Пример
Простой запрос.		
Найти документы, содержащие любое из нескольких ключевых слов. Наличие заглавной буквы конкретизирует поиск	Перечислить нужные слова через пробел	Найти все документы, в которых будут присутствовать слова поиск (и)или altavista: поиск altavista
Включить обязательное наличие в документе определенного слова	Поставить перед словом знак плюс	Найти все документы, в которых обязательно будет слово поиск и возможно слово altavista: +поиск altavista
Исключить наличие в документе	Поставить перед словом знак	Найти все документы, в которых может быть слово поиск и не будет

определенного слова	минус	слово новости: поиск - новости
Найти точную фразу.	Найти все документы, в которых будет встречаться фраза "Поиск работы в Интернет": "Поиск работы в Интернет"	
Найти слово со всевозможными окончаниями (поиск по шаблону)	Вместо окончания поставить символ звездочка	Найти все документы со словом Интернет или Интернета или Интернетом и т.д.: интернет*
Запрос с использованием специальных ключевых элементов для поиска в WWW.		
Найти все документы на определенном компьютере с данным доменным именем	host:доменный адрес	Найти все документы на сервере www.symbol.ru: host:www.symbol.ru
Найти все документы внутри	domain:имя-домена	Найти все страницы из Беларуси: domain:by

указанного домена		
Найти страницы, содержащие указанное слово или фразу в тексте или в ссылках	anchor:текст	Найти все страницы, на которых есть фраза "Click here to visit Alta Vista": anchor:"Click here to visit Alta Vista"
Найти все документы с определенным графическим файлом	image:имя-файла	Найти все страницы с файлом bird.jpg: image:bird.jpg
Найти все документы, содержащие определенный Java-апплет	applet:класс	Найти страницы, использующие апплет с названием morph: applet:morph
Найти все документы, ссылающиеся на указанный адрес	link:URL или доменный адрес	Найти все страницы со ссылками на узел Microsoft: link:www.microsoft.com
Найти документ с определенным текстом в заголовке	title:"текст"	Найти все документы с названием "Обучающие курсы по Интернет": title:"Обучающие курсы по Интернет"

Найти все документы, содержащие определенный текст в любом компоненте страницы, кроме тэга изображения, ссылки или URL	text:текст	Найти страницы, содержащие текст поисковые системы: text:"поисковые системы"
Найти документы, в адресе которых присутствует заданная часть URL	url:фрагмент URL.	Найти страницы на серверах, имеющих слово altavista в URL: url:altavista
Сложный запрос, бланк сложного запроса Advanced Search, поле Boolean query:		
Найти документы, в которых отсутствует указанное слово	(NOT Слово)	Найти документы, в которых будет слово интернет и не будет слова новости: интернет AND (NOT новости)
Найти документы, в которых будет хотя бы одно из перечисленных слов	Слово1 OR Слово2	Найти документы, связанные с радио или телевидением: радио OR телевидение
Найти документы,	Слово1 AND	Найти документы, в

в которых будут все слова	Слово2 AND Слово3	в которых присутствуют слова поиск, работа и интернет: поиск AND работа AND интернет
Найти документы, в которых слова должны быть расположены рядом, то есть между ними не должно быть более 10 слов.	Слово1 NEAR Слово2	Найти документы, в которых слово интернет и словосочетание поиск работы должны быть недалеко: интернет NEAR "поиск работы"
Формирование сложной фразы запроса	Использование скобок ()	Найти московских провайдеров Интернет: (провайдер NEAR (интернет* OR internet)) AND моск*

**Справочная карта по составлению запросов в поисковой системе Яндекс (<http://www.yandex.ru>)**

Действие	Реализация	Пример
Простой запрос.		
Найти документы, содержащие все ключевые слова	Перечислить нужные слова через пробел	Найти все документы, в которых будут присутствовать слово идет (идти, шел, шла и

со всевозможными формами в пределах одного абзаца. Учитывает верхний регистр		т.д.) и слово человек (люди): идет человек
Обязательное включение/ выключение слова	Поставить + или -перед словом соответственно	Найти документы, в которых будет слово поисковые и не будет слова машины: +поисковые -машины
Поиск по точной словоформе	Оператор !	Найти все документы, содержащие словоформу Лужкову: !Лужкову
Поиск документов с несколькими словами подряд	ЗаклЮчить слова в двойные кавычки "	Найти документы, в которых слова художественная выставка стоят подряд: "художественная выставка"
Использование логических и специальных операторов		
Найти документы, содержащие все	Оператор &	Найти все документы, в которых будут присутствовать слова

ключевые слова со всевозможными формами в пределах одного абзаца		идет(идти, шел, шла и т.д.) человек(люди): идет & человек
Поиск документов, содержащих любое из слов	Оператор   ,	Найти документы, содержащие в одном абзаце слово вода или огонь: вода   огонь вода,огонь
Исключение одного из слов в пределах абзаца	Оператор ~	Найти документы, содержащие слово холодный, рядом с которым (в пределах абзаца) нет слова горячий: холодный ~ горячий
Действие оператора в пределах документа	Удвоить соответствующий оператор	Найти документы, содержащие оба слова в пределах документа: холодная && вода Найти документы со словом квас, но без слова теплый: квас ~~ теплый

<p>Установка расстояния между словами не более определенного числа слов</p>	<p>Оператор /число</p>	<p>Найти документы, в которых содержатся слова холодный и вода в пределах одного абзаца, и расстоянием между ними не более двух слов: холодная /2 вода</p>
<p>Установка точного расстояния между словами</p>	<p>Оператор /+число</p>	<p>Найти документы, в которых слово вода должно стоять сразу за словом холодный: холодная /+1 вода</p>
<p>Использование сложных выражений</p>	<p>Скобки ()</p>	<p>Найти документы, содержащие любую из фраз холодная вода, горячая вода: (холодный, горячий) /+1 вода</p>
<p>Общий вид ограничения по расстоянию между словами. Если знаки ограничения стоят после двойных операторов, то употребленные</p>	<p>/(n m), где n - минимальное, а m - максимально допустимое расстояние</p>	<p>Найти документы, в которых слово вода должна находиться от холодный в интервале от 2 слов слева до 4 слов справа: холодная / (-2 4) вода Найти документы, в которых слово вода должно находиться в</p>

числа определяются в абзацах		том же самом, либо в соседнем со словом холодный абзаце: холодная && /1 вода
Поиск файла изображения	#image="значение"	Найти документы с изображением орла: #image="eagle*"
Поиск на заданном сайте	#url="значение"	Найти документы на сайте www.comptek.ru: #url="www.comptek.ru"
Поиск ссылок на заданный URL	#link="значение"	Найти документы, ссылающиеся на поисковую систему Yandex: #link="www.yandex.ru"
Ранжирование результатов поиска (задание веса слова или выражения)	слово:                      число (поисковое выражение):число	Найти документы со словами поисковые механизмы с сортировкой результатов по слову механизмы: поисковые механизмы:5
Задание уточняющего слова или выражения для увеличения	<-слово <-(уточняющее выражение)	Найти документы со словом компьютер, при этом первыми будут выданы документы,

релевантности документов, содержащих уточняющее выражение		содержащие слово телефон: компьютер <- телефон
Поиск в части документа		
Поиск в заголовках документов	\$title (выражение)	Найти документы, в заголовках которых есть слово КомТек: \$title (КомпТек)
Поиск в тексте ссылок	\$anchor (выражение)	Найти документы, в тексте ссылок которых есть одно из слов КомпТек или Dialogic: \$anchor (КомпТек   Dialogic)
Поиск ключевых словах	#keywords=(выражение)	Найти документы, в meta теге keywords которых будут слова поисковая система: #keywords=(поисковая система)
Поиск в описании (аннотации) документа	#abstract=(выражение)	Найти документы, в meta теге abstract которых будут слова поисковая система: #abstract=(поисковая

		система)
Поиск в подписях изображением	к #hint=(выражение)	Найти документы, в подписи под изображением которых будет слово кино: #hint=(кино)

**Справочная карта по составлению запросов в поисковой системе Рэмблер (<http://www.rambler.ru>)**

Действие	Реализация	Пример
Простой запрос.		
Поиск документов, содержащих все ключевые слова. Наличие заглавных букв не влияет на результаты поиска	Перечислить нужные слова через пробел	Найти все документы, в которых будут присутствовать слова поисковые и системы: поисковые системы
Поиск слова со всевозможными окончаниями (поиск по шаблону)	Вместо окончания поставить символ звездочка *	Найти все документы со словом информатика или информатике и т.д.: информатик*
Усечение слов	Для обозначения произвольного символа используется	Найти документы со словами поет, поют, поэт и т.д.: по?т

	метасимвол ?	
Использование логических и специальных операторов		
Поиск документов, содержащих любое из нескольких ключевых слов	Оператор OR 	Найти все документы, в которых будут присутствовать слова deep (и)или purple: deep OR purple
Исключение наличия в документе определенного слова	Оператор NOT !	Найти все документы, в которых может быть слово фрукты и не будет слова яблоки: фрукты (Not яблоки)
Определение порядка действия логических операторов	Скобки ()	Найти документы, где встречаются одновременно слова быстрый и поиск, либо качественный и поиск: (быстрый OR качественный) & поиск
Поиск документов, содержащих различные формы поискового слова	Символ @	Найти документы, в которых есть различные формы слова аминокислота (аминокислота, аминокислоту и т.д.): @аминокислота

Справочная карта по составлению запросов в поисковой системе Апорт (<http://www.aport.ru>)

Действие	Реализация	Пример
Простой запрос.		
Найти документы, содержащие все ключевые слова	Перечислить нужные слова через пробел	Найти все документы, в которых будут присутствовать слова яблоко и снег: яблоко снег яблоки на снегу снег на яблоке
Поиск слова со всевозможными окончаниями (поиск по шаблону)	Вместо окончания поставить символ звездочка	Найти все документы со словом Ельцин или Ельцину и т.д.: ельцин*
Поиск словосочетания или близкого к нему. Стоп-слова игнорируются, грамматическая форма не фиксируется	" "	Найти документы, содержащие следующие фрагменты "яблоки на снегу", "яблоки и снег", "яблоко снег": "яблоки на снегу"
Использование логических и специальных операторов		

Найти документы, содержащие любое из нескольких ключевых слов	Оператор ИЛИ OR	Найти все документы, в которых будут присутствовать слова поиск (и)или altavista: поиск ИЛИ altavista
Исключить наличие в документе определенного слова	Оператор НЕ NOT -	Найти все документы, в которых может быть слово фрукты и не будет слова яблоки: фрукты не яблоки
Определить порядок действия логических операторов	Скобки ()	Найти документы, где встречаются одновременно слова быстрый и поиск, либо качественный и поиск: (быстрый или качественный) и поиск
Ограничение расстояния в словах	w2(...)	Слова папа, мама и сын должны встречаться в пределах 5 слов: w5(папа мама сын)
Ограничение расстояния в предложении	s2(...)	Слова папа, мама и сын должны встречаться в пределах одного предложения: s1(папа мама сын)
Поиск по заданной части URL	Оператор url=	1. Найти все документы, проиндексированные Апорт на <a href="http://www.intel.ru">www.intel.ru</a> :

	url:	url=www.intel.ru 2. Найти все документы сервера www.intel.ru , содержащие слова IPI и условия url=www.intel.ru & IPI & условия
Ограничение поиска документами, попадающими в заданный интервал дат	Оператор date= date:	1. Найти все документы, содержащие слово папа и имеющие дату от 1 января 1998 до 9 марта 1999 года: папа date=01/01/1998-09/03/1999 2. Найти все документы, содержащие слово мама, и имеющие дату не позднее 10 октября 1998 года: date:-10/10/1998 мама
Поиск в части документа		
Поиск слова в заголовках документов	title=	Найти документы, содержащие слова мама или папа в заголовке: title=(папа или мама)
Поиск слов в поле META KEYWORDS	keywords=	Найти документы, в которых автор в ключевых словах указал слова мама и папа: keywords=(пап и мама)

Поиск слов или конструкций в полях комментариев к картинкам	alt=	Найти документы, в которых есть картинка с комментарием, содержащим слова Билл и Гейтс: alt=(Билл Гейтс)
Поиск документов, в которых есть ссылки на определенный сервер	link=	Найти документы, в которых есть ссылки на документы с сервера <a href="http://www.agama.ru">www.agama.ru</a> : link= <a href="http://www.agama.ru">www.agama.ru</a>
Поиск слова или конструкции в полях комментариев	comment=	Найти документы, в которых в комментариях встречаются оба слова обожаю и Интернет: comment=(обожаю Интернет)
Поиск слов в обычном тексте	text=	Найти документы, в тексте которых встречаются слова оглавление или содержание: text=(оглавление or содержание)

**Вопросы для самоконтроля:**

1. Что такое компьютерная сеть?
2. Из каких частей состоит электронное письмо?
3. Перечислить информационные услуги компьютерных сетей.
4. Какие линии связи вы знаете? Чем они отличаются?
5. Что такое протокол сети?
6. Что такое Web-браузер?
7. Какими способами можно найти нужную Web-страницу?
8. Что такое компьютер-сервер?
9. Правила создания электронного ящика
10. Что такое Интернет?
11. Почему в качестве линий связи чаще всего используются телефонные линии?
12. Какая информация может присутствовать на Web-странице?
13. Где располагается электронный ящик абонента?
14. Что такое киберпространство?
15. Для чего нужен сервер?
16. Какая сеть называется региональной?
17. Из каких частей состоит электронный адрес?
18. Что такое модем? Для чего он нужен? Каковы характеристики современных модемов (скорость передачи данных)?
19. Что такое гипермедиа?
20. Какой язык является международным языком общения в сети Интернет?
21. Как организована связь между страницами в Интернет?
22. Чем локальная сеть отличается от глобальной?
23. Перечислить технические средства компьютерных сетей.
24. Что такое телеконференции?
25. Достоинства электронной почты перед обычной?
26. Что такое хост-машина?
27. Что такое Гипертекст?
28. Какую функцию выполняют поисковые программы? Какие поисковые программы вы знаете?
29. Что такое терминал?
30. Какие функции выполняет локальная сеть?

31. Что такое Электронная почта?
32. Правила работы с электронной почтой
33. Какими возможностями обладает современный модем?
34. Что такое WWW?
35. Что такое Web-сервер?

**ЛИТЕРАТУРА**  
**СПИСОК РЕКОМЕНДУЕМОЙ ЛИТЕРАТУРЫ ПО**  
**КУРСУ «ИНФОРМАТИКА»**

Информатика.

1. Акулов О.А. Информатика: базовый курс: учеб. для студентов вузов, бакалавров, магистров, обучающихся по направлениям 552800, 654600 «Информатика и вычислительная техника»/О.А.Акулов, Н.В. Медведев. – 4-ое изд., стер. – М.: Омега Л, 2007. – 560 с.
  2. Степанов А.Н. Информатика. Учебник для вузов. 6-ое изд. – СПб.: Питер, 2010. – 720 с.
  3. **Симонович С.В. Информатика. Базовый курс : учебное пособие для вузов / ред. С. В. Симонович. - 2-е изд. - СПб.: Питер, 2006, 2007, 2008, 2009**
  4. Могилев А.В., Пак Н.И., Хеннер Е.К. Информатика. – М.: Академия, 2009. – 848 с.
  5. Острейковский В.А., Полякова И.В. Информатика. Теория и практика. – М.: Оникс, 2008, – 608 с.
  6. Соболев Б.В., Галин А.Б., Панов Ю.В. и др. Информатика: Учебник. Высшее образование. – М.: Феникс, 2007. – 446 с.  
Вычислительные методы. Численные методы.
  1. **Пантина И.В., Синчуков А.В. Вычислительная математика: учебник. / И.В. Пантина, А.В.Сивчуков. (университетская серия). – М.: Маркет ДС, 2010. – 176 с.**
  2. Сальвадори М. Дж. Численные методы в технике. / М.Дж. Сальвадори; пер. с англ. О.В. Локуциевского. – М.: Вузовская книга, 2007. – 264 с.
  3. Устинов С.М. Вычислительная математика. / С.М. Устинов, В.А. Зимницкий – СПб.: БХВ – Петербург, 2009. – 336 с.
  4. Рыжиков Ю.И. Вычислительные методы. – СПб.: БХВ – Петербург, 2007. – 400 с.
- MathCad
1. Гурский Д.А. Вычисления в MathCad.:/Д.А. Гурский. – Минск.: Новое знание, 2003. – 813 с.: ил.
  2. **Дьяконов В.П. Система MathCad.: Справочник/В.П. Дьяконов. – М.: Радио и связь, 1993. – 128 с.: ил.**

3. Кирьянов Д.В. Mathcad 14 / Д.В. Кирьянов. – СПб.: БХВ-Петербург, 2007. – 682 с.
4. Очков В.Ф. MathCad 7 Pro для студентов и инженеров / В.Ф. Очков. – М.: Компьютер-Пресс, 1998. – 384 с.: ил.
5. **Дьяконов В.П. MathCad 2001: Специальный справочник.: справочное издание / В.П. Дьяконов. – СПб.: Питер, 2002. – 832 с.: ил.**

C/C++

1. **Страуструп Б. Язык программирования C++. Специальное издание: пер. с англ. – М.: Бином-Пресс, 2008. – 1098 с.**
2. Культин Н.Б. C/C++ в задачах и примерах / Н.Б. Культин. – СПб.: БХВ-Петербург, 281 с.
3. Побегайло А.П. C/C++ для студента: [учебное пособие] / А. П. Побегайло. - СПб. : БХВ-Петербург, 2006. – 526 с.
4. Шлее М. Qt. Профессиональное программирование на C++ : Наиболее полное руководство / М. Шлее. - СПб. : БХВ-Петербург, 2005. - 544 с.

*Наиболее важные источники в списке рекомендуемой литературы выделены жирным шрифтом.*