

Документ подписан простой электронной подписью  
Информация о владельце:  
ФИО: Блинова Светлана Павловна  
Должность: Заместитель директора по учебно-воспитательной работе  
Дата подписания: 22.03.2023 14:48  
Уникальный программный ключ:  
1cafd4e102a27ce11a89a2a7ceb20237f7ab5c65

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
Высшего образования  
«Норильский государственный индустриальный институт»  
Политехнический колледж

**Методические указания для студентов  
Заочной формы обучения**

**По дисциплине**

**«Информационные технологии в профессиональной деятельности»**

Для специальности:

23.02.03 Техническое обслуживание и ремонт автомобильного транспорта

Методические указания созданы в помощь студентам заочной формы обучения. В данном пособии указаны основные требования, предъявляемые к оформлению и выполнению контрольных работ. Дана справочная информация, задания, список рекомендуемой литературы.

Методические указания учебной дисциплины «Информационные технологии в профессиональной деятельности» разработана на основе Федерального государственного образовательного стандарта по специальности среднего профессионального образования

23.02.03 Техническое обслуживание и ремонт автомобильного транспорта

Организация-разработчик: ФГБОУ ВО «Норильский государственный индустриальный институт» Политехнический колледж

Разработчик: М. В. Олейник

Рассмотрена на заседании предметной комиссии естественнонаучных дисциплин

Председатель комиссии \_\_\_\_\_ М.В. Олейник

Утверждена методическим советом ФГБОУ ВО «Норильский государственный индустриальный институт» Политехнический колледж

Протокол заседания методического совета № \_\_\_\_ от « \_\_\_\_ » \_\_\_\_\_ 20 \_\_\_\_ г.

Зам. директора по УР \_\_\_\_\_ С.П. Блинова

## **Введение**

Данные методические указания являются базовыми для изучения предмета «Информационные технологии в профессиональной деятельности» и предназначены для студентов заочной формы обучения специальности

23.02.03 Техническое обслуживание и ремонт автомобильного транспорта

### **1 Цели и задачи выполнения контрольной работы**

Задания для выполнения контрольной работы по курсу «Информационные технологии в профессиональной деятельности» составлены в соответствии с требованиями ФГОС по специальности 23.02.03 Техническое обслуживание и ремонт автомобильного транспорта и требованиями к обязательному минимуму содержания и уровню их подготовки.

В результате освоения дисциплины обучающийся должен

В результате освоения учебной дисциплины обучающийся должен **уметь:**

- использовать программное обеспечение в профессиональной деятельности;
- применять компьютерные и телекоммуникационные средства;
- работать с информационными справочно-правовыми системами;
- использовать прикладные программы в профессиональной деятельности;
- работать с электронной почтой;
- использовать ресурсы локальных и глобальных информационных сетей;

В результате освоения учебной дисциплины обучающийся должен **знать:**

- состав, функции информационных и телекоммуникационных технологий, возможности их использования в профессиональной деятельности;
- основные правила и методы работы с пакетами прикладных программ;
- понятие информационных систем и информационных технологий;
- понятие правовой информации как среды информационной системы;
- назначение, возможности, структуру, принцип работы информационных справочно-правовых систем;
- теоретические основы, виды и структуру баз данных;
- возможности сетевых технологий работы с информацией;

Учебным планом предусмотрена одна контрольная работа по курсу данной дисциплины.

В процессе выполнения контрольной работы студент должен проявить способность к самостоятельной работе с научно-технической литературой,

уметь обобщать полученные знания, делать научно-обоснованные выводы.

## **2 Методические указания по выполнению контрольной работы**

Контрольная работа является одним из основных элементов обучения студентов в рамках указанной дисциплины и выполняется студентом после консультации с преподавателем. Необходимую консультацию студент может получить и в процессе выполнения им контрольной работы. Выполненная контрольная работа регистрируется в учебном отделе заочного отделения не позднее, чем за 2 недели до сессии.

В ходе выполнения контрольной работы студенты заочной формы обучения изучают теоретический материал и отвечают на вопросы в офисной программе MS Word.

## **3 Оформление работы**

Работа должна быть выполнена на компьютере и распечатана на бумаге формата А4. Контрольная работа должна содержать титульный лист (Рисунок 3.1).

В самой работе в колонтитулах указывать Ф.И.О., группу. Установите параметры страницы размер бумаги А4; ориентация – книжная; поля: левое – 3 см, правое – 1,5 см, верхнее – 3 см, нижнее – 1,5 см, Шрифт Times New Roman, размер 14, отступ первой строки 1,25, выравнивание – по ширине, междустрочный интервал – одинарный.

## **4 Критерии оценки работы**

Оцениваются: выполнение работы, правильные ответы, использование современных источников сети Internet, соблюдение, правильность оформления.

Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«НОРИЛЬСКИЙ ГОСУДАРСТВЕННЫЙ ИНДУСТРИАЛЬНЫЙ ИНСТИТУТ»  
Политехнический колледж

Контрольная работа  
по дисциплине «Информационные технологии в профессиональ-  
ной деятельности»

Студент группы \_\_\_\_\_

Ф.И.О \_\_\_\_\_

Ф.И.О. (преподавателя) \_\_\_\_\_

Оценка \_\_\_\_\_

0000 г.

Рисунок 3.1 Титульный лист

## **5 Тематический план**

1 ТЕХНИЧЕСКИЕ СРЕДСТВА РЕАЛИЗАЦИИ ИНФОРМАЦИОННЫХ ПРОЦЕССОВ

2 ПРОГРАММНЫЕ СРЕДСТВА РЕАЛИЗАЦИИ ИНФОРМАЦИОННЫХ ПРОЦЕССОВ

3 МОДЕЛИ РЕШЕНИЯ ФУНКЦИОНАЛЬНЫХ И ВЫЧИСЛИТЕЛЬНЫХ ЗАДАЧ

4 ЛОКАЛЬНЫЕ И ГЛОБАЛЬНЫЕ СЕТИ ЭВМ. МЕТОДЫ ЗАЩИТЫ ИНФОРМАЦИИ.

## 6 Теоретические сведения

### 1 ТЕХНИЧЕСКИЕ СРЕДСТВА РЕАЛИЗАЦИИ ИНФОРМАЦИОННЫХ ПРОЦЕССОВ

#### 1.1 Этапы развития вычислительной техники

Историю совершенствования механизмов, облегчающих вычисления, можно разделить на три основных этапа:

1) **механический**: регистрируются механические перемещения элементов конструкции. Так как при этом можно предусмотреть любое количество различных состояний, конструкции этого этапа ориентированы на десятичную систему счисления. В истории развития этих механизмов можно выделить следующие этапы:

– *простейшие ручные приспособления* (период с IV тысячелетия до н.э). К ним относятся палочки, счёты абак: глиняная пластинка с желобами, в которых определённым образом раскладывались камешки, русские счёты: камешки нанизаны на проволоку;

– *вычислительные устройства*: арифмометры разных конструкций (с середины XVII века). Первый удобный для расчетов арифмометр создал Блез Паскаль в 1642 году. Его машина могла выполнять сложение и вычитание чисел с 6-8 разрядами и имела небольшие габариты. Следующий этап в принципиальном усовершенствовании арифмометров принадлежит Лейбницу. В 1673 году он представил машину, которая могла выполнять четыре арифметических действия;

– *автоматизация вычислений* — механические устройства, работающие по заданной программе. Идея разделения информации на команды и данные принадлежит Чарльзу Бэббиджу, который в 1822 году представил машину, которая могла рассчитывать таблицы не очень сложных функций.

В механических арифмометрах использовался принцип работы часового механизма: система взаимосвязанных зубчатых колес разного диаметра, в которой поворот каждого колеса на один зубчик соответствовал изменению на единицу определенного разряда числа.

2) **Электромеханический** — в счетных устройствах используются электромагнитные реле (первая половина XX века). Первая машина такого типа была построена немецким инженером Конрадом Цузе в 1941 году. В 1943 году появились машины Марк-1, затем Марк-2, созданные американцем Говардом Эйкенем. Эти машины выполняли арифметические операции с 23-значными десятичными числами и работали гораздо быстрее механических.

3) **Электронный** — регистрируются не механические смещения, а состояния элементов конструкции. При этом оказалось удобнее всего использовать не десятичную, а двоичную систему счисления (включе-

но/выключено, заряжено/разряжено, есть контакт/нет контакта). Первая машина такого типа, ENIAC (Electronic Numeral Integrator And Computer), была создана в США под руководством группы специалистов Говарда Эйкена, Дж. Моучли, П. Эккерта и введена в эксплуатацию 15.02.1945 г.

По элементной базе выделяют 5 поколений ЭВМ (периоды указаны условно):

- **первое поколение** — на электровакуумных лампах (1945— 1955 г.г.);

- **второе поколение** — на транзисторах (1955-1965 г.г.);

- **третье поколение** — на микросхемах. Разрабатываются семейства машин с единой архитектурой, что приводит к программной совместимости, т. е. при появлении новой марки ЭВМ отпала необходимость переписывать заново все программы, которые были разработаны для предыдущей марки (1965-1970 г.г.);

- **четвёртое поколение** — на интегральных схемах. Это существенно увеличило скорость работы, уменьшило энергоёмкость, стоимость и габариты ЭВМ. Происходит переход к персональным ЭВМ. Создаются многопроцессорные и многомашинные комплексы (с 1970 г.);

- **пятое поколение** — суперкомпьютеры на больших интегральных схемах. Используются магнитные, лазерные, голографические принципы различения состояний. Машины этого поколения ориентированы на логическое программирование (обслуживание экспертных систем, плохо формализованных задач).

## 1.2 Принципы работы электронной вычислительной системы

Переход от механического принципа работы счётных элементов к электронному в развитии вычислительной техники ознаменовался резким увеличением быстродействия и объёма памяти компьютеров. Это учёл Джон фон Нейман в своём докладе при формулировке основных принципов, по которым должны функционировать и строиться компьютеры (1945 г.):

- 1 Возможность ввода программы и исходных данных в память компьютера.

- 2 Использование двоичного кода для внутримашинного представления команд и данных.

- 3 Считывание первой команды из ячейки памяти и организация её выполнения.

- 4 Организация автоматического управления последовательностью выполнения последующих команд.

Для реализации этих принципов компьютер должен быть снабжён:

1. Внешним устройством для ввода/вывода информации.



2. Арифметико-логическим устройством для выполнения арифметических и логических операций.
3. Устройством управления для организации порядка выполнения программ.
4. Запоминающим устройством для хранения программ и данных.

Принципиальная схема взаимодействия перечисленных выше устройств представлена на рис. 1.1.

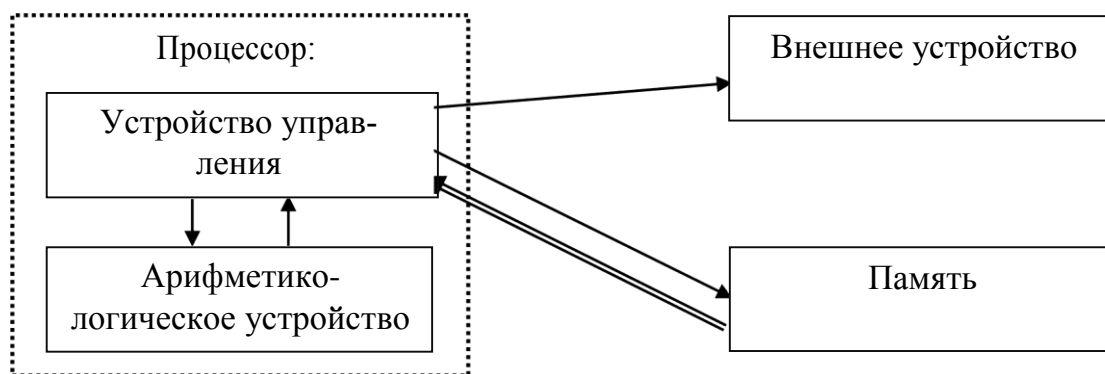


Рис. 1.1. Общая схема организации связей в компьютере (одинарные стрелки — пути и направления передачи управляющих сигналов, двойные — пути и направления движения информации).

Принципы, изложенные в докладе Джона фон Неймана используются в компьютерах всех поколений, конечно с детализацией способов реального воплощения указанных блоков и способов организации связей между ними.

### 1.3 Состав и назначение основных элементов персонального компьютера

#### Обобщающие термины

- *вычислительная техника*: совокупность устройств, предназначенных для автоматической или автоматизированной обработки данных;
- *вычислительная система*: набор взаимодействующих между собой устройств и программ, предназначенный для обслуживания одного рабочего участка;
- *архитектура ЭВМ*: описание принципов действия, информационных связей и взаимного соединения основных узлов компьютера.
- *конфигурации аппаратная 7 программная*: состав аппаратных / программных средств, входящих в вычислительную систему;
- *hardware*: совокупность аппаратных средств компьютера.

#### Виды внутренней памяти

- *оперативная (ОЗУ)*: энергозависимая. Тип памяти: RAM (Random Access Memory) — память прямого доступа. Это означает, что время обращения к любой ячейке памяти одинаково;
  - **кэш-память**: энергозависимая. Сверхоперативное ЗУ небольшого объёма для обмена информацией между ОЗУ и микропроцессором. Компенсирует разницу в скорости работы этих устройств;
- *постоянная память (ПЗУ)*: энергонезависимая, предназначена для хранения BIOS (basic input/output system), программ запуска и остановки компьютера. Тип памяти ROM (Read Only Memory) — только для чтения;
- **CMOS RAM: энергонезависимая**, энергопотребление обеспечивается внутренней батареей. Используется для хранения информации о конфигурации и составе оборудования компьютера, а также о режимах его работы;
- *flash memory*: энергонезависимая, перепрограммируемая постоянная память;
- *видеопамять (VRAM)*: энергозависимая, предназначена для хранения закодированных изображений.

#### Виды внешней памяти (ВЗУ)

ВЗУ используют для длительного хранения программ и данных в двоичном коде. Все виды памяти этого назначения энергонезависимые.

- **жёсткий диск, винчестер, HDD**: Hard Disk Drive;
- **гибкий диск, Floppy Disk**: в настоящее время практически не используется;
- **оптические диски**: CD-ROM (Compact Disk Read Only Memory) — одноразовая заводская запись, далее — только считывание. CD - R (Compact Disk Recordable) одноразовая запись на компьютере, далее — только считывание.

CD-RW (Compact Disk ReWriteable) многократная перезапись на ком-

пьютере. Емкость от 700 Мб до 1,5 Уб‘

– **цифровые видеодиски:** DVD-ROM, DVD-R, DVD-RAM, DVD-RW. Емкость до 17 Гб;

– *стримеры* — накопители на магнитных лентах, используются для резервного копирования больших объёмов информации.

### Процессор

– *центральный процессор* содержит: УУ (устройство управления), АЛУ (арифметико-логическое устройство), регистры (элементы памяти), счётчик команд, кэш-память, математический сопроцессор. Крепится на материнской плате. Основные параметры: *тактовая частота* — влияет на скорость выполнения команд (от МГц до ГГц) и *длина машинного слова* — количество бит, которое может быть обработано за один такт (обычно кратно байтам)

– *триггер, разряд* — электронная схема для хранения одной двоичной цифры;

– *регистр* — ячейка памяти процессора для кратковременного хранения данных или команды в процессе её выполнения.

### Системный блок

– **материнская (системная) плата:** используется для крепления основных устройств компьютера (процессора, ОЗУ, ПЗУ, Кэш-память, интерфейсные схемы шин, гнезда расширений (слотов), обязательные системные средства ввода/вывода).

– **шина (магистраль, системная шина):** включает в себя *шину данных* (разрядность 8, 16, 32, 64 бита), *шину с адресом* (разрядность 16, 20, 24, 32, 36 битов) и *шину управления*. Обеспечивает обмен информацией между процессором и устройствами компьютера.

Реализуется как пучок проводов, по каждому из которых передаётся 1 бит информации. Совокупность проводов, по которым передаются адреса ОЗУ, — *адресная шина*, данных из этих адресов — *шина данных*, сигналов, определяющих характер обмена информацией по магистрали (считывание, запись, синхронизация обмена информацией между устройствами и т.д.) — **шина управления.**

– *порты:* разъемы, с помощью которых к компьютеру подключаются внешние устройства. Они выведены на заднюю панель системного блока.

– *последовательные порты COM1, COM2):* передают электрические импульсы один за другим. К ним подключают мышь и внешний модем.

– **параллельный порт:** обеспечивает более высокую скорость передачи информации, чем последовательные, так как передает одновременно 8 электрических импульсов, к нему подключают принтер.

– **USB-порты** (Universal Serial BUS — универсальная последовательная шина): обеспечивают высокоскоростное подключение периферийных устройств: сканеров, цифровых камер.

— PS/2-порт: обычно подключается клавиатура.

### **Устройства ввода**

Устройства ввода преобразуют информацию из естественной формы, доступной органам чувств человека, в двоичную форму:

- **клавиатура:** для ввода текстовой информации;
- **сканеры:** для текстовой и графической информации, зафиксированной на бумаге;
- **дигитайзеры** (графические планшеты): для ввода графической информации без промежуточной фиксации её на бумаге (она рисуется на специальном планшете световым пером);
- **манипулятор мышь,**
- **манипулятор джойстик** — наклоны ручки эквивалентны перемещению мыши, для игровых программ;
- **манипулятор трекбол** — встроенный в стационарный корпус шарик, прокрутка которого эквивалентна перемещению мыши. Используется в оптических мышках и ноутбуках;
- **манипулятор пенмаус** — похож на шариковую ручку, на конце которой находится узел, регистрирующий её перемещения;
- **цифровые видеокамеры и фотоаппараты,**
- **микрофон.**

### **Устройства вывода**

Устройства вывода преобразуют информацию из двоичной формы в естественную форму, доступную органам чувств человека:

- **монитор.** Основные характеристики: тип экрана, размер экрана по диагонали (обычно в дюймах), разрешающая способность. Для жидкокристаллических мониторов — угол обзора.
- **адаптер:** устройство для соединения блоков компьютера с разными способами представления информации;
- **контроллер:** устройство для сопряжения разных устройств и управления их работой.
- **принтеры: матричные, струйные, лазерные.** Различаются по способу печати;
- **плоттеры** (графопостроители): выводят документы больших размеров (чертежи, плакаты и т. п.);
- **акустические стереоколонки.**

## 2 ПРОГРАММНЫЕ СРЕДСТВА РЕАЛИЗАЦИИ ИНФОРМАЦИОННЫХ ПРОЦЕССОВ

*Программное обеспечение ЭВМ (ПО)* — это множество программ, которые используются или могут быть использованы на компьютере.

Совокупность всех программных средств компьютера и требующихся им данных обычно обозначают термином *software*. По типу функций, которые выполняют программы, выделяют три основных группы ПО: *системное, инструментальное, прикладное*.

К нему также относят области деятельности, необходимые для проектирования и разработки программ разного типа:

технологии проектирования программ (нисходящее, восходящее проектирование, структурное, объектно-ориентированное программирование и т. п.);

методы отладки и тестирования программ;

- анализ качества работы программ;
- документирование программ и т. п.

### 2.1 Системное программное обеспечение ЭВМ

*Системное ПО* обеспечивает эффективную работу аппаратуры компьютера. Основные типы системных программ:

— *операционная система (ОС)*: совокупность программ, тестирующих все устройства компьютера и управляющих их работой, а также организующих взаимодействие пользователя с компьютером (пользовательский интерфейс).

Основные функции ОС:

- загрузка программ в оперативную память и управление ходом их выполнения;
  - обмен данными между выполняющейся программой и внешними запоминающими устройствами;
  - обслуживание нестандартных ситуаций в ходе выполнения программы;
  - цикл работ по окончанию работы программы;
  - организация хранения и поиска информации на внешних носителях;
  - организация интерфейса пользователя;
- выполнение сервисных функций: форматирование дисков, копирование файлов и т. п.

Основные ОС:

**MS DOS** — однопользовательская и однопрограммная ОС, в которой все команды по работе с файлами пользователь должен был набирать вручную в текстовом режиме;

**UNIX, LINUX** — используются на серверах сети Интернет;

**Windows NT/2000/XP/8** — многопрограммные, многопользовательские, сетевые ОС;

**Android** — портативная (сетевая) операционная система основанная на ядре Linux, применяемая в коммуникаторах, планшетных компьютерах, электронных книгах, цифровых проигрывателях, наручных часах, нетбуках и смартфонах;

- *операционные оболочки*: улучшают пользовательский интерфейс, предусмотренный в ОС. Табличные оболочки (Norton Commander для MS DOS, Far Manager и Norton Navigator для Windows) представляют каталоги в виде текстовых таблиц и основные команды манипуляций с файлами можно не вводить вручную, а запускать функциональными клавишами. В графических оболочках для указания типа файла используются графические значки, для заказа типовых действий — специальные способы указания на них (щелчки мышью, горячие клавиши вместо ввода полного текста команд). Эти оболочки используются во всех версиях Windows;

- *драйверы*: программы, которые расширяют возможности ОС по управлению работой устройств компьютера. Техническая реализация одного и того же типа устройства в разных моделях существенно различается. Для того, чтобы согласовать ее работу с остальными устройствами, для каждой модели одного и того же устройства разрабатывают специальную программу — драйвер. В процессе установки ОС определяет тип и конкретную модель каждого устройства компьютера и загружает соответствующий ему драйвер;

- *утилиты, (сервисные программы)*: небольшие программы, выполняющие действия обслуживающего характера и увеличивающие эффективность работы отдельных устройств, например, программы для диагностики различных устройств (Scandisk, Check Disk), программы оптимизации, «кэширования» и динамического сжатия дисков, и т. п.;

- *архиваторы*: программы, которые переписывают файлы в сжимающем их коде, используются для подготовки файлов к пересылке по сети и для хранения дублирующих копий важной информации;

- *антивирусные программы*: проверяют уже имеющиеся в компьютере и вновь поступающие в него программы на наличие действий, которые могут повредить ОС или файлы пользователя.

## 2.2 Файловая структура ОС. Операции с файлами.

**Файл** — это именованная совокупность любых данных, размещённая на внешнем запоминающем устройстве и хранимая, обрабатываемая и пере-

мещаемая как единое целое.

Совокупность правил и программ, по которым выполняются операции с файлами, называется *файловой системой*. Основные из них: *FAT* (File Allocation Table) используется в операционных системах для 16-и и 32-х разрядных процессоров, *NTFS* (New Technology File System) используется, начиная с ОС Windows NT для 32-х и 64-х разрядных процессоров). В них пользователю предоставляются следующие возможности:

- создание папок;
- копирование, перемещение, переименование и удаление файлов и папок;
- навигация по файловой структуре
- запуск программ и открытие документов;
- создание ярлыков;
- стандарты для обозначения пути к файлу: абсолютный и относительный адрес файла.

*Абсолютный адрес* начинается указанием диска, на котором расположен файл, и далее последовательно через символ «\» (обратный слеш) перечисляются все папки, которые следует открыть, чтобы найти нужный файл.

*В относительном адресе* через символ «\» перечисление папок ведется, начиная от той, которая активна в данный момент. Для перехода в папку нижнего уровня следует указать имя этой папки. Переход в папку верхнего уровня вне зависимости от ее реального названия обозначается символом

«..» (две точки).

Файл характеризуется свойствами и атрибутами. Их значения указываются в диалоговом окне «Свойства», которое открывается по команде «Свойства» в меню «Файл» и в контекстном меню. Основные из них:

— *имя файла* может содержать до 256 символов и состоит из двух частей, которые разделяются точкой. Первая часть — имя, которое назначает пользователь для того, чтобы в дальнейшем было понятно, что за информация хранится в этом файле. Обычно имя пользователя состоит из букв русского и латинского алфавита, цифр, пробелов, дефиса. Остальные символы использовать не рекомендуется, хотя некоторые — допускается. Вторая часть — расширение. Оно указывает на тип файла или программу, которая в дальнейшем должна работать с ним. В нем разрешается использовать не более 4-х символов. Пояснения к стандартным расширениям, с которыми часто встречаются пользователи, приведены в табл.2.1.

- *длина файла* в байтах (длина занимаемого участка на диске);
- *время и дата* создания файла (для опознания последних по времени вариантов файла).

Кроме перечисленных выше атрибутов каждому файлу сопоставляют-

ся атрибуты, определяющие допустимые действия с ним. Например:

- *только для чтения* (исправления, сделанные во время просмотра, не будут сохраняться после закрытия файла);
- *скрытый* (не будет высвечиваться в каталоге), но открыть его можно, введя имя вручную;
- *архивный* (для автоматического обновления изменённых версий в архивах);
- *системный*.

Таблица 2.1 Основные стандартные расширения файлов

Расширение	Формат/ назначение программы
Системные программы/ программы в машинных кодах	
*.com *.pcx	системные программы в машинных кодах
*.sys	программа драйвера
*.bak	автоматически создаваемая резервная копия
*.tmp	временная копия с промежуточной информацией, автоматически уничтожаемая после закрытия программы
*.hlp	файл встроенной файловой системой
*.exe	прикладная программа в машинных кодах
Графические форматы	
*. bmp	форматы растровой графики, в котором задан цвет каждого пикселя
*.jpg	форматы сжатой растровой графики, удобный для хранения отсканированных фотографий и иллюстраций. Используется в Интернете.
*.gif *.tif *.png	формат для диаграмм, создаваемых программным путем, рисунков типа аппликации с ограниченным набором цветов, несложной анимации. Используется в Интернете.
*.psd *.pdd	формат растровых графических файлов в PhotoShop
Расширения основных программ Microsoft office	
*.xls, *.xlsx	форматы файлов программы Excel до 2003 и после 2003 года
*.ppt, *.pptx	формат файлов электронных презентаций программы PowerPoint до 2003 и после 2003 года
*.mdb, *.mdbx	формат файлов программы Access до 2003 и после 2003 года
*.htm *.html	формат файлов для Веб-страниц



Для того, чтобы применить нужную команду не к одному файлу, а к нескольким, в именах которых есть общие для них фрагменты, вводится понятие *групповое имя файла (маска имени)*. В маске в явном виде указываются символы, которые обязательно должны присутствовать в нужных файлах, остальные символы заменяются шаблонными:

\* означает, что на этом месте может оказаться любое количество любых символов, в частности, ничего;

? означает, что на этом обязательно должен быть какой-нибудь символ, но только один;

^ означает, что требуется обработать одной командой все файлы, указанные в предыдущей маске, кроме файлов, подходящих под маску, указанную после него.

Примеры:

\*.\* — под эту маску подходят все файлы (любое имя и любое расширение);

\*.jpg — маска указывает на все файлы рисунков с расширением

\*экза\*.\* маска указывает на файлы с любым расширением, у которых в имени пользователя есть указанный фрагмент текста;

\*.\*|\*.bak” — означает, что требуется обработать одной командой все файлы, кроме файлов с расширением “.bak”, какими бы ни были их имена.

### 2.3 Инструментальное программное обеспечение ЭВМ

*Инструментальное ПО* служит для разработки программ, применяемых в самых разных областях деятельности человека.

В истории программирования можно выделить следующие этапы по способам написания программ:

- программирование в машинных кодах;
- использование машинно-ориентированных языков низкого уровня;
- использование алгоритмических языков высокого уровня.

Для того чтобы ЭВМ могла выполнять программу, она должна быть записана по строгим правилам в виде, доступном процессору, т. е. представлять собой последовательность двоичных чисел и кодов. Такие коды называются *машинными кодами, машинными командами*. Программа, написанная таким образом, — программой на машинном языке или исполняемым модулем.

Первоначально, когда появились ЭВМ, программисты для разработки программ использовали машинный язык. Это было очень трудно и неудобно, так как приходилось самому распределять память под команды программы и

данные, держать в памяти массу абстрактных двоичных кодов, обозначающих адреса данных и команд, которые их обрабатывают.

На втором этапе для облегчения программирования была создана программа, которая автоматически заменяла в программах удобные человеку названия переменных и операций на машинные команды. Эта процедура была обозначена термином *трансляция*, а программа, выполняющая ее, *транслятором*. На этом этапе трансляторы могли переводить в машинные коды конструкции, которые заменялись одной или несколькими (условно не больше пяти) машинными командами. Для совокупности правил, по которым следует записывать программу, чтобы ее мог обработать такой транслятор, используют термины *ассемблер*, *макроассемблер*.

Ассемблер — это машинно-ориентированный язык низкого уровня. Алгоритм, по которому строится программа, приходится дробить на очень мелкие шаги, которые умеет выполнять процессор. При появлении новых моделей процессоров с иной структурой команд и встроенных операций приходится создавать новые версии ассемблера, трансляторов с него и переписывать под них пользовательские программы.

Следующий этап развития инструментального ПО связан с появлением *алгоритмических языков*, каждый из которых содержит конструкции, удобные для записи алгоритмов в той или иной сфере человеческой деятельности. Алгоритмические языки относятся к классу языков высокого уровня. Они позволяют при составлении программы дробить алгоритм на более крупные смысловые блоки, не связанные со структурой и типом команд ЭВМ, на которой будет работать программа. При изменении модели процессора переписывается только транслятор с алгоритмического языка, а пользовательские программы остаются прежними.

## 2.4 Основные понятия алгоритмических языков. Алфавит. Синтаксис. Семантика

Обычный разговорный язык состоит из четырех основных элементов: символов, слов, словосочетаний и предложений. Алгоритмический язык содержит подобные элементы, только слова называют *элементарными конструкциями*, словосочетания — *выражениями*, предложения — *операторами*. Алгоритмический язык (как и любой другой язык), образуют три его составляющие: алфавит, синтаксис и семантика.

*Алфавит* — фиксированный для данного языка набор символов (букв, цифр, специальных знаков и т.д.), которые могут быть использованы при написании программы.

*Синтаксис* — правила использования символов алфавита в специальных конструкциях, с помощью которых составляется алгоритм.

*Семантика* — система правил толкования конструкций языка. Таким образом, программа составляется с помощью соединения символов алфавита в соответствии с синтаксическими правилами и с учетом правил семантики.

Программа, написанная на алгоритмическом языке, называется исходным модулем.

Для ее трансляции можно использовать два типа технологий: *интерпретация* или *компиляцию*. Трансляторы, реализующие эти технологии, называются соответственно *интерпретаторами* и *компиляторами*.

При интерпретации транслятор переводит в машинные коды очередную строку исходного модуля, сразу же выполняет её и переходит к обработке следующей строки, не создавая законченного исполняемого модуля всей программы в целом. Такой тип трансляции удобен на этапе отладки или тестирования программы.

При компиляции трансляция разбивается на два этапа (рис. 2.1). На первом компилятор создает *объектный модуль* — промежуточную программу, в которой конструкции исходного модуля переведены в машинные коды, но вместо реальных адресов данных и работающих с ней подпрограмм, используются относительные адреса и не добавлены служебные подпрограммы, необходимые для её работы.

На втором этапе программа-сборщик заменяет в объектном модуле относительные адреса на абсолютные и подключает к нему необходимые для работы стандартные программы.

В результате получается *исполняемый модуль* — программа, написанная в машинных кодах и полностью готовая к работе.

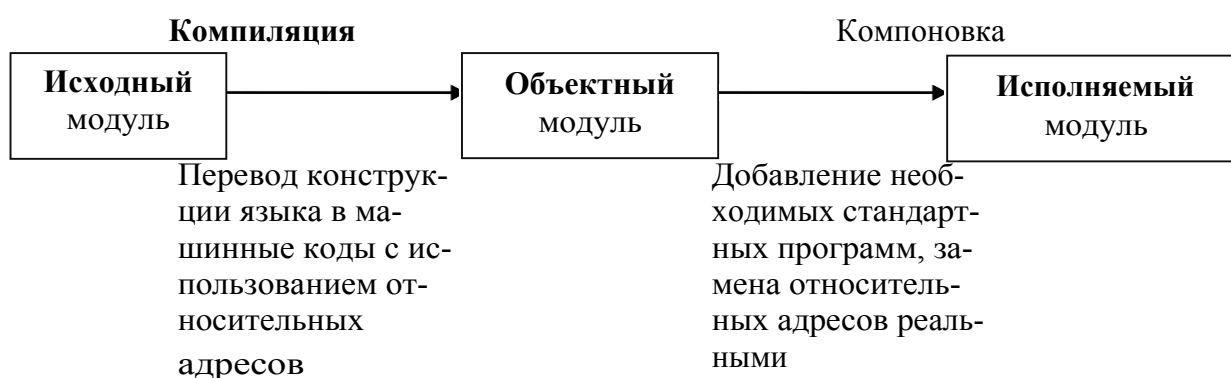


Рис. 2.1. Схема трансляции в режиме компиляции

Этот режим трансляции удобен для расчётов по уже отлаженной и протестированной программе, так как исполняемый модуль можно сохранить в памяти и при необходимости провести расчеты с другим набором исходных данных, не повторяя трансляцию.

В настоящее время для большинства алгоритмических языков разработаны **интегрированные системы программирования** (Бейсик, Паскаль, Си и т. п.). Они включают в себя следующие компоненты:

- текстовый редактор для написания программы;
- библиотеки стандартных программ;
- средства отладки;
- справочную службу;
- компилятор и/или интерпретатор;
- диалоговый интерфейс.

## 2.5 Основные алгоритмические языки высокого уровня

Один из первых языков программирования высокого уровня, **Фортран** (Formula Translation), был создан в середине 50-х годов. Он используется для инженерных и научных расчетов, для решения задач физики и других наук с развитым математическим аппаратом. Благодаря своей простоте и тому, что на этом языке накоплены большие библиотеки программ, Фортран и в наши дни остается одним из самых распространенных алгоритмических языков.

Для решения экономических задач был создан язык программирования Кобол.

Расширение областей применения ЭВМ влечет за собой создание языков, ориентированных на новые сферы применения: *Снобол* — алгоритмический язык для обработки текстовой информации. Лисп - алгоритмический язык для обработки символов. Он находит широкое применение в исследованиях по созданию искусственного интеллекта.

В 1968 г. был объявлен конкурс на лучший язык программирования для обучения студентов. Для этого конкурса *Никлаус Вирт* создал язык *Паскаль*, достаточно простой, удобный, с наличием мощных средств структурирования данных. Хотя Паскаль был разработан как язык для обучения программированию, он впоследствии получил широкое развитие и в настоящее время считается одним из самых используемых языков.

Для обучения младших школьников Самуэлем Пайпертом был разработан язык Лого. Он отличается простотой и богатыми возможностями.

Широкое распространение в школах в качестве обучающего языка получил язык *Бейсик*, позволяющий взаимодействовать с ЭВМ в режиме непосредственного диалога. Спустя много лет после изобретения Бейсика, он и сегодня самый простой для освоения из десятков языков общецелевого про-

граммирования.

Необходимость разработки больших программ, управляющих работой ЭВМ, потребовала создания специального языка программирования. В начале 70-х г. был разработан язык **СИ**, который широко используется как инструментальный язык для разработки операционных систем, трансляторов, баз данных и других системных и прикладных программ. Это язык программирования общего назначения. Во многих случаях программы, написанные на Си, сравнимы по скорости с программами, написанными на языке Ассемблера. При этом они имеют лучшую наглядность и их более просто сопровождать. В отличие от Паскаля, в нем заложены возможности непосредственного обращения к некоторым машинным командам и к определенным участкам памяти компьютера.

Появление функционального программирования привело к созданию языка **Пролог** (ПРОграммы ЛОГические). Этот язык программирования разрабатывался для задач анализа и понимания естественных языков на основе языка формальной логики и методов автоматического доказательства теорем.

В 80-х г. 20 века был создан язык Ада. В дополнение к классическим свойствам, он обеспечивает программирование задач реального времени и моделирования параллельного решения задач.

К языкам сверхвысокого уровня можно отнести Алгол-68 и APL. Повышение уровня этих языков произошло за счет введения сверхмощных операций и операторов.

В современной информатике можно выделить два основных направления развития языков программирования: процедурное и не процедурное.

В **процедурных** языках программа явно описывает определенную последовательность действий, которые необходимо выполнить, чтобы получить результат.

Среди процедурных языков выделяют в свою очередь **структурные** и **операционные языки**. В структурных языках одним оператором записываются целые алгоритмические структуры: ветвления, циклы и т.д. В операционных языках для этого используются несколько операций. Широко распространены следующие структурные языки: **Паскаль, Си, Ада, ПЛ/1**. Среди операционных известны **Фортран, Бейсик, Фокал**.

**Непроцедурное (декларативное)** программирование появилось в начале 70-х годов 20 века, но стремительное его развитие началось в 80-е годы, когда был разработан японский проект создания ЭВМ пятого поколения, целью которого явилась подготовка почвы для создания интеллектуальных машин. К непроцедурному программированию относятся **функциональные** и **логические** языки.

В **функциональных языках** программа описывает вычисление некоторой функции. Обычно эта функция задается как композиция других, более

простых, те в свою очередь разлагаются на еще более простые и т.д. Один из основных элементов в функциональных языках – *рекурсия*, то есть вычисление значения функции через значение этой же функции от других элементов. Присваивания и циклов в классических функциональных языках нет.

В *логических языках* программа вообще не описывает действий. Она задает данные и соотношения между ними. После этого системе можно задавать вопросы. Машина перебирает известные и заданные в программе данные и находит ответ на вопрос. Порядок перебора не описывается в программе, а неявно задается самим языком. Классическим языком логического программирования считается *Пролог*. Построение логической программы вообще не требует алгоритмического мышления, программа описывает статические отношения объектов, а динамика находится в механизме перебора и скрыта от программиста.

Можно выделить еще один класс языков программирования *объектно-ориентированные языки высокого уровня*. На таких языках не описывают подробной последовательности действий для решения задачи, хотя они содержат элементы процедурного программирования. Объектно-ориентированные языки, благодаря богатому пользовательскому интерфейсу, предлагают человеку решить задачу в удобной для него форме. Примером такого языка может служить язык программирования визуального общения *Object Pascal*.

*Языки описания сценарием*, такие как *Perl, Python, Rexx, Tcl* и языки оболочек *UNIX*, предназначаются не для написания приложения с нуля, а для комбинирования компонентов, набор которых создается заранее при помощи других языков. Развитие и рост популярности Internet также способствовали распространению языков описания сценариев. Так, для написания сценариев широко употребляется язык Perl, а среди разработчиков Web-страниц популярен *JavaScript*.

## 2.6 Прикладное программное обеспечение ЭВМ

*Прикладное ПО* — это программы, предназначенные для решения индивидуальных задач пользователя или классов задач в конкретной области применения информационных технологий (проблемной области). Программы этого типа можно разбить на три группы

### *Индивидуальное прикладное ПО:*

– игровые и развлекательные пакеты;

Программы, разрабатываемые для отдельного пользователя или организации.

### *Стандартное прикладное ПО:*

– текстовые редакторы и процессоры;

- графические редакторы;
- программы электронных презентаций;
- электронные таблицы;
- системы управления базами данных; бухгалтерские и финансовые пакеты;
- системы автоматизированного проектирования;
- издательские системы;
- системы документооборота; программы-переводчики;
- поддержка электронной почты;
- образовательные, обучающие программы, мультимедийные энциклопедии;
- мультимедийные программы для воспроизводства, создания и редактирования звуко- и видеозаписей;
- и т.д.

***Интегрированные пакеты программ:***

– MS Office, Open Office, Corel Word Perfect Office, Star Office и т. п. Такие пакеты представляют собой совокупность разных стандартных прикладных программ, охватывающих все типы деятельности в той предметной области, для которой он создан, обладающих однотипным интерфейсом и средствами передачи информации между различными компонентами пакета. Пример: интегрированный пакет «издательская система» должен содержать:

- текстовый редактор;
- орфографический корректор;
- программу слияния текстов;
- программу формирования оглавлений и составления указателей;
- автоматический поиск и замену слов и фраз;
- средства телекоммуникаций;
- электронную таблицу;
- СУБД;
- модули графического оформления;
- графический редактор;
- набор разных шрифтов;
- и т. п.

## 2.7 Общие сведения о графических редакторах

В настоящее время существует несколько технологий создания рисунка в ЭВМ:

- растровая;
- векторная;
- фрактальная;

– flash-графика.

*Характеристики растровой графики.* Изображение передаётся последовательной записью цвета всех пикселей рисунка. Растровыми являются все сканированные изображения и цифровая фотография.

*Основные редакторы:* Paint, Adobe Photoshop, FotoEditor.

*Основные форматы:* bmp, gif, jpg, png, tif, pdf

*Характерные особенности растровой графики:*

- самое точное воспроизведение цвета;
- в пределах одного замкнутого контура можно создать только один тип заливки, но в дальнейшем его можно корректировать;
- при увеличении изображения появляется лестничный эффект;
- при уменьшении изображения может теряться чёткость и количество цветов;
- можно корректировать часть контура, создавать любые композиции из шаблонных и индивидуальных контуров, которые после их закрепления уже нельзя разложить на составляющие;
- возможна трансформация (растянуть, сжать, отразить, увеличить, уменьшить, сдвинуть, скрутить и т. д.) только прямоугольного выделенного фрагмента изображения;
- можно использовать фильтры для изменения тональности изображения и достижения различных спецэффектов (туман, морская рябь, просмотр через мокрое стекло и т. п.);
- можно вводить в рисунки тексты.

*Характеристики векторной графики.* Изображение составляется, как мозаика, из отдельных **графических примитивов (автофигур)**, которые строятся на указанном месте с помощью формул. Такая графика удобна для создания диаграмм, блок-схем, *Основные редакторы:* графика MS Word, Adobe Illustrator, Corel Draw. 3DStudioMax, 3D Canvas, Lightwave, Maya. Большинство векторных редакторов могут работать и с растровой графикой. *Основные форматы:* wmf, cdr, ps, eps.

*Характерные особенности векторной графики:*

- файлы рисунков имеют существенно меньший объем по сравнению с аналогичными файлами растровой графики;
- допускается масштабирование рисунка без искажений;
- объединение и разделение ранее объединённых фигур на исходные графические примитивы;
- деформирование, перемещение как отдельных графических примитивов, так и их объединённых блоков;
- команды форматирования действуют на фигуру в целом, а не на отдельные её части;
- в пределах одного замкнутого контура можно создать только один



тип заливки, но в дальнейшем его можно корректировать;

- нельзя корректировать часть графического примитива;
- можно использовать эффекты тени для любых и объёма — для замкнутых контуров;
- можно вводить в рисунки тексты.

Разновидность векторной графики — трёхмерная графика.

*Характеристики фрактальной графики.* Удобна для визуализации моделей всевозможных трёхмерных объектов, природных ландшафтов и т. п. Формирование изображений целиком основано на формулах и уравнениях, описывающих эти объекты. Основной редактор: Brass.

*Характеристики flash-графики.* Используется для высококачественных анимационных изображений на Web-страницах и электронной рекламы. Позволяет кодировать качественную анимацию в небольших по размерам файлах. Основной редактор: Macromedia Flash. Помимо редакторов, обеспечивающих полный спектр работ с графикой, широко используются программы-вьюеры (просмотрщики), которые предназначены только для просмотра и печати ранее созданных изображений. К ним относятся, например, Adob Acrobat Reader (просмотр графики и текстов в формате PDF), GSview (просмотр графики и текстов в формате PS и EPS)

### 3 МОДЕЛИ РЕШЕНИЯ ФУНКЦИОНАЛЬНЫХ И ВЫЧИСЛИТЕЛЬНЫХ ЗАДАЧ

#### 3.1 Основные понятия моделирования

**Модель** — упрощённое подобие реального объекта, процесса или явления, которое отражает его существенные особенности.

**Сущность** обобщённое название объекта, явления или процесса, которое изучается с помощью моделирования.

**Атрибуты (параметры)** — характеристики сущности, которые учитываются в её модели.

**Моделирование** — метод познания, состоящий в создании и исследовании моделей изучаемых сущностей.

Каждой сущности можно сопоставить несколько моделей в зависимости от того, для какой цели она создаётся. Пример — возможные модели человека. Для отдела кадров на работе — это его анкета или резюме, в которых учитываются атрибуты, необходимые в профессиональной деятельности; для поликлиники медицинская карта, в которой учитываются атрибуты здоровья; для приятелей — черты характера и набор его хобби; для портного геометрические размеры тела.

*Необходимость создания моделей* диктуется следующими факторами:

- исследования на оригинале может быть экономически невыгодным;
- изучение может приводить к разрушению сущности (моделирова-

ние взрывов, методики лечения, хранения продуктов и т. п.);

– оригинала нет в действительности (изучение сущностей прошлого или будущего);

– необходимо исследование только некоторых свойств оригинала.

*Классификации моделей.*

В зависимости от того, какой фактор является наиболее важным при моделировании, для классификации моделей используют разные признаки:

– по области использования;

– по фактору времени;

– по отрасли знаний;

– по форме представления.

*Классификация моделей по области использования*

*Учебные* — используются при обучении.

*Опытные* — это уменьшенные или увеличенные копии проектируемого объекта. Используют для исследования и прогнозирования его будущих характеристик (аэродинамическая труба).

*Научно-технические* — для исследования процессов и явлений.

*Игровые* — репетиция поведения объекта в различных условиях.

*Имитационные* — отражение реальности в той или иной степени (это метод проб и ошибок).

*Классификация моделей по фактору времени*

*Статические* описывают состояние системы в определенный момент времени (единовременный срез информации по данному объекту). Примеры: классификация животных, строение молекул, список посаженных деревьев, отчет об обследовании состояния зубов и т. д.

*Динамические* — описывают процессы изменения и развития системы (изменения объекта во времени). Примеры: моделирование движения тел, развития организмов, процесс химических реакций.

*Классификация моделей по отрасли знаний* — это классификация по отрасли деятельности человека (математические, биологические, химические, социальные, экономические, исторические и т. д.).

*Классификация моделей по форме представления*

– *материальные (предметные, физические)* — это модели, которые имеют реальное воплощение и отражают внешние свойства или внутреннее устройство моделируемых сущностей, суть процессов и явлений в объекте-оригинале. **Материальное моделирование** — это экспериментальный метод познания окружающей среды. Примеры: детские игрушки, скелет человека, чучело, макет солнечной системы, школьные пособия, физические и химические опыты, авиамодель истребителя, полоса препятствий.

– *информационные* — это целенаправленно отобранная информация о моделируемой сущности, которая отражает ее свойства, наиболее существенные для исследователя. В информационных моделях реальный объ-

ект или процесс заменяется его формальным описанием. Такая процедура называется **формализацией**.

Например, информационной моделью движения поездов является расписание их движения, а материальной макет железной дороги с движущимися паровозиками.

*По уровню формализации различают:*

– хорошо формализованные модели. Их можно решить средствами, принятыми в данной предметной области, не используя субъективные мнения экспертов;

– плохо формализованные модели. Их нельзя решить без привлечения эксперта в данной предметной области.

*Типы информационных моделей*

**Абстрактные (мысленные)** — при построении модели используются понятия, не существующие в реальной жизни. Примеры: модель идеального газа. Она представляет каждую молекулу как материальную точку, т. е. объект, который имеет массу, но не имеет размеров. В модели движения планет вокруг солнца каждая планета тоже представляется как материальная точка.

**Вербальные** мысленные модели, выраженные в разговорной форме с помощью естественных языков. Пример: инструкция пилоту самолёта — это вербальная неформализованная модель, так как она пишется на естественном языке.

**знакомые (формализованные)** — выражены специальными символами, применяемыми в изучаемой предметной области. Например, компьютерная модель реализована средствами программной среды, математическая — формулами, которые описывают изучаемую сущность. Знаковая формализованная модель музыкального произведения — запись с помощью нот и т. д.

В знаковых информационных моделях выделяют класс **образно-знаковых** моделей. Например, к таким моделям относятся:

– **Геометрические** — рисунок, пиктограмма, чертеж, карта, план, объемное изображение;

– **Структурные** — таблица, граф, схема, диаграмма;

– **Алгоритмические** — нумерованный список действий, пошаговое перечисление, блок-схема.

*По способу организации данных* информационные модели делятся на:

– **реляционные (табличные)**: перечень объектов и их свойств оформляется в виде связанных между собою таблиц. Каждая строка таблицы содержит информацию об одном экземпляре (**сущности**) предметной области, каждый столбец — значения одной и той же характеристики (**атрибу-та**) для разных сущностей. Пример: расписание движения поездов — это табличная информационная модель реального перемещения поездов по железной дороге;

– **иерархические:** объекты распределены по уровням. Каждый элемент высокого уровня состоит из элементов нижнего уровня, а элемент нижнего уровня может входить в состав только одного элемента более высокого уровня.

Такие модели представляются ориентированным графом («деревом»), у которого начальная вершина не подчинена никакой другой, а все остальные подчинены только одной, но могут иметь в своём подчинении сколько угодно объектов нижнего уровня. Если из каждого узла выходит только два потомка, то такая структура называется бинарным деревом.

Примеры: файловая структура в компьютере (система каталогов), система доменных имён в Интернете, структура почтовых адресов, классификации животных, растений. В иерархической модели две любые вершины могут быть соединены только одним путём. Пример: относительный путь к файлу имеет только один вариант.

– **Сетевые:** между объектами моделируемой системы существуют множественные связи. Такие модели представляются графом, в котором имеются связи между вершинами, позволяющие создать разные пути перехода между ними.

Примеры: модель функционирования Интернет, где каждый сервер может связаться с любым другим сервером через цепочку промежуточных узлов, и эти цепочки могут быть разными; модель взаимодействия пациентов и врачей в больнице, где каждого больного обследует несколько врачей и в то же время каждый врач следит за здоровьем нескольких больных; модель взаимодействия студентов и преподавателей в процессе обучения.

### 3.2 Базы данных и базы знаний

В наше время любой специалист независимо от сферы деятельности в той или иной мере занимается сбором, накоплением, сортировкой разнообразных данных и прочими операциями их обработки. При структурировании данных по реляционному типу в состав базы обычно входят следующие средства: таблицы, запросы, формы, отчеты, макросы и модули.

**Таблицы** – в них накапливается информация, которая вводится с клавиатуры.

**Запросы** – программы, которые манипулируют данными из таблиц и в черновом виде подготавливают ответы на типовые вопросы пользователей.

**Формы** – программы, размещающие на экране нужную пользователю информацию. Формы используются для вывода тех сведений, которые достаточно просмотреть с экрана и на принтер выводить не нужно.

**Отчеты** – программы, подготавливающие информацию для вывода на принтер.

**Макросы и модули** – программы, в которых производится такая обработка информации, которая недоступна каждому из перечисленных выше средств по отдельности.

Наряду с базами данных в настоящее время при моделировании плохо формализованных задач используют базы знаний. Базы знаний в отличие от баз данных содержат в себе не только фактическую информацию, которая является постоянной для данной предметной области, но и правила вывода, допускающие автоматические умозаключения о вновь вводимых фактах, то есть моделируют «осмысленную» обработку информации с помощью правил логики. Компьютерные модели таких задач называются **интеллектуальными системами**. Математические методы, по которым работают интеллектуальные системы, называются **методами искусственного интеллекта**. В их основе лежат эвристические приёмы.

**Эвристика** – это неформализованная процедура, сокращающая количество шагов поиска решения за счет некоторого способа направленного поиска решения, а не простого перебора всех возможных вариантов. Часто считается, что база знаний отличается от базы данных именно наличием механизма вывода новых знаний из старых. Еще одна особенность, которая закладывается в базы знаний, это выдача системой «объяснения» хода ее рассуждений при поиске ответа.

Наиболее известный класс интеллектуальных систем — это экспертные системы (ЭС). Эти системы рассматриваются как модели поведения экспертов в определенной области знаний. База знаний ЭС создается при помощи трех групп специалистов:

- эксперты той проблемной области, к которой относятся задачи, решаемые ЭС;

- инженеры по знаниям, являющиеся специалистами по разработке интеллектуальной информационной системы (ИИС);

- программисты, осуществляющие реализацию ЭС.

ЭС может функционировать в 2-х режимах: ввод фактов по данной предметной области или проведение консультаций. В режиме ввода знаний эксперт с помощью инженера по знаниям вводит известные ему сведения о *предметной области* в базу знаний ЭС. В режиме консультации пользователь ведет диалог с ЭС, сообщая ей сведения о *текущей задаче* и получает рекомендации ЭС, сопровождающиеся «объяснением хода ее рассуждений». Например, в сфере тушения лесных пожаров на основе информации о пожаре (описание растительности в данной области, сводка погоды и т. п.), выдается некая рекомендация относительно стратегии борьбы с данным пожаром (оценка требуемых ресурсов, рекомендации по размещению техники и т. п.)

### 3.3 Этапы моделирования

Создание компьютерной модели разбивается на несколько этапов:

- 1. Постановка задачи:** описание сущности, подлежащей моделированию, цели исследования и той информации о моделируемом объекте, которую необходимо учитывать в связи с поставленной целью.

**2. Разработка модели:** выбор типа создаваемой модели (предметная, компьютерная, математическая и т. п.) и ее реализация. Если выбран компьютерный тип модели, создается соответствующая программа.

**3. Тестирование модели:** проведение компьютерного или натурального эксперимента на тестах. *Тест* набор данных, для которых заранее известен результат.

Для компьютерных программ применяются разные технологии тестирования. Если тестируется программа, которая получена путем внесения изменений, улучшающих ранее работавшую программу, то, чтобы убедиться, что внесенные изменения не испортили ее, в первую очередь ее проверяют на тех тестах, по которым проверяли предыдущую программу. Такое тестирование называется *регрессионным*. Тестирование новых сложных программ разбивается на два этапа:

*альфа-тестирование* — это проверка программы на тестах, которые составил ее разработчик.

*Бета-тестирование* — передача программы заказчикам в бесплатное пользование с условием, что при возникновении ошибок в работе разработчик будет уведомлен об этом.

**4. Анализ результатов моделирования:** процесс проверки правильности модели на совокупности тестов, охватывающих все диапазоны моделируемых параметров. Если результаты соответствуют цели, построение модели закончено, и можно использовать модель для прогнозирования реальных результатов. В противном случае модель уточняется за счет введения в нее новых данных, не учтенных на первом этапе или изменяется ее реализация.

## 4 ЛОКАЛЬНЫЕ И ГЛОБАЛЬНЫЕ СЕТИ ЭВМ. МЕТОДЫ ЗАЩИТЫ ИНФОРМАЦИИ.

### 4.1 Классификация вычислительных сетей

**Компьютерная сеть** — это два или больше компьютеров, связанных каналами передачи информации. Цель создания сетей обеспечение совместного доступа **к сетевым ресурсам**.

*Виды сетей по географическому признаку и размерам:*

– *локальная:* сеть предприятия или учреждения, в которой рабочие станции распределены на расстоянии не более 300-500м. Обозначается аббревиатурами ЛВС или LAN (Local Areal Net);

– *региональные и корпоративные:* объединяют компьютеры большого региона или филиалы учреждений, разбросанные в пределах города, большого географического региона, страны. Обозначается аббревиатурой MAN. (Metropolitan Areal Net);

– *глобальные:* объединяют всех абонентов (LAN и MAN) вне зависимости от места их расположения: страны, континента, всей земли, напри-

мер, Интернет. Обозначается аббревиатурой WAN. (World Areal Net).

*Виды сетей по возможностям передачи сигналов:*

- *узкополосные:* канал связи может передавать только один сигнал в любой момент времени (телефонная линия);
- *широкополосные:* одновременно можно передавать несколько сигналов, используя для каждого свою частоту передачи (кабельное телевидение).

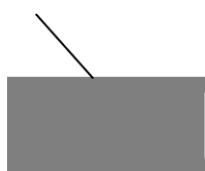
#### 4.2 Виды сетевых ресурсов

- *аппаратные:* общий принтер, общий жёсткий диск для хранения программ и данных отдельных пользователей;
- *программные:* для выполнения сложных и длительных расчётов можно подключиться к мощной ЭВМ, послать на неё задание на расчёты и исходные данные, и, по окончании расчётов просто получить готовые результаты.
- *информационные:* всевозможные справочные данные, архивы научных работ, книг и т. п.

*Виды каналов передачи информации:*

- *специальные кабели:* коаксиальный кабель, оптоволоконный кабель (самая надёжная и быстрая связь), витая пара - телефонный

Защитное покрытие

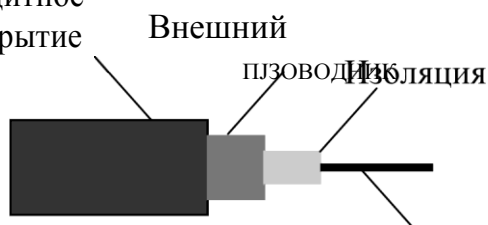


Стекло-  
покрытие

Оптическое  
волокно

Оптоволоконный кабель

Защитное  
покрытие



Внешний

проводник

Изоляция

Внутренний про-  
водник

Коаксиальный кабель

электромагнитные волны разных частот: спутниковая связь, подключение ноутбуков, мобильных телефонов;

Важнейшая характеристика канала — *скорость передачи информации*. В ЛВС обычно от 10 до 100 Мбит/сек., в крупных сетях —

\*\*\*

На скорость передачи информации влияет также тип ее кодировки: чем короче кодировка символов, тем меньше бит она содержит и, следовательно, тем быстрее она передается по тому же самому каналу.

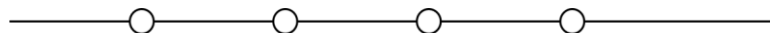
### 4.3 Топология и архитектура вычислительных сетей

**Топология сети** — это логический и физический способ соединения компьютеров. Различают следующие базовые варианты топологий:

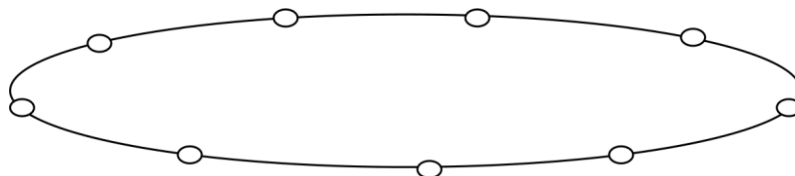
– *сети на основе сервера: сервер* — это компьютер, на котором хранится информация, необходимая разным пользователям или же выполняющий работы по поддержанию связи между компьютерами. Другое значение этого термина — программа, выполняющая запросы с рабочих станций на доступ к ресурсам. *Рабочая станция (клиент)* — это внутренний компьютер ЛВС. Другое значение этого термина — программа, создающая запрос серверу на какие-либо ресурсы;

– *одноранговая*: все компьютеры имеют одинаковые права (обычно до 10 рабочих станций);

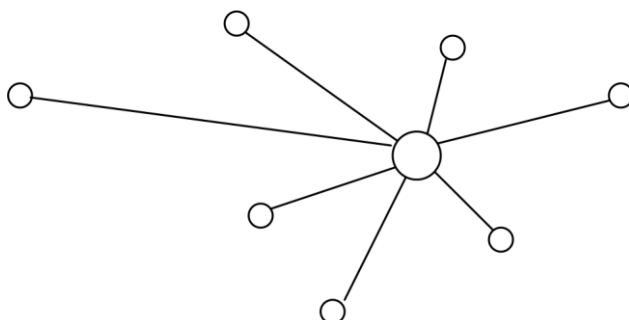
– *линейная (шинная) топология*: соединяющий кабель последовательно проходит от одного компьютера к другому:



– *кольцевая топология*:

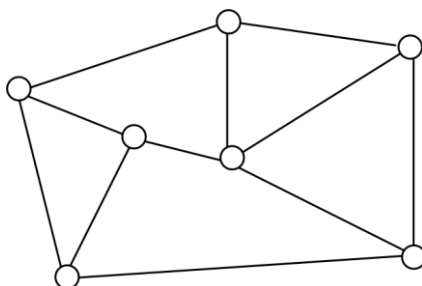


– *звездообразная топология*:





– *полносвязанная (сетевая) топология*: имеется много путей, по которым можно переслать сообщение с одного компьютера на другой.



*Архитектура сети* — это понятие, которое включает в себя топологию сети, состав ее устройств, правила их взаимодействия, кодирование, адресацию и передачу информации, управление потоком сообщений, контроль ошибок, анализ работы в аварийных ситуациях.

Основные виды архитектур (физический уровень):

– **Ethernet**: широковещательная сеть, т.е. все рабочие станции могут принимать все сообщения. Топология линейная или звездообразная.

– **Token Ring**: топология кольцевая. Каждый узел ожидает своей очереди на посылку сообщения.

– **FDDI**: высокоскоростная передача данных по оптоволоконным линиям. Топология смешанная (кольцевая + древовидная)

– **ATM**: передача цифровых данных, видеоинформации и голоса по одним и тем же линиям.

– **Wi-Fi, IrDa**: беспроводная (радиорелейная) связь.

#### 4.4 Программное обеспечение вычислительных сетей

*Сетевые протоколы* — это наборы правил для обмена информацией в сети и разработки сетевого оборудования. Более подробно их можно определить как стандарты, которые определяют формы представления и способы пересылки разных типов сообщений между компьютерами, а также правила совместной работы различного оборудования в сетях, необходимые при создании этого оборудования.

Совокупность протоколов, которые используют компьютеры при работе в сети, обозначается термином *стек*. Традиционно стек делится на 7 уровней, функции которых определяются *эталонной моделью взаимодействия открытых систем* (сетевая модель ОН — Open System Interconnection):

– **Физический уровень (1)**: процедуры управления аппаратурой передачи данных и подключённым к ней каналам связи. Примеры протоколов: Ethernet, Arc net, Token Ring.

– **Канальный уровень (2)**: отвечает за передачу информации по логическому каналу, установленному между двумя ЭВМ, соединёнными физическим каналом. На этом уровне обнаруживаются ошибки передачи пакетов, реализуется алгоритм восстановления информации в случае обнаружения

сбоев или потери данных. Детализируются способы осуществления непосредственной связи объектов сетевого уровня. Примеры протоколов: LAP-B, SNAP, HDLC

– **Сетевой уровень (3):** отвечает за маршрутизацию пакетов и связь между разными сетями, т.е. устанавливает логические каналы между *объектами* для реализации протоколов транспортного уровня. Примеры протоколов: *IP (Internet Protocol)*, IPX, IDN, X.25

– **Транспортный уровень (4):** определяет интерфейс между процессами и сетью, т.е. устанавливает логические каналы между *процессами* и обеспечивает передачу информационных пакетов, сформированных по правилам сетевого уровня. Примеры протоколов: *TCP (Transmission Control Protocol)*, UDP, NSP, X.224, NetBIOS.

– **Сеансовый уровень (5):** осуществляет установку и поддержку сеанса связи между двумя абонентами. Определяются средства, необходимые для синхронизации и управления обменом данными между сетевыми объектами. Примеры протоколов: RPC, X.225

– **Представительский уровень (6):** определяет синтаксис данных, т.е. коды и форматы данных, посылаемых в сеть. Каждая прикладная программа, каждая операционная система имеет свой способ кодировки данных и команд. На этом уровне устанавливаются стандартные способы кодировки для информации, которая выходит из локальной сети. Аппаратное обеспечение, обеспечивающее перекодировку информации при выходе из локальной сети, обозначается термином **шлюз**. Примеры протоколов: X.226

– **Прикладной уровень (7):** совокупность правил для разработки программ-приложений, которые делают запросы к ресурсам, расположенным в сети. В него входит регламентация всех работ, которые связаны с запуском программ пользователя и их выполнением. Примеры протоколов этого уровня: HTTP, SMTP, FTP, DNS, POP3. В рамках этих протоколов пользователь делает запросы на использование сетевых ресурсов.

**Основной протокол**, под управлением которого работает *Интернет*, — это *TCP/IP*. Другими словами совокупность протоколов TCP/IP является основой построения и функционирования сети Интернет.

#### 4.5 Протоколы электронной почты

Серверная часть пакета программ, обслуживающих электронную почту, состоит из трёх основных подсистем: подсистемы хранения сообщений, транспортной подсистемы, службы каталогов.

**Подсистема хранения** обеспечивает хранение и разделение пришедшей почты с помощью учётных записей. Учётная запись содержит учётное имя пользователя (логин: название почтового ящика) и ряд другой информации.

**Транспортная подсистема** обеспечивает пересылку исходящих сообщений от своих клиентов и транзитных сообщений.

**Служба каталогов** обеспечивает хранение и корректировку учётных

записей, направление сообщения именно тому, кому оно направлено.

*Основные протоколы:*

**SMTP** (Simple Mail Transfer Protocol) — простой протокол передачи исходящих сообщений. Он регламентирует передачу только символьной информации;

**POP** (Post Office Protocol) — используется для работы с входящей почтой

**MIME** (Multipurpose Internet Mail Extension) — многоцелевое расширение почты Интернета. Этот протокол регламентирует пересылку писем с присоединёнными файлами музыки или изображений.

#### 4.6 Коммуникационное оборудование

**Повторитель (репитер)** — передаёт электрические сигналы от одного участка кабеля к другому, предварительно усиливая их и восстанавливая их форму. Используется в локальных сетях для увеличения их протяжённости. В терминологии OSI функционирует *на физическом уровне*.

**Коммутаторы** — многопортовые повторители, которые считывают адрес назначения каждого входящего пакета и передают его только через тот порт, который соединён с компьютером-получателем. Могут функционировать *на разных уровнях OSI*.

**Концентратор (hub)** — многопортовое устройство для усиления сигналов при передаче данных. Используется для добавления в сеть рабочих станций или для увеличения расстояния между сервером и рабочей станцией. Работает как коммутатор, но вдобавок может усиливать сигнал.

**Мультиплексор** (устройство или программа) — позволяет передавать по одной коммуникационной линии одновременно несколько различных сигналов.

**Шлюз** — подготавливает данные к передаче между сетями или прикладными программами, использующими разные протоколы (способы кодировки, физические среды для передачи данных). Например, при подключении локальной сети к глобальной. Функционирует *на прикладном уровне*.

**Мост** — соединяет две сети с одинаковыми протоколами, усиливает сигнал и пропускает только те сигналы, которые адресованы компьютеру, находящемуся по другую сторону моста. Мостом также называют компьютер с двумя сетевыми картами, предназначенный для соединения сетей.

**Маршрутизатор** — соединяет разные ЛВС, как и мост, пропускает только ту информацию, которая предназначена для сегмента, с которым он соединён. Отвечает за выбор маршрута передачи пакетов между узлами. Выбор маршрута осуществляется на основе протокола маршрутизации, содержащего информацию о топологии сети, и специального алгоритма маршрутизации. Функционирует *на сетевом уровне OSI*.

## 4.7 Основные понятия криптографии

**Криптографии (шифрование)** — это кодирование данных, посылаемых в сеть, так, чтобы их могли прочитать только стороны, участвующие в конкретной операции. Надёжность защиты информации зависит от алгоритма шифрования и длины ключа в битах.

**Метод шифрование** — это алгоритм, описывающий порядок преобразования исходного сообщения в результирующее.

**Ключ шифрование** — это набор параметров, необходимых для применения метода шифрования. При компьютерном шифровании ключ представляется как последовательность символов, сохранённых на жёстком или съёмном диске. Различают *статические* варианты ключей — они не меняются при работе с разными сообщениями и *динамические ключи* — они изменяются для каждого сообщения.

*Типы методов шифрования:*

– **симметричные:** один и тот же ключ используется и для шифровки, и для дешифровки. Такие методы неудобны в электронной коммерции, так как у продавца и покупателя должны быть разные права к доступу информации. Продавец посылает всем покупателям одни и те же каталоги, но покупатели возвращают ему конфиденциальную информацию о своих кредитных картах, и нельзя смешивать заказы и их оплату для разных покупателей;

– **асимметричные (несимметричные):** основываются на специальных математических методах, которые создают пару ключей так, что-то, что зашифровано одним ключом, может быть дешифровано только другим, и наоборот. Один из ключей называется **открытым**, его может получить каждый желающий. Второй ключ разработчик ключа оставляет себе, он называется **закрытым (секретным)**. Например, в электронной коммерции продавец создает пару ключей. Закрытый ключ он оставляет себе, открытый — посылает покупателю. Если клиент получил файл, к которому не подходит его ключ, значит, его послала не та фирма, с которой он ведет деловую переписку. Покупатели шифруют свои заказы, договоры своим открытым ключом и посылают их продавцу. Их дешифровку может сделать только владелец закрытого ключа, т. е. продавец.

**Защищённый канал** — способ передачи сообщений, при котором обе стороны используют один и тот же метод шифрования, известный только им.

**Электронно-цифровая подпись** — код специальной структуры, который позволяет однозначно связать содержание документа, пересылаемого по сети, и его автора (*аутентифицировать* документ).

**Хэш - функции** — это функции, которые позволяют из одной последовательности чисел получить другую последовательность таким образом, что обратное преобразование невозможно.

**Хэширование** — обработка некоего сообщения хэш-функцией, при

которой двоичные коды сообщения воспринимаются как коды двоичных чисел. В результате создается уникальная последовательность символов фиксированной длины, которая однозначно соответствует содержанию исходного сообщения. Эту последовательность обозначают термином *хэш-код*. Смысл хеширования можно проиллюстрировать бытовым примером: для того, чтобы не забыть какой-нибудь тюк багажа в пути, мы запоминаем количество мест багажа, и, чтобы проверить, не забыли ли мы что-то, просто пересчитываем тюки. Но, зная только количество тюков, мы не сможем получить информацию о том, что в них содержалось.

*Дайджест сообщения, электронная печать, сводка сообщения* — уникальная последовательность символов фиксированной длины, которая создаётся на основе содержания электронного документа. Дайджест получается путём хеширования. Он уникален для сообщения, как отпечатки пальцев для человека. Если изменить хотя бы один символ в документе, хэш-код станет другим.

#### 4.8 Электронно-цифровая подпись (ЭЦП)

Технически шифрование/дешифровка с помощью хэш- функций состоят из следующих этапов:

сообщение дополняется сведениями об авторе и обрабатывается хэш-функцией. Это — ЭЦП;

1. ЭЦП добавляется к документу, который содержит передаваемую информацию и сведения об авторе;

2. полученный файл шифруется ключом асимметричного метода и посылается принимающей стороне;

3. принимающая сторона расшифровывает полученное сообщение с помощью своего ключа, отделяет от него ЭЦП и обрабатывает той же хэш-функцией, что и отправитель;

4. Затем сравнивают полученный хэш-код с тем, который пришёл в сообщении. Если они совпали, значит, сообщение не подверглось изменениям в пути и сведения об авторе правильны.

Абсолютной гарантии в невозможности подделать ЭЦП нет, так же как и для обычной подписи под документом, но подделать ЭЦП или вскрыть ключ гораздо сложнее и в интеллектуальном, и в экономическом плане. Технология взлома — это метод простого перебора вариантов следования символов, из которых можно создать ключ дешифровки. Время взлома определяется производительностью вычислительной техники, используемыми алгоритмами шифрования и длиной ключа (бит). Пример: при симметричном шифровании с ключом в 40 бит надо перебрать  $2^{40}$  вариантов. При нынешней технике это займёт меньше суток. При длине ключа в 128 бит — необходимое время взлома больше, чем возраст Вселенной.

#### 4.9 Компьютерные вирусы

*Компьютерные вирусы* — это программы, которые могут самокопиро-

ваться, скрытно внедрять свои копии в файлы, в загрузочные секторы дисков, искажать документы, работу операционной системы и прикладных программ.

*Каналы распространения вирусов:*

- электронная почта; интернет-сайты; рассылки графических и аудио-файлов;
- непроверенные съёмные диски.

*По «среде обитания» вирусы можно классифицировать как:*

- *файловые:* внедряются в программы и обычно активизируются при их запуске. После запуска программы вирус находится в оперативной памяти и остается активным до перезагрузки ОС;
- *загрузочные:* внедряются в загрузочный сектор диска и активизируются при загрузке ОС;
- *макровирусы:* заражают файлы документов, оформляются в виде макросов, которые искажают содержание документа;
- *сетевые:* распространяются по сети, внедряясь в файлы, пересылаемые по электронной почте, или заражая файлы, скачиваемые с серверов файловых архивов.

К основным разновидностям сетевых вирусов относятся:

- *сетевые черви:* распространяются по сетям, внедряясь в файлы вложений электронных писем. Способны самокопироваться и рассылать свои копии по другим сетевым адресам, имеющимся в компьютере. Парализуют работу сети и портят информацию в компьютере. Активизируются либо автоматически в момент открытия сеанса связи с сетью, либо при просмотре электронной почты.
- *тройанские программы (трояны):* вирусы, производящие несанкционированные действия с информацией и маскирующиеся под привлекательные для пользователя программы. Обычно создают ярлык с ложным названием типа Game.exe. Другими словами их можно охарактеризовать как программы, которые не создают собственных копий, но преодолевают систему защиты компьютера и оказывают вредоносное воздействие на её файловую систему;
- *скрипт-вирусы:* программы, написанные на языках Java- Script или VBScript, которые активизируются при загрузке Web- страниц и выполняют разрушительные действия в программной системе компьютера.

#### **4.10 Классификация антивирусных программ**

В настоящее время существует множество программ, выполняющих антивирусную защиту. Их можно классифицировать по следующим признакам:

*По принципу работы*

- *мониторы* — являются частью ОС, автоматически проверяют на вирусы все поступающие файлы, все открываемые и закрываемые файлы, ес-

ли нужно — попутно лечат. К программам такого типа относятся NOD32, AVP (Antiviral Toolkit Pro — лаборатория Касперского);

- *сканеры* — проверяют файлы, находящиеся в ОЗУ, выдают сообщения о подозрительных файлах, но не лечат их.

- *брандмауэр (firewall)* — программа, которая является фильтром при обмене информацией между внешней и локальной сетью. Она разрешает доступ к информации сети только уполномоченным лицам и задерживает файлы тех типов, которые указаны администратором сети (вирусы, баннеры, файлы запрещенной тематики).

*По принципу обнаружения вирусов:*

- *полифаги*: просматривают коды программ на наличие в них известных вирусных фрагментов или фрагментов кода, типичных для вирусных действий, если можно лечат. Качество работы сильно зависит от даты обновления вирусной базы. Наиболее популярные программы этого типа: NOD32, AVP, Doctor Web, Norton Antivirus,

- *ревизоры*: только диагностируют, заражён ли файл. Для этого они сверяют эталонные контрольные суммы всех неизменяемых файлов в компьютере с контрольными суммами на момент проверки. Те файлы, у которых они не совпали заражены. Качество работы зависит от того, был ли заражён компьютер в момент создания базы эталонных контрольных сумм при загрузке ревизора на компьютер. Примером программы такого типа является Aids test,

- *эвристические анализаторы*: реагируют на фрагменты кода, которые выполняют действия, похожие на вирусные, но не зарегистрированы в антивирусных базах.

*Другие типы антивирусных программ*: фаги, дезинфекторы, вирус фильтры, иммунизаторы, детекторы, вакцины.

Помимо применения антивирусных программ для уменьшения вероятности потери и порчи информации из-за вирусов рекомендуется периодически архивировать важную информацию и хранить ее на съемных носителях. Это позволяет при обнаружении испорченных вирусом файлов восстановить их предыдущие версии, а не создавать их заново. Основные архиваторы, которые используются в настоящее время, это: 7—Zip, ARJ, WinRAR, WinZip. Основные расширения архивов: \*.ZIP, \*.RAR, \*.ARJ.

#### 4.11 Облачная антивирусная защита

Активное развитие телекоммуникационных систем в настоящее время сделало возможным предоставление конечному пользователю доступа к услугам, приложениям и вычислительным ресурсам различной мощности, находящимся на разных серверах. Такие виды услуг называются интернет-облаками или облачными вычислениями.

Одна из услуг, относящаяся к облачным вычислениям, — это антивирусные облака.

Производители антивирусных программ имеют в своем распоряжении гораздо более мощное оборудование и программное обеспечение для распознавания вирусных сигнатур (участков кода, которые могут нанести вред ПО компьютера, файлов в критических местах ЭВМ), чем-то, которое имеет конечный пользователь. При работе в облаке антивирусные программы на компьютерах пользователей все подозрительные файлы, которые они не могут однозначно определить как вирусы, отправляют на сервер своей фирмы-производителя для анализа в экспертной системе. Сервер производителя, со своей стороны, выставляет по выполняемым операциям баллы опасности присланного приложения и расширяет, если нужно, базу вирусных сигнатур. Так как на сервер стекается информация от многих пользователей, такая технология ускоряет обнаружение новых вариантов вирусов и время реакции на них, а также обеспечивает обновление антивирусных баз всех своих пользователей частыми и маленькими порциями.



## 7 Контрольная работа

Работа должна быть выполнена на компьютере и распечатана на бумаге формата А4. Контрольная работа должна содержать титульный лист (Рисунок 3.1).

В самой работе в колонтитулах указывать Ф.И.О., группу. Установите параметры страницы размер бумаги А4; ориентация – книжная; поля: левое – 3 см, правое – 1,5 см, верхнее – 3 см, нижнее – 1,5 см, Шрифт Times New Roman, размер 14, отступ первой строки 1,25, выравнивание – по ширине, междустрочный интервал – одинарный.

В контрольной работе написать вопрос и ниже дать полный ответ.

Вариант последняя цифра номера вашей зачетной книжки.

### Вариант № 1

*Задание № 1.* Укажите назначение системной шины (магистральной).

*Задание № 2.* Укажите устройства, размещаемые на материнской плате ПК.

*Задание № 3.* Укажите, к какому поколению относятся персональные компьютеры.

*Задание № 4.* Укажите, какой из перечисленных терминов обозначает программы, обеспечивающие взаимодействие ОС с периферийными устройствами

1. контроллер;
2. транслятор;
3. драйвер;
4. компилятор.

*Задание № 5.* Укажите, как называется именованная область внешней памяти произвольной длины с определённым количеством информации.

*Задание № 6.* Укажите отличие модели от реального объекта, явления или процесса.

*Задание № 7.* Процесс описания объекта на искусственном языке называется \_\_\_\_\_

*Задание № 8.* Расшифруйте аббревиатуры LAN, MAN, WAN.

*Задание № 9.* Как называется иерархическая система назначения уникальных текстовых имён каждому компьютеру, находящемуся в сети?

*Задание № 10.* Укажите, какая часть электронного адреса ресурса описывает путь к файлу, расположенному на сервере:

<http://www.you.google.com/inf02000/01-02/det123.html>

### Вариант № 2

*Задание № 1.* Как называются электронные схемы для управления

внешними устройствами?

*Задание № 2.* На каких видах внешней памяти невозможно случайно стереть информацию?

*Задание № 3.* По какому признаку классифицируются принтеры?

*Задание № 4.* Укажите, что не входит в основные функции операционной системы:

1. обеспечение диалога с пользователем;
2. разработка программ для ЭВМ;
3. управление ресурсами компьютера;
4. организация файловой структур.

*Задание № 5.* Размер кластера 512 байт, размер файла — 816 байт. Укажите, сколько места на диске займет этот файл.

*Задание № 6.* В основе методов искусственного интеллекта лежат

---

*Задание № 7.* Отличительной чертой интеллектуальных систем является

*Задание № 8.* Укажите тип связи, который на сегодня является наиболее защищённым от несанкционированного доступа: оптоволоконный кабель, телефонный кабель (витая пара), электромагнитные волны.

*Задание № 9.* Укажите, какие термины не относятся к базовым топологиям сетей: линейная (шинная), кольцевая, снежинка, полносвязанная (сетевая).

*Задание № 10.* Укажите среди приведенных терминов, те, которые обозначают беспроводную связь: Ethernet, Wi-Fi, IrDa, FDDI.

### **Вариант № 3**

*Задание № 1.* Что обозначает термин «адаптер»?

*Задание № 2.* Укажите устройства, которые не входят в состав внутренней памяти современного компьютера.

*Задание № 3.* Укажите, какие из названий относятся к устройствам

**Ввод данных**

1. плоттер;
2. процессор;
3. блок питания;
4. монитор;
5. сканер.

*Задание № 4.* Укажите, что определяет расширение файла:

1. размер;

2. имя;
3. Тип;
4. расположение.

*Задание № 5.* Укажите, что относится к основным компонентам системного программного обеспечения:

1. обрабатывающие программы и система автоматизации программирования
2. операционная система и система программирования
3. монитор и супервизор
4. пакеты прикладных программ

*Задание № 6.* Можно ли отнести к плохо формализованным задачам задачи, приводящие к трудоёмким расчётам по известным формулам?

*Задание № 7.* Какие методы используют для плохо формализованных задач?

*Задание № 8.* Укажите, что в сетевой терминологии обозначают термины «сервер», «клиент».

*Задание № 9.* Укажите, что в сетевой терминологии обозначает термин «протокол».

*Задание № 10.* Укажите, какая часть электронного адреса ресурса описывает протокол передачи информации: <http://www.google.com/inf02000/01-02/det123.html>

#### **Вариант № 4**

*Задание № 1* Укажите устройства, которые входят в состав внутренней памяти современного компьютера.

*Задание № 2.* Укажите, какие из названий относятся к устройствам ввода данных:

1. жёсткий диск;
2. мышь;
3. привод CD-ROM;
4. джойстик;
5. регистры.

*Задание № 3.* Укажите устройство хранения данных, которое работает только при включённом питании:

1. ПЗУ;
2. гибкий магнитный диск; 3 ОЗУ;
- 3 жёсткий диск.

*Задание № 4.* Укажите, какая группа файлов будет выделена по маске <\*. \*|\*.bak>.

*Задание № 5.* Перечислите маски для текстовых файлов.

*Задание № 6.* К какому классу моделей относится рецепт кушанья?

*Задание № 7.* Модель гравитационного взаимодействия двух тел, записанная в виде формул, это \_\_\_\_\_

*Задание № 8.* На сколько уровней разбиты протоколы сетевой модели OSI?

*Задание № 9.* Укажите протоколы сетевой модели OSI, относящиеся к прикладному уровню, и расшифруйте их назначение.

*Задание № 10.* Укажите, на каком уровне сетевой модели OSI устанавливаются правила маршрутизации.

### **Вариант № 5**

*Задание № 1.* Укажите назначение системной шины (магистральной).

*Задание № 2.* Укажите, какие из приведенных утверждений являются верными:

1. сетевая плата не является устройством приёма-передачи данных
2. компакт-диск - это оперативная память
3. гибкий магнитный диск - это долговременная память
4. в мониторах на жидких кристаллах отсутствует электромагнитное излучение

*Вопрос № 3.* Укажите наиболее важные характеристики жидкокристаллического монитора:

1. цвет фона окна;
2. физический размер экрана;
3. объём хранимых данных;
4. скорость обработки информации;
5. угол обзора.

*Задание № 4.* Укажите обобщенное название программ для согласования работы внешних и внутренних устройств компьютера

*Задание № 5.* Укажите наиболее известные способы представления графической информации в компьютере.

*Задание № 6.* Моделью Земли для определения законов её движения вокруг Солнца является: \_\_\_\_\_

*Задание № 7.* Укажите этап, на котором осуществляется определение целей моделирования.

*Задание № 8.* Как называется компьютер с двумя сетевыми картами,

предназначенный для соединения сетей.

*Задание № 9.* Выберите правильный вариант назначения электронно цифровой подписи:

1. удостоверение истинности отправителя и целостности сообщения;
2. восстановление повреждённого сообщения;
3. шифрование сообщения для сохранения его секретности;
4. пересылка сообщения по секретному каналу.

*Задание № 10.* Укажите, по какому признаку различаются симметричные и асимметричные методы шифрования.

### **Вариант № 6**

*Задание № 1.* Укажите устройства, размещаемые на материнской плате ПК.

*Задание № 2.* Укажите, к какому поколению относятся персональные компьютеры.

*Задание № 3.* Как называются электронные схемы для управления внешними устройствами?

*Задание № 4.* Укажите графический формат, приводящий при сохранении фотографий к наименьшему объёму файла.

*Задание № 5.* Укажите этапы трансляции, при которой создается исполняемый файл.

*Задание № 6.* Термины иерархическая, сетевая, реляционная — это

*Задание № 7.* Дана таблица моделирования:

1	моделируемый процесс	A	ракета
2	моделируемый объект	B	исследование траектории
3	цель моделирования	C	полёт ракеты
4	моделируемые характеристики	D	координаты местоположения

Укажите правильный порядок установки соответствия в этой таблице.

*Задание № 8.* Укажите типы вирусных программ.

*Задание № 9.* Укажите классификацию антивирусных программ по принципу работы.

*Задание № 10.* Расшифруйте аббревиатуры LAN, MAN, WAN.

### **Вариант № 7**

*Задание № 1.* На каких видах внешней памяти невозможно случайно стереть информацию?

*Задание № 2.* По какому признаку классифицируются принтеры?

*Задание № 3.* Что обозначает термин «адаптер»?

*Задание № 4.* Укажите, к какому классу языков программирования относится Ассемблер

*Задание № 5.* Укажите язык наиболее удобный для системного программирования

*Задание № 6.* Дана таблица моделирования:

1	моделируемый процесс	А	человек
2	моделируемый объект	В	разработка метода лечения
3	цель моделирования	С	температура и давление
4	моделируемые характеристики	Д	влияние лекарства на состояние больно-гоорганизма

Укажите правильный порядок установки соответствия в этой таблице.

*Задание № 7.* Укажите, какие модели относятся к классу предметных:

1. модель молекулы в виде кристаллической решётки;
2. алгоритм работы станка с числовым программным управлением;
3. макет нефтяной вышки;
4. электрическая схема радиоприемника.

*Задание № 8.* Как называется иерархическая система назначения уникальных текстовых имён каждому компьютеру, находящемуся в сети?

*Задание № 9.* Укажите, какая часть электронного адреса ресурса описывает путь к файлу, расположенному на сервере:

<http://www.google.com/inf02000/01-02/det123.html>

*Задание № 10.* Укажите тип связи, который на сегодня является наиболее защищённым от несанкционированного доступа: оптоволоконный кабель, телефонный кабель (витая пара), электромагнитные волны.

### **Вариант № 8**

*Задание № 1* Укажите устройства, которые не входят в состав внутренней памяти современного компьютера.

*Задание № 2* Укажите, какие из названий относятся к устройствам

**Ввод данных**

1. плоттер;
2. процессор;
3. блок питания;
4. монитор;
5. сканер.

*Задание № 3* Укажите устройства, которые входят в состав внутренней памяти современного компьютера.

*Задание № 4.* Укажите язык наиболее удобный для логического про-

граммирования

1. можно применять разные варианты заливки;
2. при увеличении рисунка может появиться лестничный эффект;
3. можно сохранять рисунок в различных графических форматах;
4. можно использовать пересечение объектов как отдельный графический примитив.

*Задание № 5.* Укажите неверные утверждения для растрового графического редактора:

1. можно рисовать с помощью манипулятора мышью линии произвольной формы;
2. нельзя сохранять рисунки на внешних носителях;
3. нельзя масштабировать фрагменты изображения;
4. возможна тональная коррекция изображения.

*Задание № 6.* Укажите типы моделей в классификации по фактору времени.

*Задание № 7.* Эвристика — это \_\_\_\_\_

*Задание № 8.* Укажите, какие термины не относятся к базовым топологиям сетей: линейная (шинная), кольцевая, снежинка, полносвязанная (сетевая).

*Задание № 9.* Укажите среди приведенных терминов, те, которые обозначают беспроводную связь: Ethernet, Wi-Fi, IrDa, FDDI.

*Задание № 10.* Укажите, что в сетевой терминологии обозначают термины «сервер», «клиент».

### **Вариант № 9**

*Задание №1.* Укажите, какие из названий относятся к устройствам ввода данных:

1. жёсткий диск;
2. мышь;
3. привод CD-ROM;
4. джойстик;
5. регистры.

*Задание № 2.* Укажите устройство хранения данных, которое работает только при включённом питании:

1. ПЗУ;
2. гибкий магнитный диск; 3 ОЗУ;
- 4 жёсткий диск.

*Задание № 3.* Укажите, какие из приведенных утверждений являются верными:

1. сетевая плата не является устройством приёма-передачи данных
2. компакт-диск - это оперативная память
3. гибкий магнитный диск - это долговременная память
4. в мониторах на жидких кристаллах отсутствует электромагнитное излучение

*Задание № 4.* Укажите неверные утверждения для векторного графического редактора:

5. можно применять разные варианты заливки;
6. при увеличении рисунка может появиться лестничный эффект;
7. можно сохранять рисунок в различных графических форматах;
8. можно использовать пересечение объектов как отдельный графический примитив.

*Задание № 5.* Укажите верные утверждения для векторного графического редактора:

1. можно формировать разную заливку одного объекта
2. можно объединять графические объекты
3. нельзя сохранять рисунок на внешнем носителе
4. возможно удаление части графического примитива

*Задание № 6.* Аэродинамическая труба это \_\_\_\_\_

*Задание № 7.* Укажите типы тестирования компьютерных моделей.

*Задание № 8.* Укажите, что в сетевой терминологии обозначает термин «протокол».

*Задание № 9.* Укажите, какая часть электронного адреса ресурса описывает протокол передачи информации:

<http://www.google.com/inf02000/01-02/det123.html>

*Задание № 10.* На сколько уровней разбиты протоколы сетевой модели OSI?

### **Вариант № 10**

*Вопрос № 1.* Укажите наиболее важные характеристики жидкокристаллического монитора:

1. цвет фона окна;
2. физический размер экрана;
3. объём хранимых данных;



4. скорость обработки информации;
5. угол обзора.

*Задание № 2.* Укажите назначение системной шины (магистрالی).

*Задание № 3.* Укажите устройства, размещаемые на материнской плате ПК.

*Задание № 4.* Укажите системы программирования среди перечисленных названий:

1. Adobe PhotoShop
2. Visual FoxPro
3. Visual C++
4. Borland Delphi

*Задание № 5.* Укажите основные особенности трансляции в режиме интерпретации

*Задание № 6.* Укажите отличие модели от реального объекта, явления или процесса.

*Задание № 7.* Процесс описания объекта на искусственном языке называется \_\_\_\_\_

*Задание № 8.* Укажите протоколы сетевой модели OSI, относящиеся к прикладному уровню, и расшифруйте их назначение.

*Задание № 9.* Укажите, на каком уровне сетевой модели OSI устанавливаются правила маршрутизации.

*Задание № 10.* Как называется компьютер с двумя сетевыми картами, предназначенный для соединения сетей.

### **Список использованной литературы:**

1. Макарова Н.В., Волков В.Б. Информатика: Учеб. для вузов. – СПб.: Питер, 2019. – 576 с. (для бакалавров)
2. Информатика: Учеб. пособие / Под ред. Б.Е. Одинцова, А. Н. Романова. 2-е изд., перераб. и доп. – М.: Вузовский учебник; ИНФРА–М, 2017. – 410 с.
3. Гуриков С.Р. Информатика: Учеб. – М.: Форум; ИНФРА– М, 2018. – 464 с. – (Высшее образование. Бакалавриат)
4. Сергеева И.И., Музалевская А.А., Тарасова Н.В. Информатика: Учеб. 2-е изд., перераб. и доп. – М.: Форум; ИНФРА–М, 2019. –384 с. – (Профессиональное образование)

#### **Интернет-источники**

5. Википедия: свободная энциклопедия: <http://ru.wikipedia.org/>
6. Система дистанционного обучения НИУ ИТМО: [http:// de.ifmo.ru/](http://de.ifmo.ru/)
7. Электронный учебник: <http://256bit.ru/informat/>
8. Электронный учебник: <http://book.kbsu.ru/>
9. Информация и методы ее измерения, формулы и т. п.: <http://marklv.narod.ru/inf/izminf.htm>
10. Разные варианты классификации моделей: <http://umk-model.narod.ru/p3.html>
11. Основные понятия информации и других разделов дисциплины «Информатика»: <http://sesia5.ru/blok/1/11-1.htm>
12. Краткое изложение основных разделов информатики: <http://informatika.sch880.ru/p26aa1.html>
13. Разъяснение основных терминов информатики: <http://informatique.org.ru/svoystva-informacii.php>