

# Лекция 9

## Программное обеспечение компьютера

# Вопросы лекции:

- 1. Понятие и классификация программного обеспечения компьютера**
- 2. Системные программные средства**
- 3. Пакеты прикладных программ**
- 4. Системы программирования, языки программирования**

Вся совокупность программ, хранящихся на всех устройствах долговременной памяти компьютера, составляет его *программное обеспечение (ПО)*.



**Прикладное ПО** – программы, с помощью которых пользователь решает свои информационные задачи. К ним относятся текстовые и графические редакторы, табличные процессоры, СУБД, сетевые программы и т.д.

**Системы программирования (СП)** – инструмент для создания программ. Всякая СП ориентирована на один из языков программирования: ПАСКАЛЬ, БЭЙСИК, ФОРТРАН, СИ, АССЕМБЛЕР и др.

Под **программным обеспечением (ПО)** будем понимать совокупность программных средств для ЭВМ и их систем любого класса и типа, обеспечивающих функционирование, диагностику и тестирование их аппаратных средств, а также разработку, отладку и выполнение любых задач пользователя с соответствующим документированием, где в качестве пользователя может выступать как человек, так и любое внешнее устройство, подключенное к ЭВМ и нуждающееся в ее вычислительных ресурсах. Таким образом, **ПО служит интерфейсом между аппаратными ресурсами ЭВМ и проблемной средой.**

# Программное обеспечение

## Системные ПС

Тестовые, диагностические и системные утилиты (Norton Utilities, CheckIt, PowerQuest, Diskeeper и др.)

Операционные системы (ОС)

Файловые менеджеры (Norton Commander, Windows Commander, FAR, Volcov Commander и др.)

Антивирусные программы (Office Scan, DrWeb, AVP, Norton Antivirus, Avast! и др.)

Другие программы (архиваторы, программы резервного копирования и др.)

ОС фирмы Microsoft:  
– DOS;  
– Windows 1.0 — 3.1, 3.11 для рабочих групп;  
– Windows 95, 98, Me;  
– Windows NT 3.51, 4.0;  
– Windows 2000;  
– Windows XP;  
– Windows 2003

Альтернативные ОС:  
– OS/2 (IBM);  
– Mac OS (Apple);  
– Linux, Unix, FreeBSD и др.

## Прикладные ПС

Программные средства управления общими бизнес-процессами

Профессионально-ориентированные программные средства

Программы подготовки текстовых документов

Программы финансового анализа и планирования инвестиций

Программы подготовки табличных документов

Программы управления проектами

Программы подготовки деловой и презентационной графики

Маркетинговые программы

СУБД

Автоматизированные банковские системы

Программы-планировщики

Программы бухгалтерского учёта и аудита

Программы электронной почты и Internet-браузеры

Другие программы (создания и редактирования изображений, переводчики и т. д.)

Интегрированные программные средства

**Системное ПО (СПО)** управляет всеми ресурсами ЭВМ и осуществляет общую организацию процесса обработки информации и интерфейс ЭВМ с проблемной средой, в частности с пользователем. СПО включает в себя операционные системы (ОС), операционные оболочки (О.о.), средства расширения функций ОС и средства тестирования и диагностики ЭВМ.

Кроме ОС к системному ПО относятся множество программ обслуживающего характера: программы обслуживания дисков (копирование, форматирование и т.д.), сжатия файлов (архиваторы), антивирусные программы и др.

**Прикладное ПО (ППО)** составляют **пакеты прикладных программ (ППП)**, предназначенные для решения определенного круга задач из различных проблемных областей, сюда же относится все разработанное многочисленными пользователями ПО. ППП создаются для решения наиболее массовых задач научно-технических, инженерных, экономических и других классов задач; суть большинства ППП состоит в максимальном упрощении интерфейса с ЭВМ проблемного пользователя. Именно на пользователей различного уровня компьютерной образованности и ориентированы многочисленные ППП, которые можно условно классифицировать по четырем основным группам :

- ППП общего назначения;
- проблемно- ориентированные ППП;
- расширяющие функции ОС ППП;

**ППП общего назначения** ориентированы на широкий круг пользователей в различных проблемных областях, позволяя автоматизировать наиболее часто используемые функции и работы.

К пакетам такого типа относятся :

- текстовые процессоры - Ms Word, WordPerfect, ChiWriter, Лексикон
- электронные таблицы - Ms Excel, SuperCalc ;
- системы управления базами данных (СУБД) – Access, dBase, FoxPro, FoxBase, Clipper;
- телекоммуникационные - BBS, Novell Netware и др.

**Проблемно- ориентированные ППП** имеют достаточно узкое применение, используя особые методы представления и обработки информации, учитывающие специфику поддерживаемых задач пользователя. Из пакетов данной группы можно отметить :

- графические - PaintBrush, CorelDraw ;
- математические - **MathCad, Mathematica**;
- статистические - StatGraf, Statistica ;
- экспертные системы - **Guru**;
- издательские системы - Page Maker, Ventura Publisher, Adobe Illustrator;
- разработки систем управления (САСЕ -группа);
- проектирования систем управления (САСДС -группа);
- проектирования и изготовления чертежей (САД -группа) - PCAD, OrCAD, AutoCAD ;
- планирования (САР -группа).

**Расширяющие функции ОС ППП** составляют пакеты,

- обеспечивающие сопряжение ЭВМ с научными приборами и установками;
- обеспечивающие подключение к ЭВМ дополнительных внешних устройств, поддержку работы ЭВМ в локальных сетях и машинных комплексах;
- для расширения функций внешних устройств ЭВМ (монитора, клавиатуры, мыши и др.).

**Интегрированные ППП** объединяют основные функции ряда пакетов, как правило, общего назначения. В таких пакетах используются не только единый формат файлов, но и методы доступа к файлам отдельных пакетов, т.е. обеспечивается полная информационная совместимость составляющих совокупность пакетов. Типичным средством данного класса является известный пакет Ms Office , включающий текстовый процессор Ms Word, табличный процессор Ms Excel , графическое средство Power Point, СУБД MS Access и др.

Основное назначение программных продуктов мультимедиа - создание и использование аудио- и видеоинформации для расширения информационного пространства пользователя.

Класс продуктов мультимедиа нашел свое приложение в сфере библиотечного информационного обслуживания, процессе обучения, организации досуга. Базы данных компьютерных изображений произведений искусств, библиотеки звуковых записей составляют основу прикладных обучающих систем, компьютерных игр, библиотечных каталогов и фондов.

Главной частью *системного программного обеспечения* является *операционная система (ОС)*.

*Операционная система* – это набор программ, управляющих оперативной памятью, процессором, внешними устройствами и файлами, ведущих диалог с пользователем (примеры: MS-DOS, WINDOWS). Операционная система обеспечивает совместное функционирование всех устройств компьютера и предоставляет пользователю доступ к его ресурсам. ОС является базовой и необходимой составляющей программного обеспечения компьютера, без нее компьютер не может работать в принципе.

С одной стороны, ОС опирается на базовое программное обеспечение компьютера, входящее в его систему *BIOS (базовая система ввода-вывода)*; с другой стороны, она сама является опорой для программного обеспечения более высоких уровней — прикладных и большинства служебных приложений. *Приложениями операционной системы* принято называть программы, предназначенные для работы под управлением данной системы.

В *графических операционных системах* принцип управления приложениями состоит во взаимодействии активных и пассивных элементов управления. **Активный элемент управления** — *указатель мыши* (его предоставляет операционная система). **Пассивные элементы управления** — *графические кнопки, поля, флажки, переключатели, меню, списки* и прочие. Их предоставляют конкретные приложения. В момент взаимодействия активного и пассивного элементов управления пользователь выдает управляющие сигналы с помощью органов управления графического манипулятора.

В *неграфических операционных системах* управление приложениями ограничено и осуществляется путем ручного ввода текстовых команд в поле командной строки. Органом управления в данном случае является клавиатура.

**Основная функция всех операционных систем** — посредническая.

Она заключается в обеспечении нескольких видов интерфейса:

- интерфейса между пользователем и программно-аппаратными средствами компьютера (*интерфейс пользователя*);
- интерфейса между программным и аппаратным обеспечением (*аппаратно-программный интерфейс*);
- интерфейса между разными видами программного обеспечения (*программный интерфейс*).

Основные достоинства персональной вычислительной техники проявляются в *диалоговом режиме* работы с пользователем. Диалоговый режим отличается от *пакетного* тем, что в ходе работы процессор регулярно приостанавливает выполнение текущих задач и обращается к другим устройствам и к программам, проверяя их состояние. Если пользователь использует какое-либо средство управления или извне поступает управляющий сигнал, процессор устанавливает этот факт и реагирует на него переходом на исполнение другой программы. Несмотря на то, что в любой момент времени процессор работает по жестко заданным программам, динамичное переключение между ними создает впечатление гибкого управления работой компьютера.

Организацией работы процессора в диалоговом режиме ведаёт относительно небольшая группа системных программ. Она образует *ядро операционной системы*.

Дополнительно к ядру операционная система обладает средствами для:

- управления пользовательским интерфейсом компьютера;
- управления аппаратно-программными интерфейсами компьютера;
- обслуживания файловой системы;
- управления распределением оперативной памяти между процессами;
- установки программ и управления их работой;
- обеспечения надёжности и устойчивости работы оборудования и программ.

Чем **шире** функциональные возможности операционной системы, тем большие требования она предъявляет к техническим ресурсам компьютерной системы, но тем проще работа с компьютером с точки зрения пользователя.

Кроме всего вышеперечисленного, современные операционные системы могут включать минимальный набор прикладного программного обеспечения, которое можно использовать для исполнения простейших практических задач:

- чтение, редактирование и печать текстовых документов;
- создание и редактирование простейших рисунков;
- выполнение арифметических и математических расчетов;
- ведение дневников и служебных блокнотов;
- создание, передача и прием сообщений электронной почты;
- создание и редактирование факсимильных сообщений;
- воспроизведение и редактирование звукозаписи;
- воспроизведение видеозаписи;
- разработка и воспроизведение комплексных электронных документов, включающих текст, графику, звукозапись и видеозапись.

Этим возможности операционных систем не исчерпываются. По мере развития аппаратных средств вычислительной техники и средств связи функции операционных систем непрерывно расширяются, а средства их исполнения совершенствуются.

**Программа** — это логически упорядоченная последовательность команд, необходимых для управления компьютером (выполнения им конкретных операций), поэтому программирование сводится к созданию последовательности команд, необходимой для решения определенной задачи.

Программа, с которой работает процессор, представляет собой последовательность чисел, называемую **машинным кодом**.

Управлять компьютером нужно по определенному алгоритму.

**Алгоритм** — это точно определенное описание способа решения задачи в виде конечной (по времени) последовательности действий. Такое описание еще называется **формальным**. Для представления алгоритма в виде, понятном компьютеру, служат **языки программирования**. Сначала всегда разрабатывается алгоритм действий, а потом он записывается на одном из таких языков. В итоге получается **текст программы** — полное, законченное и детальное описание алгоритма на языке программирования. Затем этот текст программы специальными служебными приложениями, которые называются **трансляторами**, либо переводится в машинный код, либо исполняется

Сегодня практически все программы создаются с помощью языков программирования. Теоретически программу можно написать и средствами обычного человеческого (естественного) языка — это называется программированием на **метаязыке** (подобный подход обычно используется на этапе составления алгоритма), но автоматически перевести такую программу в машинный код пока невозможно из-за высокой неоднозначности естественного языка.

**Языки программирования** — **искусственные языки**. От естественных они отличаются ограниченным числом «слов», значение которых понятно транслятору, и очень строгими правилами записи команд (**операторов**). Совокупность подобных требований образует **синтаксис языка программирования**, а **смысл** каждой команды и других конструкций языка — его **семантику**. Нарушение формы записи программы приводит к тому, что транслятор не может понять назначение оператора и выдает сообщение о синтаксической ошибке, а правильно написанное, но не отвечающее алгоритму использование команд языка приводит к семантическим ошибкам (называемым еще **логическими ошибками** или **ошибками времени выполнения**).

Процесс поиска ошибок в программе называется **тестированием**,

процесс устранения ошибок — **отладкой**.

С помощью языка программирования создается не готовая программа, а только ее текст, описывающий ранее разработанный алгоритм. Чтобы получить работающую программу, надо этот текст либо автоматически перевести в машинный код (для этого служат **программы-компиляторы**) и затем использовать отдельно от исходного текста, либо сразу выполнять команды языка, указанные в тексте программы (этим занимаются **программы-интерпретаторы**).

**Разные типы процессоров имеют разные наборы команд.** Если язык программирования ориентирован на конкретный тип процессора и учитывает его особенности, то он называется **языком программирования низкого уровня**. В данном случае «низкий уровень» не значит «плохой». Имеется в виду, что операторы языка близки к **машинному** коду и ориентированы на конкретные команды процессора.

**Языком самого низкого уровня является язык ассемблера, который просто представляет каждую команду машинного кода, но не в виде чисел, а с помощью символьных условных обозначений, называемых мнемониками.** Однозначное преобразование одной машинной инструкции в одну команду ассемблера называется **транслитерацией**. Так как наборы инструкций для каждой модели процессора отличаются, конкретной компьютерной архитектуре соответствует свой язык ассемблера, и написанная на нем программа может быть

***Языки программирования высокого уровня*** значительно ближе и понятнее человеку, нежели компьютеру. Особенности конкретных компьютерных архитектур в них не учитываются, поэтому создаваемые программы на уровне исходных текстов легко **переносимы** на другие платформы, для которых создан транслятор этого языка. Разрабатывать программы на языках высокого уровня с помощью понятных и мощных команд значительно проще, а ошибок при создании программ допускается гораздо меньше.

***Языки программирования баз данных*** отличается от алгоритмических языков прежде всего решаемыми задачами.

**База данных** — это файл (или группа файлов), представляющий собой упорядоченный набор записей, имеющих единообразную структуру и организованных по единому шаблону (как правило, в табличном виде). База данных может состоять из нескольких таблиц. Удобно хранить в базах данных различные сведения из справочников, картотек, журналов бухгалтерского учета и т. д.

При работе с базами данных чаще всего требуется выполнять следующие операции:

- создание/модификация свойств/удаление таблиц в базе данных;
- поиск, отбор, сортировка информации по запросам пользователей;
- добавление новых записей;
- модификация существующих записей;
- удаление существующих записей.

Первые базы данных появились очень давно, как только появилась потребность в обработке больших массивов информации и выборки групп записей по определенным признакам. Для этого был создан **структурированный язык запросов SQL (Structured Query Language)**. Он основан на мощной математической теории и позволяет выполнять эффективную обработку баз данных, манипулируя не отдельными записями, а *группами* записей.

Для управления большими базами данных и их эффективной обработки разработаны **СУБД** (Системы Управления Базами Данных).

Практически в каждой СУБД помимо поддержки языка *SQL* имеется также свой уникальный язык, ориентированный на особенности этой СУБД и не переносимый на другие системы. Сегодня в мире насчитывается три ведущих производителя СУБД: *Microsoft (SQL Server)*, *IBM (DB2)* и *Oracle*.

С появлением персональных компьютеров были созданы так называемые настольные СУБД. Родоначальником современных языков программирования баз данных для ПК принято считать СУБД *dBase II*, язык которой был интерпретируемым. Затем для него были созданы компиляторы, появились СУБД *FoxPro* и *Clipper*, поддерживающие диалекты этого языка. Сегодня самой распространенной настольной СУБД стала система *Microsoft Access*.

С активным развитием глобальной сети было создано немало реализаций популярных языков программирования, адаптированных специально для Интернета. Все они отличаются характерными особенностями: языки являются интерпретируемыми, интерпретаторы для них распространяются бесплатно, а сами программы — в исходных текстах. Такие языки называют **скрипт-языками** (*HTML, Perl, PHP, Tcl/Tk, VRML, XML*).

## Для создания программы необходимы:

- текстовый редактор;
- компилятор;
- редактор связей;
- библиотеки функций.

Как правило, в стандартную поставку входят как минимум три последних компонента, но хорошая *интегрированная система* включает в себя и *специализированный текстовый редактор*, причем почти все этапы создания программы в ней автоматизированы; после того как исходный текст введен, его компиляция и сборка выполняются одним нажатием клавиши. Это очень удобно, так как не требует ручной настройки множества параметров запуска компилятора и редактора связей, указания им нужных файлов вручную и т.д. Процесс компиляции обычно демонстрируется на экране: показывается, сколько строк исходного текста откомпилировано, или выдаются сообщения о найденных ошибках.

В современных интегрированных системах имеется еще один компонент — *отладчик*, который позволяет анализировать работу программы во время ее выполнения. С его помощью можно последовательно выполнять отдельные операторы исходного текста *по шагам*, наблюдая при этом, как меняются значения различных переменных.

Из универсальных языков программирования сегодня наиболее популярны следующие:

**Бейсик** (*Basic*) — для освоения требует начальной подготовки (общеобразовательная школа);

**Паскаль** (*Pascal*) — требует специальной подготовки (школы с углубленным изучением предмета и общетехнические вузы);

**Си++** (*C++*), **Ява** (*Java*), **Си Шарп** (*C#*) — требуют профессиональной подготовки (специализированные средние и высшие учебные заведения).

Для каждого из этих языков программирования сегодня имеется немало систем программирования, выпускаемых различными фирмами и ориентированных на различные модели ПК и операционные системы. Наиболее популярны следующие **визуальные среды быстрого проектирования программ для Windows**:

- *Basic: Microsoft Visual Basic;*
- *Pascal; Borland Delphi;*
- *C++: Microsoft Visual C++;*
- *Java: Borland J Builder,*
- *C#: Microsoft Visual Studio .NET, Borland CttBuilder.*

Для разработки серверных и распределенных приложений можно использовать систему программирования *Microsoft Visual C++*, продукты фирмы *Borland* практически любые средства