

Поколения компьютеров

Поколение	Годы
1	1930 - 1950-е
2	1960 - 1970-е
3	1970 - 1980-е
4	Вторая половина 70 - начало 90-х
5	90-е - наше время
6	В разработке

Поколение	Технологическая база	Тактовая частота выполнения операций
1	Вакуумные лампы	Несколько килогерц
2	Полупроводники	Сотни КГц
3	Интегральные схемы	Мегагерцы
4	Большие и сверхбольшие схемы	Десятки МГц
5	Параллельно-векторные технологии	Сотни МГц, Гигагерцы
6	Нейронные принципы	Критерии измерения прорабатываются

Что такое программное обеспечение?

Под программным обеспечением (Software) понимается совокупность программ, выполняемых вычислительной системой.

К программному обеспечению (ПО) относится также вся область деятельности по проектированию и разработке ПО:

- технология проектирования программ (например, нисходящее проектирование, структурное и объектно-ориентированное проектирование и др.);
- методы тестирования программ;
- методы доказательства правильности программ;
- анализ качества работы программ;
- документирование программ; разработка и использование программных средств, облегчающих процесс проектирования программного обеспечения, и многое другое.

Сам по себе компьютер не обладает знаниями ни в одной области применения.

Все эти знания сосредоточены в выполняемых на компьютерах программах.

Программное обеспечение — неотъемлемая часть компьютерной системы. Оно является логическим продолжением технических средств. Сфера применения конкретного компьютера определяется созданным для него ПО.

Программное обеспечение современных компьютеров включает миллионы программ — от игровых до научных.

Как классифицируется программное обеспечение?

Все программы, работающие на компьютере, можно условно разделить на **три категории**:

прикладные программы, непосредственно обеспечивающие выполнение необходимых пользователям работ;

системные программы, выполняющие различные вспомогательные функции, например:

- управление ресурсами компьютера;
- создание копий используемой информации;
- проверка работоспособности устройств компьютера;
- выдача справочной информации о компьютере и др.;

инструментальные программные системы (системы программирования), облегчающие процесс создания новых программ для компьютера.



Сложились следующие группы программного обеспечения:

- операционные системы и оболочки;
- системы программирования (трансляторы, библиотеки подпрограмм, отладчики и т.д.);
- инструментальные системы;
- интегрированные пакеты программ;
- динамические электронные таблицы;
- системы машинной графики;
- системы управления базами данных (СУБД);
- прикладное программное обеспечение.

Эту классификацию нельзя считать исчерпывающей, но она более или менее наглядно отражает направления совершенствования и развития программ.

СТРУКТУРА ПРОГРАММНОГО ОБЕСПЕЧЕНИЯ ПК

Системное ПО

Прикладное ПО

Системы программирования

Прикладное ПО – это программы, с помощью которых пользователь имеет возможность решать свои информационные задачи, не прибегая к программированию.

Системное программное обеспечение – обязательная часть ПО. Его ядро составляет **операционная система (ОС)**.

Операционная система (ОС)

Диалоговые оболочки

Сервисные программы

Прикладное ПО общего назначения

Пакет прикладных программ MS Office, компьютерные игры

Прикладное ПО специального назначения

- бухгалтерские программы
- математические пакеты
- экспертные системы
- обучающие программы
- конструкторские пакеты
- графические пакеты

Системы программирования – это инструменты для работы программистов, ориентированные на определенный язык программирования.

Язык программирования – это фиксированная система обозначений для описания алгоритмов и структур данных.

Существует множество языков программирования разного уровня и назначения:

- Ассемблеры
- ФОРТРАН
- Паскаль
- С (Си)
- ЛОГО
- ЛИСП
- Пролог
- Делфи
- Бейсик
-

Системные программы выполняются вместе с прикладными и служат для управления ресурсами компьютера — центральным процессором, памятью, вводом-выводом.

Это программы общего пользования, которые **предназначены для всех пользователей компьютера**. Системное программное обеспечение разрабатывается так, чтобы компьютер мог эффективно выполнять прикладные программы.

операционные системы, обеспечивают управление ресурсами компьютера с целью их эффективного использования.

утилиты (лат. *utilitas* — польза), они либо **расширяют и дополняют соответствующие возможности операционной системы**, либо **решают самостоятельные важные задачи**.

Некоторые классы утилит:

программы контроля, тестирования и диагностики, которые используются для проверки правильности функционирования устройств компьютера и для обнаружения неисправностей в процессе эксплуатации; указывают причину и место неисправности;

программы-драйверы, которые расширяют возможности операционной системы по управлению устройствами ввода-вывода, оперативной памятью и т.д.; с помощью драйверов возможно подключение к компьютеру новых устройств или нестандартное использование имеющихся;

программы-упаковщики (архиваторы).

Что такое операционная система?

Операционная система — это комплекс взаимосвязанных системных программ, назначение которого — организовать взаимодействие пользователя с компьютером и выполнение всех других программ.

Операционная система выполняет роль связующего звена между аппаратурой компьютера, с одной стороны, и выполняемыми программами, а также пользователем, с другой стороны.

Операционная система обычно хранится во внешней памяти компьютера — *на диске*. При включении компьютера она считывается с дисковой памяти и размещается в *ОЗУ*.

Этот процесс называется *загрузкой операционной системы*.

В функции операционной системы входит:

- осуществление диалога с пользователем;
- ввод-вывод и управление данными;
- планирование и организация процесса обработки программ;
- распределение ресурсов (оперативной памяти и кэша, процессора, внешних устройств);
- запуск программ на выполнение;
- всевозможные вспомогательные операции обслуживания;
- передача информации между различными внутренними устройствами;
- программная поддержка работы периферийных устройств (дисплея, клавиатуры, дисковых накопителей, принтера и др.).

Операционную систему можно назвать программным продолжением устройства управления компьютера. Операционная система скрывает от пользователя сложные ненужные подробности взаимодействия с аппаратурой, образуя прослойку между ними. В результате этого люди освобождаются от очень трудоёмкой работы по организации взаимодействия с аппаратурой компьютера.

В зависимости от количества одновременно обрабатываемых задач и числа пользователей, которых могут обслуживать ОС, различают четыре основных класса операционных систем:

- 1. однопользовательские однозадачные**, которые поддерживают одну клавиатуру и могут работать только с одной (в данный момент) задачей;
- 2. однопользовательские однозадачные с фоновой печатью**, которые позволяют помимо основной задачи запускать одну дополнительную задачу, ориентированную, как правило, на вывод информации на печать. Это ускоряет работу при выдаче больших объёмов информации на печать;
- 3. однопользовательские многозадачные**, которые обеспечивают одному пользователю параллельную обработку нескольких задач. Например, к одному компьютеру можно подключить несколько принтеров, каждый из которых будет работать на "свою" задачу;
- 4. многопользовательские многозадачные**, позволяющие на одном компьютере запускать несколько задач нескольким пользователям. Эти ОС очень сложны и требуют значительных машинных ресурсов.

В различных моделях компьютеров используют операционные системы с разной архитектурой и возможностями. Для их работы требуются разные ресурсы. Они предоставляют разную степень сервиса для программирования и работы с готовыми программами.

В различных моделях компьютеров используют операционные системы с разной архитектурой и возможностями. Для их работы требуются разные ресурсы. Они предоставляют разную степень сервиса для программирования и работы с готовыми программами.

Операционная система для персонального компьютера, ориентированного на профессиональное применение, должна содержать следующие основные компоненты:

- программы управления вводом/выводом;
- программы, управляющие файловой системой и планирующие задания для компьютера;
- процессор командного языка, который принимает, анализирует и выполняет команды, адресованные операционной системе.

Каждая операционная система имеет свой **командный язык**, который позволяет пользователю выполнять те или иные действия:

- обращаться к каталогу;
- выполнять разметку внешних носителей;
- запускать программы;
- ... другие действия.

Анализ и исполнение команд пользователя, включая загрузку готовых программ из файлов в оперативную память и их запуск, осуществляет **командный процессор** операционной системы.

Для управления внешними устройствами компьютера используются специальные системные программы — **драйверы**. Драйверы стандартных устройств образуют в совокупности **базовую систему ввода-вывода** (BIOS), которая обычно заносится в постоянное ЗУ компьютера.

Что такое файловая система ОС?

Файл (англ. *file* — папка) — это именованная совокупность любых данных, размещенная на внешнем запоминающем устройстве и хранимая, пересылаемая и обрабатываемая как единое целое. Файл может содержать программу, числовые данные, текст, закодированное изображение и др.

Файловая система — это средство для организации хранения файлов на каком-либо носителе.

Каждый файл занимает некоторое количество блоков дисковой памяти. Обычная длина блока — 512 байт.

Обслуживает файлы специальный модуль операционной системы, называемый **драйвером файловой системы**. Каждый файл имеет имя, зарегистрированное в **каталоге** — оглавлении файлов.

Каталог (иногда называется ***директорией*** или ***папкой***) доступен пользователю через командный язык операционной системы. Его можно просматривать, переименовывать зарегистрированные в нем файлы, переносить их содержимое на новое место и удалять. Каталог может иметь собственное имя и храниться в другом каталоге наряду с обычными файлами: так образуются иерархические файловые структуры.

Папки

- Локальный диск (C:)
 - Documents and Settings
 - 1u11
 - 1u11.EDUDOMAIN
 - Application Data
 - Cookies
 - Local Settings
 - NetHood
 - PrintHood
 - Recent
 - SendTo
 - Главное меню
 - Избранное
 - Мои документы
 - Мои рисунки
 - Мои Сайты
 - _private
 - _vti_cnf
 - _vti_pvt
 - images
 - komp**
 - _borders
 - _derived

Что происходит, когда пользователь подает операционной системе команду "**открыть файл ...**", в которой указано *имя файла* и *имя каталога*, в котором размещён этот файл?

Для выполнения этой команды драйвер файловой системы обратится к своему **справочнику**, выясняет, какие блоки диска соответствуют указанному файлу, а затем передает запрос на считывание этих блоков драйверу диска.

При выполнении команды "**сохранить файл**" драйвер файловой системы ищет на диске незанятые блоки, отмечает их, как распределённые для вновь созданного файла, и передаёт драйверу диска запрос на запись в эти блоки данных пользователя.

Драйвер файловой системы обеспечивает доступ к информации, записанной на магнитный диск, по имени файла и распределяет пространство на магнитном диске между файлами.

Для выполнения этих функций драйвер файловой системы хранит на диске не только информацию пользователя, но и свою собственную служебную информацию. **В служебных областях диска хранится список всех файлов и каталогов**, а также различные дополнительные справочные таблицы, служащие для повышения скорости работы драйвера файловой системы.

К файловой системе имеет доступ также и любая прикладная программа, для чего во всех языках программирования имеются специальные процедуры.

Понятие файла может быть обращено на любой источник или потребитель информации в машине, например, в качестве файла для программы могут выступать *принтер, дисплей, клавиатура* и др.

Структура файловой системы и структура хранения данных на внешних магнитных носителях определяет удобство работы пользователя, скорость доступа к файлам и т.д.

Что такое транслятор, компилятор, интерпретатор?

Транслятор (англ. *translator* — переводчик) — это программа-переводчик. Она преобразует программу, написанную на одном из языков высокого уровня, в программу, состоящую из машинных команд.

Трансляторы реализуются в виде компиляторов или интерпретаторов. С точки зрения выполнения работы компилятор и интерпретатор существенно различаются.

Компилятор (англ. *compiler* — составитель, собиратель) читает всю программу *целиком*, делает ее перевод и создает законченный вариант программы на машинном языке, который затем и выполняется.

Интерпретатор (англ. *interpreter* — истолкователь, устный переводчик) переводит и выполняет программу *строка за строкой*.

После того, как программа откомпилирована, ни сама исходная программа, ни компилятор более не нужны. В то же время программа, обрабатываемая интерпретатором, должна заново *переводиться* на машинный язык при каждом очередном запуске программы.

Откомпилированные программы работают быстрее, но интерпретируемые проще исправлять и изменять.

Система программирования — это система для разработки новых программ на конкретном языке программирования.

Современные системы программирования обычно предоставляют пользователям **мощные и удобные средства разработки программ**. В них входят:

- компилятор или интерпретатор;
- интегрированная среда разработки;
- средства создания и редактирования текстов программ;
- обширные библиотеки стандартных программ и функций;
- отладочные программы, т.е. программы, помогающие находить и устранять ошибки в программе;
- "дружественная" к пользователю диалоговая среда;
- многооконный режим работы;
- мощные графические библиотеки; утилиты для работы с библиотеками
- встроенный ассемблер;
- встроенная справочная служба;
- другие специфические особенности.

Популярные системы программирования — *Visual Basic for Applications, Visual C++, Delphi*.

Библиотека стандартных подпрограмм — это совокупность подпрограмм, составленных на одном из языков программирования и удовлетворяющих определенным единым требованиям к структуре, организации их входов и выходов, описаниям подпрограмм и т.п.

Стандартные подпрограммы имеют единую форму обращения, что обеспечивает простоту и удобство настройки параметров подпрограммы на решение конкретной задачи.

В качестве примера можно привести библиотеку стандартных подпрограмм по численным математическим методам решения уравнений, вычисления интегралов, нахождения экстремумов и т.п.

Инструментальные программные средства — это программы, которые используются в ходе разработки, корректировки или развития других прикладных или системных программ.

По своему назначению они близки системам программирования. К инструментальным программам, например, относятся:

- редакторы;
- средства компоновки программ;
- отладочные программы, т.е. программы, помогающие находить и устранять ошибки в программе;
- вспомогательные программы, реализующие часто используемые системные действия;
- графические пакеты программ и т.п.

Инструментальные программные средства могут оказать помощь на всех стадиях разработки ПО.

Прикладная программа — это любая конкретная программа, способствующая решению какой-либо задачи в пределах данной проблемной области.

Например, там, где на компьютер возложена задача контроля за финансовой деятельностью какой-либо фирмы, прикладной будет программа подготовки платежных ведомостей.

Прикладные программы могут носить и общий характер, например, обеспечивать составление и печатание документов и т.п.

В противоположность этому, операционная система или инструментальное ПО не вносят прямого вклада в удовлетворение конечных потребностей пользователя.

Прикладные программы могут использоваться либо автономно, то есть решать поставленную задачу без помощи других программ, либо в составе программных комплексов или пакетов.

Пакеты прикладных программ (ППП) — это специальным образом организованные программные комплексы, рассчитанные на общее применение в определенной проблемной области и дополненные соответствующей технической документацией.

В зависимости от характера решаемых задач различают следующие разновидности ППП:

- пакеты для решения типовых инженерных, планово-экономических, общенаучных задач;
- пакеты системных программ;
- пакеты для обеспечения систем автоматизированного проектирования и систем автоматизации научных исследований;
- пакеты педагогических программных средств и другие.

Чтобы пользователь мог применить ППП для решения конкретной задачи, пакет должен обладать средствами настройки (иногда путём введения некоторых дополнений).

Каждый ППП обладает обычно рядом возможностей по методам обработки данных и формам их представления, полноте диагностики, что дает возможность пользователю выбрать подходящий для конкретных условий вариант.

ППП обеспечивают значительное снижение требований к уровню профессиональной подготовки пользователей в области программирования, вплоть до возможности эксплуатации пакета без программиста.

Часто пакеты прикладных программ располагают базами данных для хранения данных и передачи их прикладным программам.

Что такое системы управления базами данных?

В базе данных предприятия, например, может храниться:

- вся информация о штатном расписании, о рабочих и служащих предприятия;
- сведения о материальных ценностях;
- данные о поступлении сырья и комплектующих;
- сведения о запасах на складах;
- данные о выпуске готовой продукции;
- приказы и распоряжения дирекции и т.п.

Даже небольшие изменения какой-либо информации могут приводить к значительным изменениям в разных других местах.

Пример. Издание приказа о повышении в должности одного работника приводит к изменениям не только в личном деле работника, но и к изменениям в списках подразделения, в котором он работает, в ведомостях на зарплату, в графике отпусков и т. п. Базы данных используются под управлением *систем управления базами данных (СУБД)*.

Система управления базами данных (СУБД) — это система программного обеспечения, позволяющая обрабатывать обращения к базе данных, поступающие от прикладных программ конечных пользователей.

Системы управления базами данных позволяют *объединять* большие объемы информации и *обрабатывать* их, *сортировать*, *делать выборки* по определённым критериям и т.п.

Современные СУБД дают возможность включать в них не только *текстовую* и *графическую* информацию, но и звуковые фрагменты и даже видеоклипы.

Простота использования СУБД позволяет создавать новые базы данных, не прибегая к программированию, а пользуясь только встроенными функциями.

СУБД обеспечивают *правильность*, *полноту* и *непротиворечивость* данных, а также *удобный доступ* к ним.

Примеры СУБД — *dBase, FoxPro, MS Access, Paradox.*

Для менее сложных применений вместо СУБД используются **информационно-поисковые системы** (ИПС), которые выполняют следующие функции:

- *хранение* большого объема информации;
- *быстрый поиск* требуемой информации;
- *добавление, удаление* и *изменение* хранимой информации;
- *вывод* ее в удобном для человека виде.

Интегрированные пакеты представляют собой набор нескольких программных продуктов, объединенных в единый удобный инструмент. Наиболее развитые из них включают в себя текстовый редактор, органайзер, электронную таблицу, СУБД, средства поддержки электронной почты, программу создания презентационной графики.

Интегрированные пакеты, как правило, содержат некоторое ядро, обеспечивающее возможность тесного взаимодействия между составляющими.

Пример: **интегрированный пакет для написания книг, содержащих иллюстрации.**
Он содержит:

- текстовый редактор;
- орфографический корректор на 80000 слов (программу обнаружения орфографических ошибок);
- программу слияния текстов;
- программу формирования оглавлений и составления указателей;
- автоматический поиск и замену слов и фраз;
- средства телекоммуникации;
- электронную таблицу;
- систему управления базами данных;
- модули графического оформления;
- графический редактор;
- возможность печати сотнями разных шрифтов и т.д.