

Оглавление

Лекция 1.	2
Информатика и информация	2
Представление данных в компьютере	2
Бит	2
Байт	3
Килобайт	3
История развития информатики и вычислительной техники	3
Поколение первое.	3
Поколение второе.	4
Поколение третье.	4
Поколение четвертое.	5
Компьютеры пятого поколения	5
Машина фон Неймана.	6
Микропроцессор (МП) или CPU (Central Processing Unit).	7
Характеристики процессора:	8
Интерфейсная система - это:	8
Память	9
Контроллеры	10
Устройства вывода.	10
Устройства ввода	10
Микросхема ПЗУ и система BIOS	11
Лекция 2	11
Операционная система.	11
Из истории Windows	12
MS-DOS	12
Графические интерфейсы и расширения для DOS	13
ОС Windows	13
Семейство ОС Windows Mobile	13
ОС Linux.	14
Языки программирования	14
Машинный код процессора	14
Алгоритм и программа	14
Что такое язык программирования	14
Трансляторы	15
Уровни языков программирования	15
Способы изображения алгоритмов. Блок-схемы.	16
Схемы основных вычислительных процессов	16
Технологии программирования.	18
Структурное программирование.	18
Объектно-ориентированное программирование	20
Лекция 3	21
Программное обеспечение компьютеров	21
Системное ПО.	21
Инструментальное ПО.	22
Прикладные программы.	22
Классификация ПО по условиям распространения и использования.	24
Commercial software	24
FreeWare	24
Free Software Definition	24
ShareWare	24
Trial	24
Demo	24
Adware	24
Donationware	24
Postcardware	24
Классификация вредоносных программ	24
Классификация вредоносных программ по версии Microsoft:	25
Классификация вредоносных программ по версии Лаборатории Касперского (опубликована в вирусной энциклопедии):	25
1. Сетевые черви	25
2. Классические компьютерные вирусы	25
3. Троянские программы	27
4. Хакерские утилиты и прочие вредоносные программы	27
Антивирусная программа	28
Классификация антивирусов	28
Сканеры	28
Ревизоры	28
Вакцины	28
Часто используемые Антивирусные программы:	28

Лекция 1.

Информатика и информация

Информация в современном мире приравнивается по своему значению для развития общества или страны к важнейшим ресурсам наряду с сырьем и энергией. Еще в 1971 году президент Академии наук США Ф.Хандлер говорил: "Наша экономика основана не на естественных ресурсах, а на умах и применении научного знания".

В развитых странах большинство работающих заняты не в сфере производства, а в той или иной степени занимаются обработкой информации. Поэтому философы называют нашу эпоху постиндустриальной. В 1983 году американский сенатор Г.Харт охарактеризовал этот процесс так: «Мы переходим от экономики, основанной на тяжелой промышленности, к экономике, которая все больше ориентируется на информацию, новейшую технику и технологию, средства связи и услуги».

Понятие «информация»

В русском языке до второй половины XX века термин и понятие «информация» практически не использовались

А. С. Пушкин: «сведения», «замечания», «известия», «наблюдения»

Л. Н. Толстой: «донесения», «письма», «записочка», «memorandum»

В энциклопедии Брокгауза и Ефрона (1894 г.) – нет

В Советской энциклопедии (1 издание, 1935 г.) – нет

Н. Винер, К. Шеннон (1948 г.) – количество информации

А. П. Ершов: «Данные в информатике – факты или идеи, выраженные средствами формальной системы, обеспечивающей возможности их хранения, обработки или передачи. Такую формальную систему называют языком представления данных; синтаксис этого языка – способом представления информации; его семантику или прагматику – информатикой».

Указанное соотношение терминов «данные» и «информация» рекомендовано большинством терминологических справочников, но на практике они обычно трактуются как **синонимы**; да и сама **информатика занимается не столько информацией, сколько данными**».

Для человека существенно прежде всего содержание, смысл сообщения, сигнала, а для компьютера смысловой аспект безразличен. Смысловой аспект сообщения – это и есть **информация**. Все технические системы имеют дело с **данными**.

Данные не имеют смысла (М. Мамардашвили). Смысл генерируется человеком на основе, полученных данных, и сформированного ранее смысла (информации).

Если математика – это язык наук, то информатика – это инструмент наук.

Информация - это данные, сопровождающиеся смысловой нагрузкой. При этом, очевидно, то, что для одних является данными, для других вполне может быть информацией. Но всегда можно точно сказать, что нужно предпринять для того, чтобы те или иные данные стали информативными для наибольшей аудитории: их нужно снабдить смысловым содержанием. Чем более полным будет это содержание, тем более информативной будет соответствующее сообщение.

Представление данных в компьютере

Бит

Бит (от англ. binary digit; также игра слов: англ. bit — немного) .

- По Шеннону бит — это двоичный логарифм вероятности равновероятных событий или сумма произведений вероятности на двоичный логарифм вероятности при равновероятных событиях. (Замена основания логарифма 2 на e или 10 приводит соответственно к редко употребляемым единицам нат и хартли.)
- Один разряд двоичного кода (двоичная цифра). Может принимать только два взаимоисключающих значения: да/нет, 1/0, включено/выключено, и т. п.
- Базовая единица измерения количества информации, равная количеству информации, содержащемуся в опыте, имеющем два равновероятных исхода. Это тождественно количеству информации в ответе на вопрос, допускающий ответы «да» либо «нет» и никакого другого (то есть такое количество информации, которое позволяет однозначно ответить на поставленный вопрос). В одном двоичном разряде содержится один бит информации.

В вычислительной технике и сетях передачи данных обычно значения 0 и 1 передаются различными уровнями напряжения либо тока.

В вычислительной технике, особенно в документации и стандартах, слово «бит» часто применяется в значении двоичный разряд. Например: первый бит — первый двоичный разряд байта или слова о котором идет речь.

В настоящее время бит — это наименьшая возможная единица измерения информации в вычислительной технике. Аналогом бита в квантовых компьютерах является q-бит.

Байт

Для представления символа с максимально возможным кодом (255) нужно 8 бит. Эти 8 бит называются байтом. Т.о. один любой символ - это всегда 1 байт.

Байт (англ. byte) — единица измерения количества информации, обычно равная восьми битам (в этом случае может принимать 256 (28) различных значений).

Вообще, байт — это минимально адресуемая последовательность фиксированного числа битов. В современных компьютерах общего назначения байт равен 8 битам. Для того, чтобы подчеркнуть, что имеется в виду восьмибитный байт, в описании сетевых протоколов используется термин «октет» (лат. octet).

Килобайт

Килобайт (Кбайт, КБ) — единица измерения количества информации, равная 210 стандартным (8-битным) байтам или 1024 байтам. Применяется для указания объёма памяти в различных электронных устройствах.

Название «килобайт» общепринято, но формально неверно, так как приставка кило-, означает умножение на 1 000, а не 1 024. Правильной для 210 является двоичная приставка киби-

Измерения в байтах

Десятичная приставка

Двоичная приставка

Название	Символ	Степень	Название	Символ		Степень
				МЭК	ГОСТ	
килобайт	kB	10 ³	кибибайт	KiB	Кбайт	2 ¹⁰
мегабайт	MB	10 ⁶	мебибайт	MiB	Мбайт	2 ²⁰
гигабайт	GB	10 ⁹	гибибайт	GiB	Гбайт	2 ³⁰
терабайт	TB	10 ¹²	тебибайт	TiB		2 ⁴⁰
петабайт	PB	10 ¹⁵	пебибайт	PiB		2 ⁵⁰

История развития информатики и вычислительной техники

Электронно-вычислительная машина (ЭВМ) или компьютер (англ. computer — «вычислитель»),— машина для проведения вычислений, а также приёма, переработки, хранения и выдачи информации по заранее определённом алгоритму (компьютерной программе). На заре эры компьютеров считалось, что основная функция компьютера — вычисление. Однако в настоящее время полагают, что основная их функция — управление.

Поколение первое.

Компьютеры на электронных лампах.

Компьютеры на основе электронных ламп появились в 40-х годах XX века. Первая электронная лампа - вакуумный диод - была построена Флемингом лишь в 1904 году, хотя эффект прохождения электрического тока через вакуум был открыт Эдисоном в 1883 году. Триггер, изобретенный М. А. Бонч-Бруевичем (1918) и - независимо - американцами У. Икклзом и Ф. Джорданом (1919), содержит 2 лампы и в каждый момент может находиться в одном из двух устойчивых состояний; он представляет собой электронное реле. Подобно электромеханическому, оно может быть использовано для хранения одной двоичной цифры. Использование электронной лампы в качестве основного элемента ЭВМ создавало множество проблем. Из-за того, что высота стеклянной лампы - 7см, машины были огромных размеров. Каждые 7-8 мин. одна из ламп выходила из строя, а так как в компьютере их было 15 - 20 тысяч, то для поиска и замены поврежденной лампы требовалось очень много времени. Кроме того, они выделяли огромное количество тепла, и для эксплуатации "современного" компьютера того времени требовались специальные системы охлаждения.



Чтобы разобраться в запутанных схемах огромного компьютера, нужны были целые бригады инженеров. Устройств ввода в этих компьютерах не было, поэтому данные заносились в память при помощи соединения нужного штеккера с нужным гнездом.

ЭНИАК (ENIAC, сокр. от англ. Electronic Number Integrator And Computer — Электронный числовой интегратор и вычислитель) — первый широкомасштабный электронный цифровой компьютер, который можно было перепрограммировать для решения полного диапазона задач (предыдущие компьютеры имели только часть из этих свойств). Построен в 1946 году по заказу Армии США в Лаборатории баллистических исследований для расчётов таблиц стрельбы. Запущен 14 февраля 1946 года.

Архитектуру компьютера разработали в 1943 году Джон Преспер Экерт и Джон Уильям Мокли, учёные из Университета Пенсильвании. В отличие от созданного в 1941 году немецким инженером Конрадом Цузе комплекса Z3, использовавшего механические реле, в ЭНИАКе в качестве основы компонентной базы применялись вакуумные лампы. Всего комплекс включал 17468 ламп, 7200 кремниевых диодов, 1500 реле, 70000 резисторов и 10000 конденсаторов. Потребляемая мощность — 150 кВт. Вычислительная мощность — 300 операций умножения или 5000 операций сложения в секунду. Вес - 27 тонн. Вычисления производились в десятичной системе.

На сегодняшний день производительность компьютеров около 38,4 миллиардов операций в секунду, производительность суперкомпьютеров при пиковой мощности в 95,04 триллионов операций в секунду.

До 1948 года для перепрограммирования ENIAC нужно было его перекоммутировать его заново.

Механические устройства для обработки числовой информации были изобретены в США и впервые использовались практически для обработки результатов переписи населения в 1890г. Изобретатель этих машин, Герман Холерит, в конце XIX века основал фирму, называемой IBM.

Современные вычислительные машины основаны на 2-х принципах:

- в памяти ЭВМ хранятся не только данные, но и сама программа;
- и то и другое хранится в виде многозначных двоичных чисел.

Первыми вычислительными машинами, базирующихся на этих принципах были: EDSAC (1949, Англия), EDVAC (1950, США), МЭСМ (1951, СССР). Появление ЭВМ произвело революцию в технологии процессов создания, накопления, передачи и обработки информации.

Поколение второе.

Транзисторные компьютеры.



1 июля 1948 года на одной из страниц "Нью-Йорк Таймс", посвященной радио и телевидению, было помещено скромное сообщение о том, что фирма "Белл телефон лабораториз" разработала электронный прибор, способный заменить электронную лампу. Физик-теоретик Джон Бардин и ведущий экспериментатор фирмы Уолтер Брайттен создали первый действующий транзистор. Это был точечно-контактный прибор, в котором три металлических "усика" контактировали с брусом из поликристаллического германия.

Первые компьютеры на основе транзисторов появились в конце 50-х годов, а к середине 60-х годов были созданы более компактные внешние устройства, что позволило фирме Digital Equipment выпустить в 1965 г. первый мини-компьютер PDP-8 размером с холодильник (!!) и стоимостью всего 20 тыс. долларов (!!).

Созданию транзистора предшествовала упорная, почти 10-летняя работа, которую еще в 1938 году начал физик теоретик Уильям Шокли. Применение транзисторов в качестве основного элемента в ЭВМ привело к уменьшению размеров компьютеров в сотни раз и к повышению их надежности.

И все-таки самой удивительной способностью транзистора является то, что он один способен трудиться за 40 электронных ламп и при этом работать с большей скоростью, выделять очень мало тепла и почти не потреблять электроэнергию. Одновременно с процессом замены электронных ламп транзисторами совершенствовались методы хранения информации. Увеличился объем памяти, а магнитную ленту, впервые примененную в ЭВМ Юнивак, начали использовать как для ввода, так и для вывода информации. А в середине 60-х годов получило распространение хранение информации на дисках. Большие достижения в архитектуре компьютеров позволило достичь быстродействия в миллион операций в секунду!

Поколение третье.

Интегральные схемы.



Подобно тому, как появление транзисторов привело к созданию второго поколения компьютеров, появление интегральных схем ознаменовало собой новый этап в развитии вычислительной техники - рождение машин третьего поколения. Интегральная схема, которую также называют кристаллом, представляет собой миниатюрную электронную схему, вытравленную на поверхности кремниевого кристалла площадью около 10 мм. Это специально выращенный полупроводниковый кристалл, на котором располагаются транзисторы, соединенные напыленными алюминиевыми проводниками. Кристалл помещается в керамический корпус

с контактами.

Интегральная (микро)схема (ИС, ИМС, м/сх, англ. integrated circuit, IC, microcircuit), чип, микрочип (англ. microchip, silicon chip, chip — тонкая пластинка — первоначально термин относился к пластинке кристалла микросхемы) — микроэлектронное устройство — электронная схема произвольной сложности, изготовленная на полупроводниковом кристалле (или плёнке) и помещённая в неразборный корпус.

Первые интегральные схемы (ИС) появились в 1964 году. Сначала они использовались только в космической и военной технике. Сейчас же их можно обнаружить где угодно, включая автомобили и бытовые приборы. Что же касается компьютеров, то без интегральных схем они просто немислимы!

Появление ИС означало подлинную революцию в вычислительной технике. Ведь она одна способна заменить тысячи транзисторов, каждый из которых в свою очередь уже заменил 40 электронных ламп. Другими словами, один крошечный кристалл обладает такими же вычислительными возможностями, как и 30-тонный Эниак! Быстродействие ЭВМ третьего поколения возросло в 100 раз, а габариты значительно уменьшились.

Ко всем достоинствам ЭВМ третьего поколения добавилось еще и то, что их производство оказалось дешевле, чем производство машин второго поколения. Благодаря этому, многие организации смогли приобрести и освоить такие машины. А это, в свою очередь, привело к росту спроса на универсальные ЭВМ, предназначенные для решения самых различных задач. Большинство созданных до этого ЭВМ являлись специализированными машинами, на которых можно было решать задачи какого-то одного типа.

Поколение четвертое.

Большие интегральные схемы.

Вы уже знаете, что электромеханические детали счетных машин уступили место электронным лампам, которые в свою очередь уступили место транзисторам, а последние - интегральным схемам. Могло создаться впечатление, что технические возможности ЭВМ исчерпаны. В самом деле, что же можно еще придумать?

Чтобы получить ответ на этот вопрос, давайте вернемся к началу 70-х годов. Именно в это время была предпринята попытка выяснить, можно ли на одном кристалле разместить больше одной интегральной схемы. Оказалось, можно! *Развитие микроэлектроники привело к созданию возможности размещать на одном-единственном кристалле тысячи интегральных схем.* Так, уже в 1980 году, центральный процессор небольшого компьютера оказался возможным разместить на кристалле, площадью всего в четверть квадратного дюйма (1,61 см²). Началась эпоха микрокомпьютеров.

Современные процессоры изготавливаются по 0,13-микронной технологии, т.е. толщина кристалла процессора составляет 0,13 микрон. Для сравнения - толщина кристалла первого процессора Intel была 10 микрон.

В первом процессоре компании Intel - i4004, выпущенном в 1971 году, на одном кристалле было 2300 транзисторов, а в процессоре Intel Pentium 4, выпущенном 14 апреля 2003 года, их уже 55 миллионов.

Каково же быстродействие современной микроЭВМ? Оно в 10 раз превышает быстродействие ЭВМ третьего поколения на интегральных схемах, в 1000 раз - быстродействие ЭВМ второго поколения на транзисторах и в 100000 раз - быстродействие ЭВМ первого поколения на электронных лампах.

Компьютеры пятого поколения

Компьютеры пятого поколения (яп. 第五世代コンピュータ) — широкомасштабная правительственная программа в Японии по развитию компьютерной индустрии и искусственного интеллекта, предпринятая в 1980-е годы. Целью программы было создание «эпохального компьютера» с производительностью суперкомпьютера и мощными функциями искусственного интеллекта. Начало разработок — 1982, конец разработок — 1992, стоимость разработок — 57 млрд ¥ (порядка 500 млн \$).

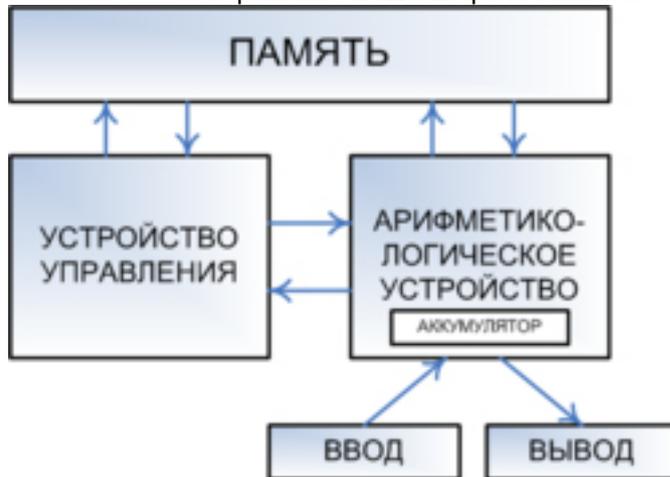
С любых точек зрения проект можно считать абсолютным провалом. За десять лет на разработки было истрочено более 50 млрд ¥, и программа завершилась, не достигнув цели.

Рабочие станции так и не вышли на рынок, потому что однопроцессорные системы других фирм превосходили их по параметрам, программные системы так и не заработали, появление Интернета сделало все идеи проекта безнадежно устаревшими.

Машина фон Неймана.

Основная компоновка частей компьютера и связь между ними называется архитектурой. При описании архитектуры компьютера определяется состав входящих в него компонент, принципы их взаимодействия, а также их функции и характеристики.

Схематичное изображение машины фон Неймана.



Принципы фон Неймана

1. Принцип использования двоичной системы счисления для представления данных и команд.
2. Принцип программного управления.

Программа состоит из набора команд, которые выполняются процессором друг за другом в определенной последовательности.

3. Принцип однородности памяти.

Как программы (команды), так и данные хранятся в одной и той же памяти (и кодируются в одной и той же системе счисления — чаще всего двоичной). Над командами можно выполнять такие же действия, как и над данными.

4. Принцип адресуемости памяти.

Структурно основная память состоит из пронумерованных ячеек; процессору в произвольный момент времени доступна любая ячейка.

5. Принцип последовательного программного управления

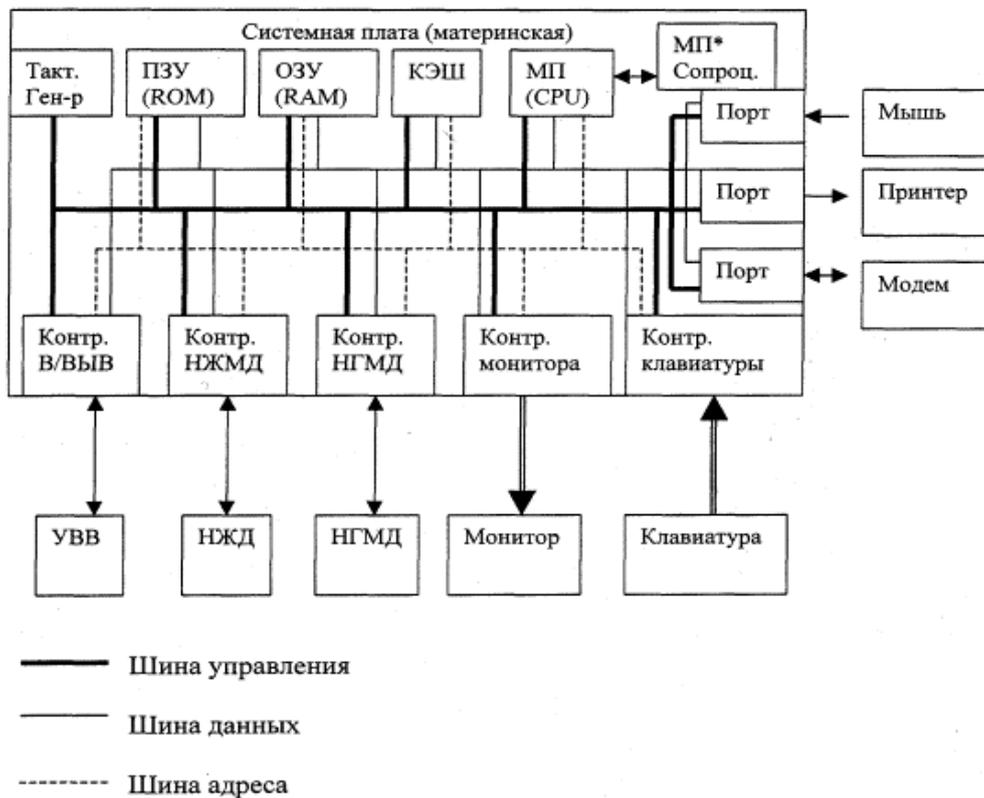
Все команды располагаются в памяти и выполняются последовательно, одна после завершения другой.

6. Принцип условного перехода.

Конкретный набор компонент, входящих в данный компьютер, называется его

конфигурацией. Минимальная конфигурация ПК необходимая для его работы включает в себя системный блок (там находятся МП, ОП, ПЗУ, НЖМД), клавиатуру (как устройство ввода информации) и монитор (как устройство вывода информации).

Рассмотрим устройства подробнее.

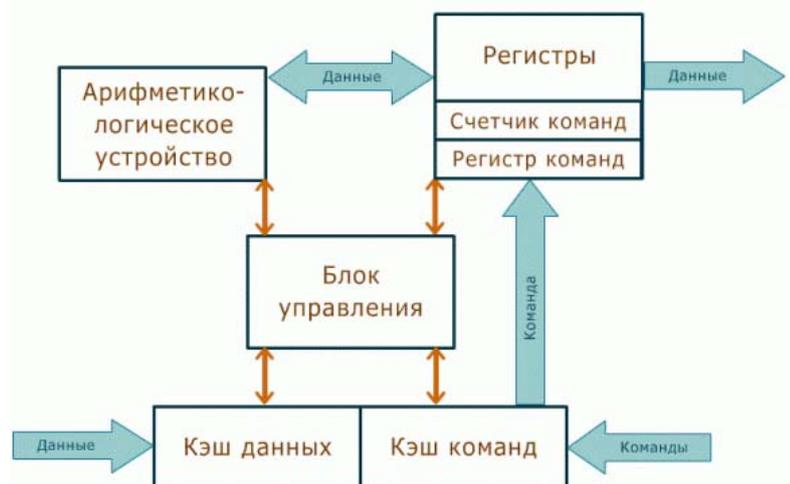


Микропроцессор (МП) или CPU (Central Processing Unit).

Центральный процессор (ЦП, или центральное процессорное устройство — ЦПУ; англ. central processing unit, CPU, дословно — центральное обрабатывающее устройство) — исполнитель машинных инструкций, часть аппаратного обеспечения компьютера или программируемого логического контроллера; отвечает за выполнение операций, заданных программами. МП имеет сложную структуру в виде электронных логических схем. В качестве его компонент можно выделить:

- 1) АЛУ - арифметико-логическое устройство, предназначенное для выполнения арифметических и логических операций над данными и адресами памяти;
- 2) Регистры или микропроцессорная память — сверхоперативная память, работающая со скоростью процессора, АЛУ работает именно с ними;
- 3) БУ - блок управления - управление работой всех узлов МП посредством выработки и передачи другим его компонентам управляющих импульсов, поступающих от кварцевого тактового генератора, который при включении ПК начинает вибрировать с постоянной частотой. Эти колебания и задают темп работы всей системной платы;

Схема процессора (упрощенная)



Процессор «общается» с другими устройствами (оперативной памятью) с помощью шин данных, адреса и управления. Разрядность шин всегда кратна 8 (понятно почему, если мы имеем дело с байтами), изменчива в ходе исторического развития компьютерной техники и различна для разных моделей, а также не одинакова для шины данных и адресной шины.

Разрядность шины данных говорит о том, какое количество информации (сколько байт) можно передать за раз (за такт). От разрядности шины адреса зависит максимальный объем оперативной памяти, с которым процессор может работать вообще.

На мощность (производительность) процессора влияют не только его тактовая частота и разрядность шины данных, также важное значение имеет объем кэш-памяти.

Характеристики процессора:

1. Тактовая частота — это количество операций, которое процессор может выполнить в секунду. Единица измерения МГц и ГГц (мегагерц и гигагерц). 1 МГц — значит, что процессор может выполнить 1 миллиард операций в секунду, если процессор 3,16 ГГц — следовательно он может выполнить 3 Миллиарда 166 миллионов операций за 1 секунду.

Существует два типа тактовой частоты — внутренняя и внешняя.

Внутренняя тактовая частота — это тактовая частота, с которой происходит работа внутри процессора.

Внешняя тактовая частота или частота системной шины — это тактовая частота, с которой происходит обмен данными между процессором и оперативной памятью компьютера.

До 1992 года в процессорах внутренняя и внешняя частоты совпадали, а в 1992 году компания Intel представила процессор 80486DX2, в котором внутренняя и внешняя частоты были различны — внутренняя частота была в 2 раза больше внешней. Было выпущено два типа таких процессоров с частотами 25/50 МГц и 33/66 МГц, затем Intel выпустила процессор 80486DX4 с утроенной внутренней частотой (33/100 МГц).

В современных процессорах, например, при тактовой частоте процессора 3 ГГц, частота системной шины 800 МГц.

2. Другой основной характеристикой процессора является его разрядность.

Разрядность процессора определяется разрядностью его регистров.

Компьютер может оперировать одновременно ограниченным набором единиц информации. Этот набор зависит от разрядности внутренних регистров. Разряд — это хранилище единицы информации. За один рабочий такт компьютер может обработать количество информации, которое может поместиться в регистрах. Если регистры могут хранить 8 единиц информации, то они 8-разрядные, и процессор 8-разрядный, если регистры 16-разрядные, то и процессор 16-разрядный и т.д. Чем большая разрядность процессора, тем большее количество информации он может обработать за один такт, а значит, тем быстрее работает процессор.

Процессор Pentium 4 является 32-разрядным. Сейчас всё больше процессоров 64 разрядные.

3. Кэш процессора — довольно важный параметр. Чем он больше, тем больше данных хранится в особой памяти, которая ускоряет работу процессора. В кэше процессора находятся данные, которые могут понадобиться в работе в самое ближайшее время. Чтобы вы не путались в уровнях кэша — запомните одно свойство: кэш первого уровня самый быстрый, но самый маленький, второго — помедленней, но побольше и кэш третьего уровня самый медленный и самый большой (если он есть)

4. Технический процесс (иногда пишут технология) — не основная характеристика процессора для обычного обывателя, но знать о нем надо, чтобы понимать заумные статьи на компьютерных сайтах. Чем меньше тех процесс, тем как говорится, лучше. По факту — это площадь кристалла на процессоре. Чем кристаллы меньше, тем их больше можно уместить, следовательно увеличить тактовую частоту. Да и на меньший кристалл нужно меньше подавать напряжения, поэтому и тепловыделение уменьшается, поэтому опять же можно увеличить тактовую частоту. Эта цепочка приведена в пример, что бы вы поняли как всё взаимосвязано. Тех процесс в прайсах могут и не написать, но в обзорах его упоминают почти всегда.

5. Socket — этот параметр нужен для стандартизации всех процессоров по разъемам подключения к материнской плате. Например, Socket LGA775 — если вы такую характеристику встретите на материнской плате, то к ней подойдут только процессоры с маркировкой Socket LGA775 и никакие другие. Обратное правило тоже действует.

Интерфейсная система - это:

- шина управления (ШУ) - предназначена для передачи управляющий импульсов и синхронизации сигналов ко всем устройствам ПК;
- шина адреса (ША) - предназначена для передачи кода адреса ячейки памяти или порта ввода/вывода внешнего устройства;
- шина данных (ШД) - предназначена для параллельной передачи всех разрядов числового кода;
- шина питания - для подключения всех блоков ПК к системе электропитания.

Интерфейсная система обеспечивает три направления передачи информации:

- между МП и оперативной памятью;
 - между МП и портами ввода/вывода внешних устройств;
 - между оперативной памятью и портами ввода/вывода внешних устройств.
- Обмен информацией между устройствами и системной шиной происходит с помощью кодов ASCII.

Память

Память - устройство для хранения информации в виде данных и программ. Память делится прежде всего на внутреннюю (расположенную на системной плате) и внешнюю (размещенную на разнообразных внешних носителях информации).

Внутренняя память в свою очередь подразделяется на:

- **- ПЗУ** (постоянное запоминающее устройство) или ROM (read only memory), которое содержит - постоянную информацию, сохраняемую даже при отключенном питании, которая служит для тестирования памяти и оборудования компьютера, начальной загрузки ПК при включении. Запись на специальную кассету ПЗУ происходит на заводе фирмы-изготовителя ПК и несет черты его индивидуальности. Объем ПЗУ относительно невелик - от 64 до 256 Кб.
- **- ОЗУ** (оперативное запоминающее устройство, ОП — оперативная память) или RAM (random access memory), служит для оперативного хранения программ и данных, сохраняемых только на период работы ПК. Она энергозависима, при отключении питания информация теряется. ОП выделяется особыми функциями и спецификой доступа:
 - ОП хранит не только данные, но и выполняемую программу;
 - МП имеет возможность прямого доступа в ОП, минуя систему ввода/вывода.
- **Кэш-память** - имеет малое время доступа, служит для временного хранения промежуточных результатов и содержимого наиболее часто используемых ячеек ОП и регистров

Логическая организация памяти — адресация, размещение данных определяется ПО, установленным на ПК, а именно ОС.

Внешняя память. Устройства внешней памяти весьма разнообразны. Предлагаемая классификация учитывает тип носителя, т.е. материального объекта, способного хранить информацию.

- **Накопители на магнитной ленте** исторически появились раньше, чем накопители на магнитном диске. Бобинные накопители используются в суперЭВМ и mainframe.
- **Диски** относятся к носителям информации с прямым доступом, т.е. ПК может обратиться к дорожке, на которой начинается участок с искомой информацией или куда нужно записать новую информацию, непосредственно.

Магнитные диски (МД)— в качестве запоминающей среды используются магнитные материалы со специальными свойствами, позволяющими фиксировать два направления намагниченности. На сегодняшний день редко используемые.

НЖМД или «винчестеры» изготовлены из сплавов алюминия или из керамики и покрыты ферролаком, вместе с блоком магнитных головок помещены в герметически закрытый корпус. Емкость накопителей за счет чрезвычайно плотной записи достигает нескольких гигабайт, быстродействие также выше, чем у съемных дисков (за счет увеличения скорости вращения, т.к. диск жестко закреплен на оси вращения). Первая модель появилась на фирме IBM в 1973 г. Она имела емкость 16 Кб и 30 дорожек/30 секторов, что случайно совпало с калибром популярного ружья 30'730" «винчестер».

Каждый ЖМД проходит процедуру низкоуровневого форматирования — на носитель записывается служебная информация, которая определяет разметку цилиндров диска на сектора и нумерует их, маркируются дефектные сектора для исключения их из процесса эксплуатации диска. В ПК имеется один или два накопителя. Один ЖД можно разбить при помощи специальной программы на несколько логических дисков и работать с ними как с разными ЖД.

НОД (накопители на оптических дисках) лазерно-оптические диски или компакт-диски (CD, DVD). В оптическом дисководе ПК эта дорожка читается лазерным лучом. Ввиду чрезвычайно плотной записи имеют емкость до 8 Гб.

Флеш-память (англ. flash memory) — разновидность твердотельной полупроводниковой энергонезависимой перезаписываемой памяти (ПППЗУ).

Она может быть прочитана сколько угодно раз (в пределах срока хранения данных, типично — 10-100 лет), но писать в такую память можно лишь ограниченное число раз (максимально — около миллиона циклов). Распространена флеш-память, выдерживающая около 100 тысяч циклов перезаписи — намного больше, чем способна выдержать дискета или CD-RW.

Контроллеры

Служат для обеспечения прямой связи с ОП, минуя МП, они используются для устройств быстрого обмена данными с ОП - НГМД, НЖД, дисплей и др., обеспечения работы в групповом или сетевом режиме. *Клавиатура, дисплей, мышь являются медленными устройствами*, поэтому они связаны с системной платой контроллерами и имеют в ОП свои отведенные участки памяти.

Порты бывают входными и выходными, универсальными (ввод - вывод), они служат для обеспечения обмена информацией ПК с внешними, не очень быстрыми устройствами. Информация, поступающая через порт, направляется в МП, а потом в ОП.

Устройства вывода.

Видеомониторы — устройства, предназначенные для вывода информации от ПК пользователю. Самые качественные RGB-мониторы, обладают высокой разрешающей способностью для графики и цвета. Используется тот же принцип электронной лучевой трубки как у телевизора. Сейчас часто используют электролюминесцентные или жидкокристаллические панели. Мониторы могут работать в текстовом и графическом режимах. В текстовом режиме изображение состоит из знакомест — специальных знаков, хранимых в видеопамяти дисплея, а в графическом изображение состоит из точек определенной яркости и цвета. Основные характеристики видеомониторов - разрешающая способность (как правило 1024x768 точек), число цветов (для цветных) -от 16 до 16 миллионов.

Принтеры — это устройства вывода данных из ЭВМ, преобразовывающие информационные ASCII-коды в соответствующие им графические символы и фиксирующие эти символы на бумаге. Принтеры - наиболее развитая группа внешних устройств, насчитывается более 1000 модификаций.

Принтеры бывают черно-белые или цветные по способу печати они делятся на:

матричные — в этих принтерах изображение формируется из точек ударным способом, игольчатая печатающая головка перемещается в горизонтальном направлении, каждая иголочка управляется электромагнитом и ударяет бумагу через красящую ленту.

струйные — в печатающей головке имеются вместо иголок тонкие трубочки - сопла, через которые на бумагу выбрасываются мельчайшие капельки чернил.

термографические — матричные принтеры, оснащенные вместо игольчатой печатающей головки головкой с термоматрицей, при печати используется специальная термобумага;

лазерные — используется электрографический способ формирования изображений, лазер служит для создания сверхтонкого светового луча, вычерчивающего на поверхности светочувствительного барабана контуры невидимого точечного электронного изображения. После проявления изображения порошком красителя (тонера), налипающего на разряженные участки, выполняется печать - перенос тонера на бумагу и закрепление изображения на бумаге при помощи высокой температуры.

3D-принтеры — устройство, использующее метод создания физического объекта на основе виртуальной 3D-модели. 3D-печать может осуществляться разными способами и с использованием различных материалов, но в основе любого из них лежит принцип послойного создания (выращивания) твердого объекта.

Устройства ввода

Манипуляторы - компьютерные устройства, управляемые руками оператора:

мышь — устройство для определения относительных координат (смещения относительно предыдущего положения или направления) движения руки оператора. Относительные координаты передаются в компьютер и при помощи специальной программы могут вызывать перемещения курсора на экране. Для отслеживания перемещения мыши используются различные виды датчиков.

джойстик — рычажный указатель - устройство для ввода направления движения руки оператора, их чаще используют для игр на компьютере;

дигитайзер или оцифровывающий планшет — устройство для точного ввода графической информации (чертежей, графиков, карт) в компьютер. Он состоит из плоской панели (планшета) и связанной с ней ручной устройства - пера. Оператор ведет вдоль графика перо, при этом абсолютные координаты поступают в компьютер.

клавиатура — устройство для ввода информации в память компьютера. Внутри расположена микросхема, клавиатура связана с системной платой, нажатие любой клавиши продуцирует сигнал (код символа в системе ASCII -16-ричный порядковый номер символа в таблице), в памяти ЭВМ специальная программа по коду восстанавливает внешний вид нажатого символа и передает его изображение на монитор.

Микросхема ПЗУ и система BIOS

В момент включения компьютера в его оперативной памяти нет ничего — ни данных, ни программ, поскольку оперативная память не может ничего хранить без подзарядки ячеек более сотых долей секунды. Но процессору нужны команды, в том числе и в первый момент после включения. Поэтому сразу после включения на адресной шине процессора выставляется стартовый адрес. Это происходит аппаратно, без участия программ (всегда одинаково). Процессор обращается по выставленному адресу за своей первой командой и далее начинает работать по программам. Этот исходный адрес не может указывать на оперативную память, в которой пока ничего нет. Он указывает на другой тип памяти — постоянное запоминающее устройство (ПЗУ). Микросхема ПЗУ способна длительное время хранить информацию, даже когда компьютер выключен. Программы, находящиеся в ПЗУ, называют «защитами» — их записывают туда на этапе изготовления микросхемы.

Комплект программ, находящихся в ПЗУ, образует базовую систему ввода-вывода (BIOS — Basic Input Output System). Основное назначение программ этого пакета состоит в том, чтобы проверить состав и работоспособность компьютерной системы и обеспечить взаимодействие с клавиатурой, монитором, жестким диском и дисководом гибких дисков. Программы, входящие в BIOS, позволяют нам наблюдать на экране диагностические сообщения, сопровождающие запуск компьютера, а также вмешиваться в ход запуска с помощью клавиатуры.

BIOS выполняет самотестирование устройств (англ. POST – Power-On Self Test), а затем ищет загрузчик операционной системы (англ. Boot Loader) на доступных носителях информации. Если загрузчик не найден, BIOS выдаёт сообщение об ошибке.

Также BIOS содержит минимальный набор сервисных функций (например, для вывода сообщений на экран или приёма символов с клавиатуры), что и обуславливает расшифровку её названия: Basic Input-Output System — Базовая система ввода-вывода.

Устройства, инициализируемые BIOS

Загрузочное устройство — устройство, которое должно быть проинициализировано до загрузки операционной системы. К ним относятся устройства ввода (клавиатура, мышь), базовое устройство вывода (дисплей), и устройство, с которого будет произведена загрузка ОС — дисковод, жесткий диск, CD-ROM, флэш-диск, SCSI-устройство, сетевая карта (при загрузке по сети; например, при помощи PXE).

Загрузочная последовательность стандартного IBM-совместимого персонального компьютера

После включения персонального компьютера его процессор начинает работу. Первая выполняемая команда расположена по адресу FFFF0h и принадлежит пространству адресов BIOS. Как правило, данная команда просто передает управление программе инициализации BIOS. Программа инициализации BIOS с помощью программы POST проверяет, что устройства компьютера работают корректно и инициализирует их. Затем BIOS опрашивает устройства перечисляемые в заранее созданном списке пока не найдет загрузочное устройство. Если такое устройство найдено не будет, будет выведено сообщение об ошибке а процесс загрузки будет остановлен. Если BIOS обнаружит загрузочное устройство, он считывает с него начальный загрузчик и передает ему управление. В случае жесткого диска, начальный загрузчик называется главной загрузочной записью (MBR) и часто не зависит от операционной системы. Обычно он ищет активный разделы жесткого диска загружает загрузочный сектор данного раздела и передает ему управление. Этот загрузочный сектор, как правило, зависит от операционной системы. Он должен загрузить в память ядро операционной системы и передать ему управление. Если активного раздела не существует, или загрузочный сектор активного раздела некорректен, MBR может загрузить резервный начальный загрузчик и передать управление ему. Резервный начальный загрузчик должен выбрать раздел (зачастую с помощью пользователя), загрузить его загрузочный сектор и передать ему управление.

Лекция 2

Операционная система.

Операционная система — базовый комплекс компьютерных программ, обеспечивающий управление аппаратными средствами компьютера, работу с файлами, ввод и вывод данных, а также выполнение прикладных программ и утилит.

Операционная система сокр. ОС (англ. operating system) — комплекс управляющих и обрабатывающих программ, которые, с одной стороны, выступают как интерфейс между устройствами вычислительной системы и прикладными программами, а с другой — предназначены для управления устройствами, управления вычислительными процессами,

эффективного распределения вычислительных ресурсов между вычислительными процессами и организации надёжных вычислений.

При включении компьютера операционная система загружается в память раньше остальных программ и затем служит платформой и средой для их работы. Помимо вышеуказанных функций ОС может осуществлять и другие, например, предоставление пользовательского интерфейса, сетевое взаимодействие и т. п. 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.

Основные функции (простейшие ОС):

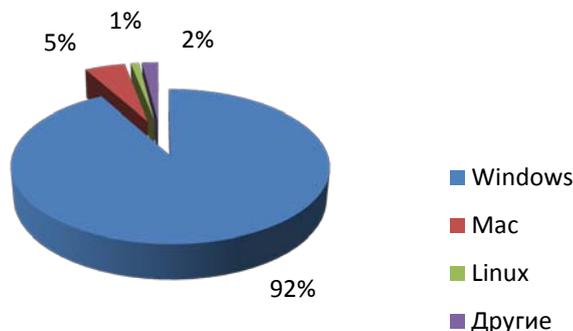
- Загрузка приложений в оперативную память и их выполнение;
- Стандартизированный доступ к периферийным устройствам (устройства ввода-вывода);
- Управление оперативной памятью (распределение между процессами, виртуальная память);
- Управление энергонезависимой памятью (Жёсткий диск, Компакт-диск и т.д.), как правило с помощью файловой системы;
- Пользовательский интерфейс;

Дополнительные функции (развитые современные ОС):

- Параллельное или псевдопараллельное выполнение задач (многозадачность);
- Взаимодействие между процессами;
- Межмашинное взаимодействие (компьютерная сеть);
- Защита самой системы, а также пользовательских данных и программ от злонамеренных действий пользователей или приложений;
- Разграничение прав доступа и многопользовательский режим работы (аутентификация, авторизация).

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. С 1990-х наиболее распространёнными операционными системами являются ОС семейства Microsoft Windows и системы класса UNIX (особенно GNU/Linux).

В настоящее время Microsoft Windows установлена примерно на 92 % персональных компьютеров и рабочих станций. По данным компании Net Applications, в марте 2010 года рыночная доля Windows составляла ▼91,63 %, Mac — ▲5,33 %, Linux — ▲1,03 % прочие — менее 1 %.



Из истории Windows

Уильям Генри Гейтс III вместе с одним из школьных друзей – Полом Алленом – в 1975 году создал фирму Micro-Soft, впоследствии переименованную в Microsoft. Ради работы в компании в 1975 году ушел с третьего курса Гарвардского университета. Первой масштабной сделкой Microsoft стала продажа компании IBM операционной системы для персонального компьютера – MS-DOS. Дальнейшая продажа этой системы другим производителям компьютеров породила индустрию IBM-совместимых ПК, зависимых от программной продукции Microsoft.

MS-DOS

MS-DOS была выпущена компанией Microsoft в 1981 году. DOS (англ. Disk Operating System — дисковая операционная система, ДОС) — семейство операционных систем для персональных компьютеров. Ориентировано на использование дисковых накопителей, таких как жёсткий диск и дискета.

Графические интерфейсы и расширения для DOS

В 1985 году была выпущена первая версия графической оболочки Windows, представлявшей собой дополнение к MS-DOS, но популярность она завоевала далеко не сразу — а только в 1990 году, когда вышла версия **Windows 3.0**.и выпущенные затем **Windows 3.1** и **Windows for Workgroups 3.11**.

Эти версии Windows не были полноценными операционными системами, а являлись надстройками к операционной системе MS-DOS и были по сути многофункциональным расширением, добавляя поддержку новых режимов работы процессора, поддержку многозадачности, обеспечивая стандартизацию интерфейсов аппаратного обеспечения и единообразие для пользовательских интерфейсов программ. Предоставляли встроенные средства (GDI) для создания графического интерфейса пользователя.

Графический интерфейс позволяет работать с объектами вашего компьютера не с помощью команд, а с помощью наглядных и понятных действий над значками, обозначающими эти объекты. Возможность одновременной работы с несколькими программами значительно повысила удобство и эффективность работы. Кроме того, удобство и легкость написания программ для Windows привели к появлению все больше разнообразных программ, работающих под управлением Windows. Наконец, лучше была организована работа с разнообразным компьютерным оборудованием, что также определило популярность системы. Последующие версии Windows были направлены на повышение надежности, а также поддержку средств мультимедиа (версия 3.1) и работу в компьютерных сетях (версия 3.11).

ОС Windows

Процесс развития операционных систем не стоит на месте, и в 1995 появилась **система Windows 95**, ставшая новым этапом в истории Windows: значительно изменился интерфейс, выросла скорость работы программ, в состав системы был включен браузер Internet Explorer.

Продолжением развития Windows 95 стала операционная система, появившаяся в 1998 году (**Windows 98**). При сохранившемся интерфейсе внутренняя структура была значительно переработана. Много внимания было уделено работе с Интернетом, а также поддержке современных протоколов передачи информации — стандартов, обеспечивающих обмен информацией между различными устройствами.

Следующим этапом в развитии Windows стало появление **Windows 2000** и **Windows Me** (Millennium Edition — редакция тысячелетия). Система Windows 2000 разработана на основе Windows NT и унаследовала от нее высокую надежность и защищенность информации от постороннего вмешательства. Операционная система Windows Me стала наследницей Windows 98, но приобрела многие новые возможности. Прежде всего, это улучшенная работа со средствами мультимедиа, возможность записывать не только аудио, но и видеoinформацию, мощные средства восстановления информации после сбоев и многое другое.

Windows XP (кодовое название при разработке — Whistler; внутренняя версия — Windows NT 5.1) — операционная система семейства Windows NT от компании Microsoft. Она была выпущена 25 октября 2001 года и является развитием Windows 2000 Professional. Название XP происходит от англ. experience (опыт, впечатление).

Windows Vista как и Windows XP, исключительно клиентская система. Серверный вариант выйдет предположительно в октябре-ноябре 2007 года и будет называться **Windows Server 2008**.

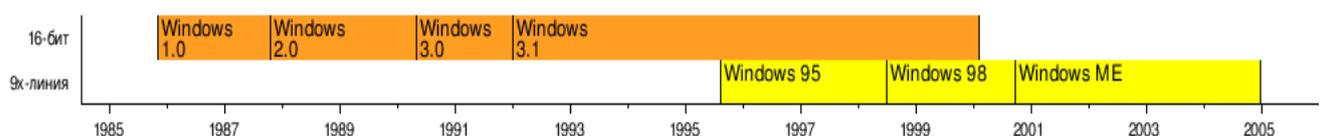
Windows 7 — операционная система следующая за Windows Vista. Серверной версией является Windows Server 2008 R2, версией для интегрированных систем — Windows Embedded Standard 2011 (Quebec),[1] мобильной — Windows Embedded Compact 2011 (Chelan, Windows CE 7.0). Операционная система поступила в продажу 22 октября 2009 года меньше чем через три года после выпуска предыдущей операционной системы, Windows Vista.

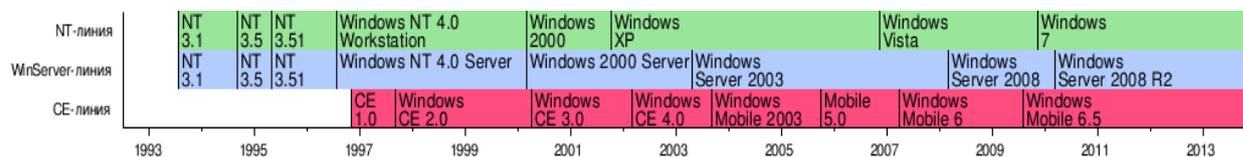
В состав Windows 7 вошли как некоторые разработки, исключённые из Windows Vista, так и новшества в интерфейсе и встроенных программах. Из состава Windows 7 были исключены игры Inkball, Ultimate Extras; приложения, имеющие аналоги в Windows Live (Почта Windows, Календарь Windows и пр.), технология Microsoft Agent, Windows Meeting Space; из меню «Пуск» исчезла возможность вернуться к классическому меню и автоматическая пристыковка браузера и клиента электронной почты.

Семейство ОС Windows Mobile для карманных компьютеров

Это семейство операционных систем реального времени было специально разработано для встраиваемых систем. Поддерживаются процессоры ARM, MIPS, SuperH и x86. В отличие от остальных операционных систем Windows, операционные системы этого семейства продаются только в составе готовых устройств, таких как смартфоны, карманные компьютеры, GPS навигаторы, MP3 проигрыватели, и другие.

В настоящее время под термином «Windows CE» понимают только ядро операционной системы. Например Windows Mobile 5.0 включает в себя ядро Windows CE 5.0, хотя в некоторых устройствах ядро Windows CE используется и без Windows Mobile.





ОС Linux.

Linux - это современная POSIX-совместимая и Unix-подобная операционная система для персональных компьютеров и рабочих станций.

Это многопользовательская сетевая операционная система с сетевой оконной графической системой X Window System. ОС Linux поддерживает стандарты открытых систем и протоколы сети Internet и совместима с системами Unix, DOS, MS Windows. Все компоненты системы, включая исходные тексты, распространяются с лицензией на свободное копирование и установку для неограниченного числа пользователей.

Разработка ОС Linux выполнена Линусом Торвалдсом из университета Хельсинки и не поддающейся подсчету обширной командой из тысяч пользователей сети Internet, сотрудников исследовательских центров, фондов, университетов и т.д

Debian — это свободная операционная система (ОС) и набор прикладных программ для вашего компьютера. В Debian используется ядро Linux, но большинство утилит ОС разработано в рамках проекта GNU; поэтому полное название проекта — Debian GNU/Linux.

Как уже было отмечено выше, Debian GNU/Linux — это не только операционная система. В его состав входит более 25000 пакетов заранее скомпилированного программного обеспечения, которые легко могут быть установлены.

Языки программирования

Машинный код процессора

Процессор компьютера— это большая интегральная микросхема. Все команды и данные он получает в виде электрических сигналов. Их можно представить как совокупности нулей и единиц, то есть числами. Разным командам соответствуют разные числа. Поэтому реально программа, с которой работает процессор, представляет собой последовательность чисел, называемую *машинным кодом*.

Алгоритм и программа

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 вписывается 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.

Что такое язык программирования

Самому написать программу в машинном коде весьма сложно, причем эта сложность резко возрастает с увеличением размера программы и трудоемкости решения нужной задачи. Условно можно считать, что машинный код приемлем если размер программы не превышает нескольких десятков байтов, и нет потребности в операциях ручного ввода/вывода данных.

Поэтому 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

Язык программирования — формальная знаковая система, предназначенная для описания алгоритмов в форме, которая удобна для исполнителя (например, компьютера). Язык программирования определяет набор лексических, синтаксических и семантических правил, используемых при составлении компьютерной программы. Он позволяет программисту точно определить то, на какие события будет реагировать компьютер, как будут храниться и передаваться данные, а также какие именно действия следует выполнять над этими данными при различных обстоятельствах.

Со времени создания первых программируемых машин *человечество придумало уже более двух с половиной тысяч языков программирования* (См. Список языков программирования(англ.)). Каждый год их число пополняется новыми. Некоторыми языками умеет пользоваться только небольшое число их собственных разработчиков, другие становятся известны миллионам людей. Профессиональные программисты иногда применяют в своей работе более десятка разнообразных языков программирования.

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 выполнения). Процесс поиска ошибок в программе называется *тестированием*, процесс устранения ошибок — *отладкой*.

Трансляторы

Компиляторы и интерпретаторы

Транслятор — программа, которая принимает на вход программу на одном языке (он в этом случае называется исходный язык, а программа — исходный код), и преобразует её в программу, написанную на другом языке (соответственно, целевой язык и объектный код).

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).

Интерпретатор берет очередной оператор языка из текста программы, анализирует его структуру и затем сразу исполняет (обычно после анализа оператор транслируется в некоторое промежуточное представление или даже машинный код для более эффективного дальнейшего исполнения). Только после того как текущий оператор успешно выполнен, интерпретатор перейдет к следующему. При этом, если один и тот же оператор должен выполняться в программе многократно, интерпретатор всякий раз будет выполнять его так, как будто встретил впервые. Вследствие этого, программы, в которых требуется осуществить большой объем повторяющихся вычислений, могут работать медленно. Кроме того, для выполнения такой программы на другом компьютере там также должен быть установлен интерпретатор — ведь без него текст программы является просто набором символов.

Интерпретатор - разновидность транслятора. Переводит и выполняет программу с языка высокого уровня в машинный код строка за строкой.

Компиляторы — выдает результат в виде исполняемого файла (в данном случае считаем, что компоновка входит в компиляцию). Этот файл:

- транслируется один раз — может быть запущен самостоятельно
- не требует для работы наличия на машине создавшего его транслятора

Компиляторы полностью обрабатывают весь текст программы (он иногда называется *исходный код*). Они просматривают его в поисках синтаксических ошибок (иногда несколько раз), выполняют определенный смысловой анализ и затем автоматически переводят (*транслируют*) на машинный язык — генерируют машинный код.

Уровни языков программирования

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**.

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 переносятся 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.

Популярным на сегодня является язык программирования: **Pascal (Паскаль)**. Язык Паскаль, создан в конце 70-х годов основоположником множества идей современного

программирования Никлаусом Виртом и имеет возможности, позволяющие успешно применять его при создании крупных проектов.

Basic (Бейсик), Для этого языка имеются и компиляторы, и интерпретаторы, а по популярности он занимает первое место в мире. Он создавался в 60-х годах в качестве учебного языка и очень прост в изучении. Его современная модификация Visual Basic совместима с Microsoft office, имеет возможности, аналогичные Паскалю, и позволяет, например, расширять возможности пакетов Excel и Access.

С (Си), Данный язык был создан в лаборатории Bell и первоначально не рассматривался как массовый. Он планировался для замены ассемблера, чтобы иметь возможность создавать столь же эффективные и компактные программы, и в то же время не зависеть от конкретного типа процессора. Си во многом похож на Паскаль и имеет дополнительные средства для прямой работы с памятью (*указатели*). На этом языке в 70-е годы написано множество прикладных и системных программ и ряд известных операционных систем (Unix).

Некоторые языки, например, Java и C#, находятся между компилируемыми и интерпретируемыми. А именно, программа компилируется не в машинный язык, а в машинно-независимый код низкого уровня, байт-код. Далее байт-код выполняется виртуальной машиной. Для выполнения байт-кода обычно используется интерпретация, хотя отдельные его части для ускорения работы программы могут быть транслированы в машинный код непосредственно во время выполнения программы по технологии компиляции «на лету» (Just-in-time compilation, JIT). Для Java байт-код исполняется виртуальной машиной Java (Java Virtual Machine, JVM), для C# — Common Language Runtime.

Способы изображения алгоритмов. Блок-схемы.

Схемы основных вычислительных процессов

Алгоритм, от имени учёного аль-Хорезми— точный набор инструкций, описывающих порядок действий исполнителя для достижения результата решения задачи за конечное время.

Алгоритм — это точная, однозначная, конечная последовательность действий, которую должен выполнить пользователь для достижения конкретной цели либо для решения конкретной задачи или группы задач.

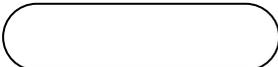
Алгоритмы и программы – необходимый элемент высшего образования не только как основа для понимания работы компьютера и разумного использования имеющихся программных средств, но и как азбука для планирования любой работы.

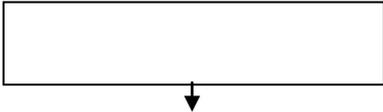
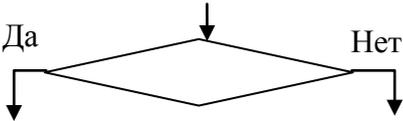
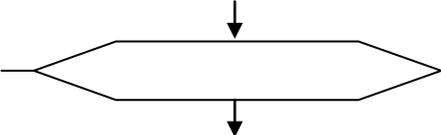
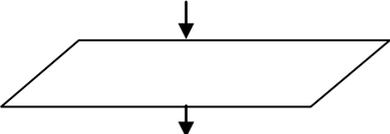
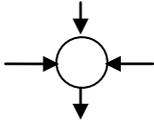
Под алгоритмом понимают последовательность действий, приводящих к решению задачи или достижению какого-либо результата. Другими словами, алгоритм – синоним слов план, программа действий. Применительно к ЭВМ принято говорить «алгоритм», когда речь идет вообще о плане решения, и «программа», когда задачу готовят к решению на ЭВМ.

Самое трудное в программировании – это не правила алгоритмического языка, которые при необходимости быстро запоминаются, а придумывание самого плана, т.е. идеи и последовательности расчета. Большую роль в процессе придумывания играет наглядность и четкость изображения плана и его частей. Наиболее распространен способ изображения алгоритмов, называемый «блок-схемы» и особенно удобный для изображения последовательной детализации плана.

Блок-схема — распространенный тип схем, описывающий алгоритмы или процессы, изображая шаги в виде блоков различной формы, соединенных между собой стрелками.

Другой способ – операторный, к которому относятся алгоритмические языки, отличается меньшей наглядностью и большей строгостью правил, что мешает на начальных этапах планирования.

Фигура	Характер действий
	Начало, конец алгоритма

	<p>Любые действия, чаще всего арифметические; блоки действий</p>
	<p>Проверка условий</p>
	<p>Начало, конец цикла</p>
	<p>Ввод, вывод данных</p>
	<p>Соединитель в разветвляющихся алгоритмах. Ссылка при переносе части алгоритма на другую страницу (хотя этого следует избегать)</p>

На блок-схеме действия различного характера изображают фигурами разной формы, а их последовательность показывают стрелками. Внутри фигур текстом или другими изобразительными средствами описывают действия, которые нужно выполнить.

Таблица 2

Основные фигуры международного стандарта для блок-схем

Блок-схемы – это изобразительное средство, помогающее выполнять последовательную детализацию больших и сложных планов работ, в том числе программ вычислительных работ для компьютеров.

Приведем схемы наиболее часто встречающихся в вычислительном моделировании частей алгоритмов (Рис. 2).

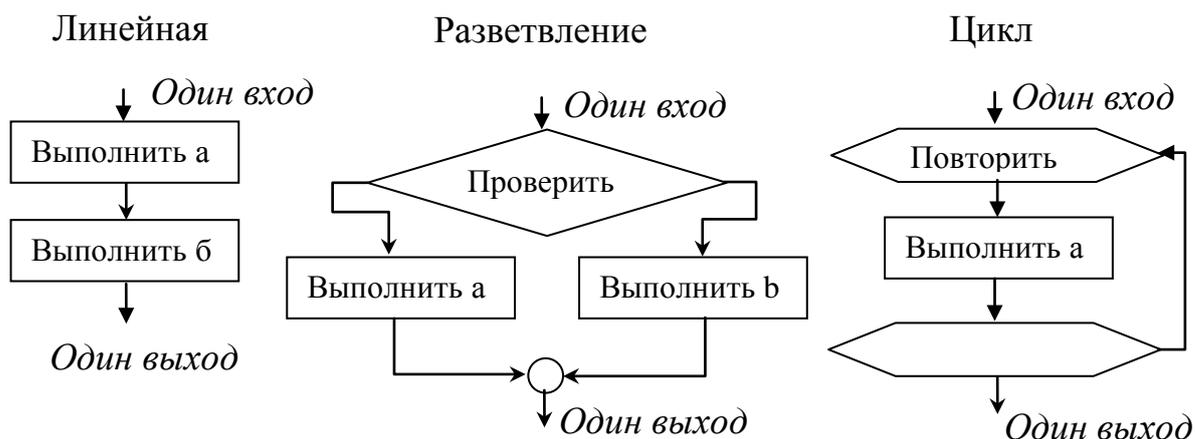


Рис. 2. Схемы простых вычислительных процессов

В схеме последовательных приближений предусмотрена необходимость задавать количество итераций (повторений) *kit*. Величину *kit* следует задавать достаточно большой, чтобы в случае, если процесс идет нормально, счет закончился после достижения нужной точности; но когда задана слишком высокая точность, или задача не имеет решения, или в программе есть ошибка, выполнение последовательных приближений может никогда не закончиться (программа заикнется), если не предусмотреть ограничение количества повторений, как это сделано в рассматриваемой блок-схеме.

Технологии программирования.

Структурное программирование.

Структурное программирование (Дал, Дейкстра, Хоор, 1972).

Пользователи первых ЭВМ (математики-программисты) стремились создавать максимально короткие и быстрые программы, чтобы максимально эффективно использовать весьма ограниченные вычислительные ресурсы (память и быстродействие), а заодно блеснуть талантом и эрудицией. Однако еще в 50-е годы было замечено, что такой путь редко ведет к успеху, так как он вынуждает усложнять алгоритм, а значит, увеличивать вероятность ошибок. Появилось выражение «Программа, как блюдо спагетти», описывающее ее запутанность, и шутливый «закон природы»: чем больше в программе найдено ошибок, тем больше их в ней осталось (действительно, ошибки обнаруживаются тогда, когда их наделали). Многие блестящие по замыслу программы начинали работать тогда, когда уже переставали быть нужными. Поэтому еще в начале развития вычислительной техники специалисты стали говорить о необходимости технологии программирования, обеспечивающей достаточно быструю и качественную реализацию программных проектов. Было разработано несколько таких технологий. Одна из наиболее популярных называется «Структурное программирование» [7].

Под этим условным названием объединяют четыре основных принципа – правила:

- **модульность**
- **нисходящее проектирование**
- **простота структуры**
- **сквозной структурный контроль.**

1. Модулями называют отдельные независимые составные части, из которых можно собирать разные конфигурации инженерного объекта. Например, в жилищном строительстве принято называть модулем отдельную квартиру (одно-, двух-, трех-, четырехкомнатную); в зависимости от цели проекта, из них собирают здания разной конфигурации (малосемейные, полнметражные, смешанные).

Чтобы составить большой вычислительный алгоритм, его делят на небольшие независимые составные части – программные модули (подпрограммы, процедуры). При этом следует придерживаться основного принципа: выделять составные части не по размеру программы, а по функциям, придумывая и четко формулируя, какую часть всей работы будет выполнять каждый модуль. Изображают модульную структуру программы в виде иерархической схемы (рис. 3).

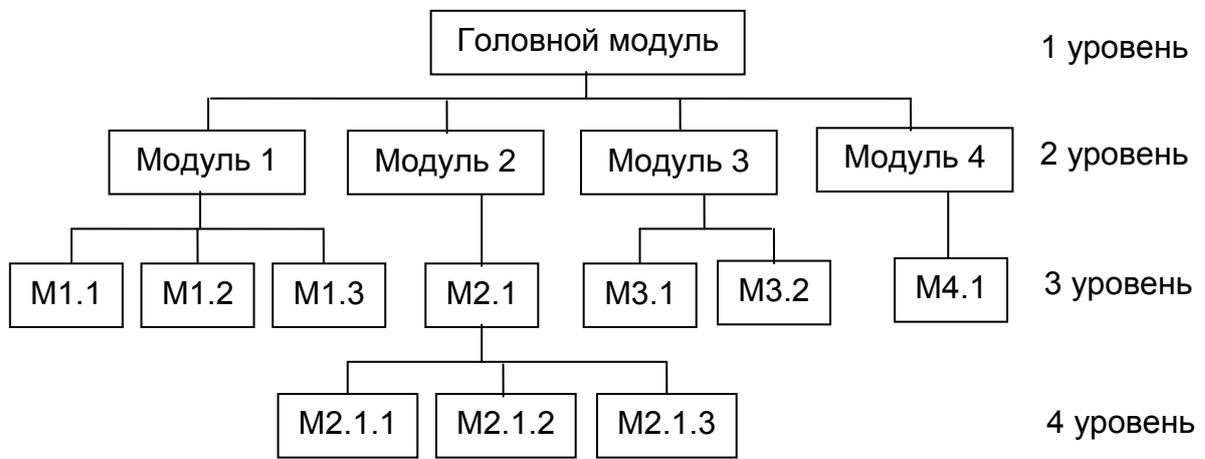


Рис. 3. Иерархия программных модулей

Иерархией называют последовательность подчинения в системах управления; например, в феодальном обществе верхнее звено – король, ему подчиняются феодалы первого уровня, им – второго и т.д. Причем «вассал моего вассала – не мой вассал», т.е. иерархические системы наиболее просты и надежны, в них все связи вертикальные. Причем здесь управление? – А вспомните, что делает программа в компьютере: она управляет вычислительным процессом, и должна делать это надежно.

В иерархической схеме программы всегда присутствует (на первом уровне иерархии) головной программный модуль, основная функция которого – распределять вычислительную работу, обращаясь по очереди к каждому модулю второго уровня, передавая ему необходимую информацию, принимая от него результаты и формируя из них данные для следующего модуля второго уровня. Если какой-либо модуль второго уровня получился слишком сложным, из него выделяют модули третьего уровня, выполняющие хорошо определенные части его сложной работы. И т.д.; в большом проекте могут быть десятки уровней иерархии. В этом случае на первой странице рисуют укрупненную схему иерархии, а далее детализируют иерархии отдельных модулей.

Все современные системы программирования обеспечивают реализацию модульности. В частности, в Vb модулями являются подпрограммы (т.е. процедуры и функции), которые создаются как отдельные независимые части всей программной модели.

2. Нисходящее проектирование, т.е. проектирование сверху вниз, имеет два смысла: *вниз по уровням иерархии модулей и последовательная детализация* каждого модуля, от общего плана к частным мелким деталям.

Первое из этих правил рекомендует не разрабатывать модуль нижнего уровня раньше своего вышестоящего, так как необходимо согласовывать формы представления исходных данных, передаваемых сверху вниз, и результатов, возвращаемых снизу вверх. Характерный пример: модуль верхнего уровня обычно формирует расчетную модель и составляет систему уравнений, а ее решение передает на нижний уровень. Если неизвестных немного, то система уравнений представляется таблицей (двумерным массивом) коэффициентов при них. Большую систему уравнений обычно представляют списком ненулевых коэффициентов с номерами, так как их значительно меньше, чем нулевых; но алгоритм усложняется, потому что в процессе решения некоторые нулевые коэффициенты становятся ненулевыми.

Второе правило рекомендует весь процесс обработки данных, предназначенный модулю, делить на крупные блоки, четко определяя функцию, исходные данные и результаты каждого из них, и только после этого делить их на более мелкие и т.д.

3. Под простой структурой программы подразумевают ее блок-схему, которая не должна быть запутанной и должна легко просчитываться, проверяться при простых исходных данных. Этого добиваются за счет использования при детализации небольшого стандартного набора простых блок-схем (рис. 3). Простым считают алгоритм, который можно представить как последовательную детализацию в виде таких схем – простых блоков с одним входом и одним выходом (например, рис. 4).

4. Сквозной структурный контроль означает для больших проектов необходимость коллективной работы, документирования всех этапов и взаимной проверки на всех этапах проектирования в соответствии с первыми тремя принципами. Сценарий примерно такой. Несколько единомышленников берутся за создание сложного программного продукта. Они собираются, обсуждают план, определяют (предварительно, вчерне) крупные программные модули и структуры данных, которыми они обмениваются, распределяют между собой и расходятся их проектировать. По мере готовности отдельных частей каждый участник собирает сессию, на которой показывает коллегам задокументированные по принципам 1 – 3 структуры

данных и программ и записывает все замечания. При этом достигаются две цели: напарники сжмим глазом видят недостатки и ошибки, и происходит согласование структур данных и взаимодействующих программ всех участников.

Стремясь к наибольшей простоте и наглядности, участники такого проекта получают в короткие сроки программный продукт, отличающийся большой надежностью и дающий отдачу, хотя, может быть, и не самый эффективный по быстрдействию и экономии ресурсов ЭВМ. В нашем веке вычислительные ресурсы компьютеров неизмеримо выросли, и такой подход еще более актуален.

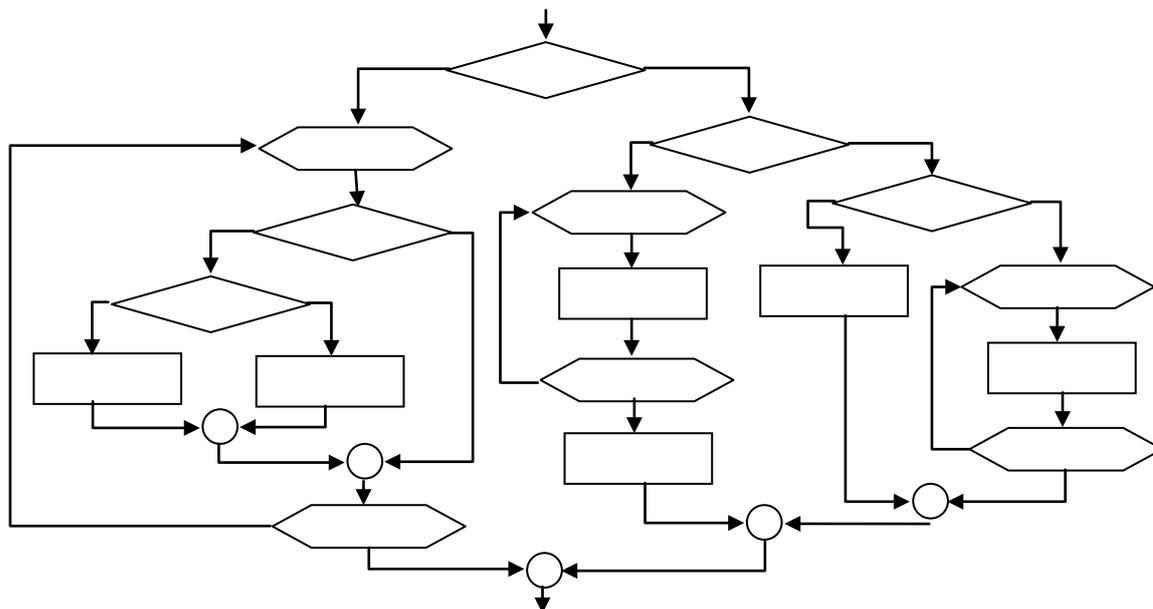


Рис. 4. Простая схема вычислительного процесса

Объектно-ориентированное программирование

Объектно-ориентированное программирование (Гради Буч, 1981) .

Структурное программирование (1970-е годы) – относительно ранняя технология компьютерного моделирования систем, основанная на анализе и моделирование *процесса* вычислений. В последующие десять лет, сталкиваясь с все более сложными задачами, специалисты в области программирования и инженерного проектировании осознали, что моделируемая система, *во-первых*, состоит из элементов (объектов), обладающих различными свойствами, а *во-вторых*, из процессов, происходящих между объектами. Сложную систему можно понять и смоделировать, если ее представить как состоящую из более простых подсистем, которые в свою очередь состоят из еще более простых подсистем и т.д. В качестве альтернативы программированию процессов появился метод потоков данных, в котором обработку данных выполняют по мере их появления (по аналогии с производством на заводах). Такая точка зрения имеет преимущества в задачах производства и коммерции, в управлении базами данных.

Новые точки зрения и потребности в 80-х годах были объединены в новую более универсальную технологию создания новых систем «Объектно-ориентированное проектирование». Этот процесс проектирования основан на нескольких принципах:

- абстрагирование
- *Инкапсуляция*
- *Наследование*
- Модульность
- Иерархия
- Типизация
- Параллелизм
- Устойчивость .

Компьютерную модель представляют как *объект*, характеризующийся набором *свойств*, выраженных числами или наборами чисел, и процедурами (методами), которые задают или изменяют эти свойства. В сложной системе некоторые свойства являются более мелкими объектами, которые в свою очередь могут состоять из еще более мелких объектов и т.д., образуя иерархию объектов. Такая инкапсуляция (объединение) в одном объекте числовых данных и методов их обработки позволяет проектировщику (в том числе программисту) разделить необозримую целиком сложную систему на более мелкие подсистемы и разработать каждую из них в отдельности. **Модель системы – это не программа (как в других технологиях), а совокупность объектов и их взаимодействий** (рис. 5).

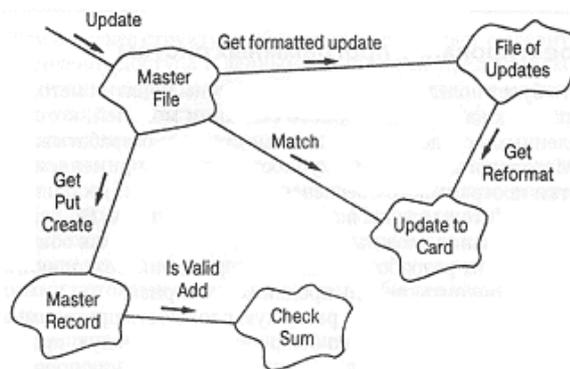


Рис. 5. Объектно-ориентированная декомпозиция

В современных ОО языках используются методы:

Наследование. Создание нового класса объектов путем добавления новых элементов (методов). В данный момент ОО языки позволяют выполнять множественное наследование, т. е. объединять в одном классе возможности нескольких других классов.

Инкапсуляция. Соккрытие данных, которое (при грамотной реализации) позволяет вносить изменения в части программы безболезненно для других её частей. Что существенно упрощает сопровождение и модернизацию ПО.

Полиморфизм. При полиморфизме некоторые части (методы) родительского класса заменяются новыми, реализующими специфические для данного потомка действия. Таким образом, интерфейс классов остаётся прежним, а реализация методов с одинаковым названием и набором параметров различается. С понятием «Полиморфизм» тесно связано понятие «Позднего связывания».

Типизация. Позволяет устранить многие ошибки на момент компиляции, операции проводятся только над объектами подходящего типа.

Лекция 3

Программное обеспечение компьютеров

Существует разделение вычислительных машин на две составляющие: аппаратную и программную. Программная часть называется программным обеспечением ЭВМ.

В области вычислительной техники и программирования программное обеспечение — это совокупность всей информации, данных и программ, которые обрабатываются компьютерными системами.

Программы — это упорядоченные последовательности команд. Конечная цель любой компьютерной программы — управление аппаратными средствами. Даже если на первый взгляд программа никак не взаимодействует с оборудованием, не требует никакого ввода данных с устройств ввода и не осуществляет вывод данных на устройства вывода, все равно ее работа основана на управлении аппаратными устройствами компьютера.

Состав программного обеспечения вычислительной системы называют программной конфигурацией. Между программами, как и между физическими узлами и блоками существует взаимосвязь — многие программы работают, опираясь на другие программы более низкого уровня, то есть мы можем говорить о межпрограммном интерфейсе.

По назначению ПО разделяется на системное, прикладное и инструментальное.

Одна часть - это комплекс программных средств предназначенных для того, чтобы на вычислительной машине можно было организовать выполнение программ. Вторая - множество тех программ, которые нацелены на решение конкретных задач. *Эти две части программного обеспечения принято соответственно называть Системными программами и прикладными программами.*

Системное ПО.

- **Загрузчик операционной системы**
- **Операционные системы** — общего назначения, реального времени, сетевые ОС, встраиваемые. Основная задача таких программ - планирование вычислительного процесса, распоряжение ресурсами машины, организация взаимодействия отдельных процессов, протекающих в машине во время выполнения программ. К этим программам примыкают программные системы, обеспечивающие отображение информации в удобном для пользователя виде (например, на дисплее), диалоговые программы для общения на естественном языке, а также системы трансляции (трансляторы), обеспечивающие перевод программ с языков программирования в машинные коды.

- **Сервисные программы**, отладчики, диагностические программы, программы для борьбы с компьютерными вирусами и др. Эти программы облегчают пользователю взаимодействие с машиной.
- **Драйверы устройств** Драйвер (англ. driver) (множественное число драйверы, вариант драйвера разговорный и профессиональный) — компьютерная программа, с помощью которой другая программа (обычно операционная система) получает доступ к аппаратному обеспечению стандартным образом. В общем случае для использования каждого устройства, подключенного к компьютеру, необходим специальный драйвер. Обычно с операционными системами поставляются драйверы для ключевых компонентов аппаратного обеспечения, без которых система не сможет работать. Однако для более специфических устройств (таких, как графическая плата или принтер) могут потребоваться специальные драйверы, обычно предоставляемые производителем устройства.
- **Программы обеспечения работы в сети.** Эти программы реализуют протоколы обмена информацией между машинами, работу с базами данных, телеобработку данных.
- **Программные средства защиты:**
 - Кривошлюзы
 - Средства аутентификации
 - Средства мониторинга и аудита
 - Сканеры защищенности
 - Средства разграничения доступа
 - Системы криптографической защиты, шифрования и ЭЦП
 - Антивирусные программы
 - Антиспамовые программы
 - Межсетевой экран

Инструментальное ПО.

- **Средства разработки программного обеспечения** — среды разработки (в том числе RAD), SDK.
- **Системы управления базами данных (СУБД)** — реляционные (например, DB2, Informix, Interbase, Firebird, Microsoft SQL Server, MySQL, Oracle Database PostgreSQL), объектно-ориентированные, .

Базами данных называют огромные массивы данных, организованных в табличные структуры. Основными функциями систем управления базами данных являются:

- создание пустой (незаполненной!) структуры базы данных.
- предоставление средств ее заполнения или импорта данных из таблиц другой базы;
- обеспечение возможности доступа к данным, а также предоставление средств поиска и фильтрации

Многие системы управления базами данных (СУБД) дополнительно предоставляют возможности проведения простейшего анализа данных и их обработки. В результате возможно создание новых таблиц баз данных на основе имеющихся. В связи с широким распространением сетевых технологий к современным системам управления базами данных предъявляется также требование возможности работы с удаленными и распределенными ресурсами, находящимися на серверах всемирной компьютерной сети. Наиболее распространенными являются Microsoft ACCESS.

Прикладные программы.

Прикладные программы удобно разделить на следующие классы:

1. Офисные приложения

- **Текстовые редакторы** Основные функции этого класса прикладных программ заключаются в вводе и редактировании текстовых данных. Дополнительные функции состоят в автоматизации процессов ввода и редактирования. Для операций ввода - вывода и сохранения данных текстовые редакторы используют системное программное обеспечение. Впрочем, это характерно и для всех прочих видов прикладных программ, и в дальнейшем мы не будем специально указывать на этот факт.
- **Текстовые процессоры** Основное отличие текстовых процессоров от текстовых редакторов в том, что они позволяют не только вводить и редактировать текст, но и форматировать его, то есть оформлять. Соответственно, к основным средствам текстовых процессоров относятся средства обеспечения взаимодействия текста, графики, таблиц и других объектов, составляющих итоговый документ, а к дополнительным — средства автоматизации процесса форматирования. Наиболее популярный текстовый процессор - Microsoft Word
- **Табличные процессоры** Электронные таблицы (ЭТ) предоставляют комплексные средства для хранения различных типов данных и их обработки. В некоторой степени они аналогичны системам управления базами данных, но основной акцент смещен не на хранение массивов данных и обеспечение к ним доступа, а на преобразование данных, причем в соответствии с их внутренним содержанием. Наиболее распространены ЭТ Excel.

- **Редакторы презентаций**
- 2. Системы проектирования и производства**
- Системы автоматизированного проектирования (САПР, CAD-системы) Система автоматизации проектных работ (САПР) или CAD (англ. Computer-Aided Design) — организационно-техническая система, предназначенная для выполнения проектной деятельности с применением вычислительной техники, позволяющая создавать конструкторскую и/или технологическую документацию.
 - PDM-системы. PDM-система (Product Data Management — система управления данными об изделии) — организационно-техническая система обеспечивающая управление всей информацией об изделии. При этом в качестве изделий могут рассматриваться различные сложные технические объекты (корабли и автомобили, самолеты и ракеты, компьютерные сети и др.). PDM-системы являются неотъемлемой частью PLM-систем.
 - PLM-системы PLM (сокр. от англ. Product Lifecycle Management) — технология управления жизненным циклом изделий. Организационно-техническая система обеспечивающая управление всей информацией об изделии и связанных с ним процессах на протяжении всего его жизненного цикла, начиная с проектирования и производства до снятия с эксплуатации. При этом в качестве изделий могут рассматриваться различные сложные технические объекты (корабли и автомобили, самолеты и ракеты, компьютерные сети и др.).
 - АСУТП (Системы SCADA) Автоматизированная система управления технологическим процессом (АСУ ТП) — комплекс программных и технических средств, предназначенный для управления технологическим оборудованием на предприятиях.
 - АСТПП (Системы MES) MES (сокр. от англ. Manufacturing Execution System) — исполнительная система производства. Системы такого класса решают задачи синхронизации, координируют, анализируют и оптимизируют выпуск продукции в рамках какого-либо производства.
- 3. Мультимедиа**
- Компьютерные игры
 - Музыкальные редакторы.
 - Графические редакторы. Графический редактор — программа (или пакет программ), позволяющая создавать и редактировать двумерные изображения с помощью компьютера.
 - Видео редакторы
 - Мультимедиа проигрыватели
 - Редакторы HTML (Web-редакторы). Это особый класс редакторов, объединяющих в себе свойства текстовых и графических редакторов. Они предназначены для создания и редактирования так называемых ЖЕВ-документов (Web-страниц Интернета). Web-документы — это электронные документы, при подготовке которых следует учитывать ряд особенностей, связанных с приемом/передачей информации в Международной компьютерной сети Интернет. Теоретически для создания Web-документов можно использовать обычные текстовые редакторы и процессоры, а также некоторые из графических редакторов векторной графики, но Web-редакторы обладают рядом полезных функций, повышающих производительность труда web-дизайнеров. Программы этого класса можно также эффективно использовать для подготовки электронных документов и мультимедийных изданий.
- 4. Клиенты для доступа к интернет-сервисам:**
- электронная почта
 - веб
 - мгновенная передача сообщений
 - чат-каналы
 - IP-телефония
 - P2P обмен файлами
 - потоковое вещание
 - Банк-клиент
- 5. Корпоративные информационные системы**
- Бухгалтерские программы
 - Системы Управления проектами (Project Management)
 - Системы автоматизации документооборота (EDM-системы)
 - Системы управления архивами документов (DWM-системы)
- 6. Экспертные системы.** Предназначены для анализа данных, содержащихся в базах знаний, и выдачи рекомендаций по запросу пользователя. Такие системы применяют в тех случаях, когда исходные данные хорошо формализуются, но для принятия решения требуются обширные специальные знания. Характерными областями использования

экспертных систем являются юриспруденция, медицина, фармакология, химия. По совокупности признаков заболевания медицинские экспертные системы помогают установить диагноз и назначить лекарства, дозировку и программу лечебного курса. По совокупности признаков события юридические экспертные системы могут дать правовую оценку и предложить порядок действий, как для обвиняющей стороны, так и для защищающейся. С использованием экспертных систем связана особая область научно-технической, деятельности, называемая инженерией знаний. Инженеры знаний — это специалисты особой квалификации, выступающие в качестве промежуточного звена между разработчиками экспертной системы (программистами) и ведущими специалистами в конкретных областях науки и техники (экспертами).

Классификация ПО по условиям распространения и использования.

Лицензия информирует пользователя о том, на каких условиях распространяется данное программное обеспечение.

Наиболее часто встречающиеся следующие виды лицензий:

Commercial software - коммерческое программное обеспечение — программное обеспечение, созданное коммерческой организацией с целью получения прибыли от его использования другими, например, путем продажи экземпляров. Прежде чем работать с такой программой её надо купить.

FreeWare - абсолютно бесплатное программное обеспечение без каких-либо ограничений по функциональности и времени работы.

Free Software Definition - свободное программное обеспечение — широкий спектр программных решений, в которых права пользователя на неограниченные установку, запуск, а также свободное использование, изучение, распространение и изменение программ защищены юридически авторскими правами при помощи свободных лицензий (имеется в виду свобода копировать, распространять и изменять его).

ShareWare - условно бесплатное программное обеспечение. За использование такой программы Вы должны заплатить деньги. До тех пор, пока Вы этого не сделаете, у Вас могут возникнуть, например, такие проблемы:

программа не будет позволять использовать все свои возможности;

программа запустится только несколько раз;

программа будет обрабатывать ограниченное количество файлов;

Trial - условно-бесплатная программа. Не имеет ограничений в функциональности, но имеет ограниченный срок работы.

Demo - демонстрационная версия программного обеспечения. Дает представление об интерфейсе и функциональности программы. Попробовать работать с такой программой удастся не всегда, поскольку она может представлять собой видеоролик.

Adware - бесплатное программное обеспечение. За использование такой программы пользователь должен не деньги заплатить, а смотреть рекламу. Деньги автору будет платить рекламодатель.

Donationware - «пожертвование», за такое программное обеспечение платят те, кому оно понравилось и столько, сколько они могут. Никаких ограничений в функциональности такого программного обеспечения нет.

Postcardware - за использование такого программного обеспечения надо написать письмо их авторам. Обычно авторам интересно кто, где, как и для чего использует их программу.

Классификация вредоносных

В наше время каждый знает, что такое компьютерный вирус. Однако не каждый знает, что компьютерные вирусы - только часть вредоносного программного обеспечения. На самом деле не каждая программа, которая может негативно повлиять на работу компьютера, является вирусом.

Как правило, у каждой антивирусной корпорации есть своя классификация, согласно которой эксперты ее лаборатории определяют принадлежность нового вредоносного кода. У разных корпораций один и тот же код будет иметь разные названия. Именно разность классификаций тому виной.

Попытка создать единую систему классификации и именования вирусов была предпринята на встрече CARO (Computer Antivirus Researchers Organization - организация исследователей компьютерных вирусов) в 1991 году.

Кроме того, в 2005 году американская организация US-CERT, занимающаяся проблемами в области компьютерной безопасности, предложила использовать стандартные названия для интернет-червей и других вредоносных программ. Члены US-CERT назвали свою программу "Общая классификация вредоносных программ" (CME). Цель программы - не вводить пользователей в заблуждение, используя разные названия для одних и тех же вирусов.

Классификация вредоносных программ по версии Microsoft:

В Microsoft разделяют все вредоносные программы (Malware) на:

1. Viruses (вирусы и черви):
 - Trojan horse
 - Virus
 - Worm
2. Spyware (шпионские программы).

Классификация вредоносных программ по версии Лаборатории Касперского (опубликована в вирусной энциклопедии):

1. Сетевые черви

Сетевой червь — разновидность самовоспроизводящихся компьютерных программ, распространяющихся в локальных и глобальных компьютерных сетях. В отличие от других типов компьютерных вирусов червь является самостоятельной программой.

Механизмы распространения

Черви могут использовать различные механизмы («векторы») распространения. Некоторые черви требуют определенного действия пользователя для распространения (например, открытия инфицированного сообщения в клиенте электронной почты). Другие черви могут распространяться автономно, выбирая и атакуя компьютеры в полностью автоматическом режиме. Иногда встречаются черви с целым набором различных векторов распространения, стратегий выбора жертвы, и даже эксплойтов под различные операционные системы.

Зачастую черви даже безо всякой полезной нагрузки перегружают и временно выводят из строя сети только за счёт интенсивного распространения. Типичная осмысленная полезная нагрузка может заключаться в порче файлов на компьютере-жертве (в том числе, изменение веб-страниц, «deface»), заранее запрограммированной DDoS-атаке с компьютеров жертв на отдельный веб-сервер, или бэкдор для удалённого контроля над компьютером-жертвой. Часто встречаются случаи, когда новый вирус эксплуатирует бэкдоры, оставленные старым.

К данной категории относятся программы, распространяющие свои копии по локальным и/или глобальным сетям с целью:

- проникновения на удаленные компьютеры;
- запуска своей копии на удаленном компьютере;
- дальнейшего распространения на другие компьютеры в сети.

К ним относятся:

- Email-Worm - почтовые черви
- IM-Worm - черви, использующие интернет-пейджеры
- IRC-Worm - черви в IRC-каналах
- Net-Worm - прочие сетевые черви
- P2P-Worm - черви для файлообменных сетей

2. Классические компьютерные вирусы

Компьютерный вирус — разновидность компьютерных программ, отличительной особенностью которой является способность к размножению (саморепликация). В дополнение к этому вирусы могут повредить или полностью уничтожить все файлы и данные, подконтрольные пользователю, от имени которого была запущена заражённая программа, а также повредить или даже уничтожить операционную систему со всеми файлами в целом.

Неспециалисты к компьютерным вирусам иногда причисляют и другие виды вредоносных программ, такие как трояны, программы-шпионы и даже спам. Известны десятки тысяч компьютерных вирусов, которые распространяются через Интернет по всему миру, организуя вирусные эпидемии.

Вирусы распространяются, внедряя себя в исполняемый код других программ или же заменяя собой другие программы. Какое-то время даже считалось, что, являясь программой, вирус может

заразить только программу — какое угодно изменение не-программы является не заражением, а просто повреждением данных. Подразумевалось, что такие копии вируса не получают управления, будучи информацией, не используемой процессором в качестве инструкций. Так, например неформатированный текст не мог бы быть переносчиком вируса.

Однако, позднее злоумышленники добились, что вирусным поведением может обладать не только исполняемый код, содержащий машинный код процессора. Были написаны вирусы на языке пакетных файлов. Потом появились макровирусы, внедряющиеся через макросы в документы таких программ, как Microsoft Word и Excel.

Некоторое время спустя взломщики создали вирусы, использующие уязвимости в популярном программном обеспечении (например, Adobe Photoshop, Internet Explorer, Outlook), в общем случае обрабатывающем обычные данные. Вирусы стали распространяться посредством внедрения в последовательности данных (например, картинки, тексты, и т. д.) специального кода, использующего уязвимости программного обеспечения.

Ныне существует немало разновидностей вирусов, различающихся по способу распространения и функциональности. Если изначально вирусы распространялись на дискетах и других носителях, то сейчас доминируют вирусы, распространяющиеся через Интернет. Растёт и функциональность вирусов, которую они перенимают от других видов программ: руткитов, бэкдоров (создают «чёрный ход» в систему), кейлоггеров (регистрация активности пользователей), программ-шпионов (крадут пароли от банковских счётов и номера кредитных карт), ботнетов (превращают заражённые компьютеры в станции по рассылке спама или в часть компьютерных сетей, занимающихся спамом и прочей противоправной активностью).

Создание и распространение компьютерных вирусов и вредоносных программ преследуется в России согласно Уголовному Кодексу РФ (глава 28, статья 273).

К данной категории относятся программы, распространяющие свои копии по ресурсам локального компьютера с целью:

- последующего запуска своего кода при каких-либо действиях пользователя;
- дальнейшего внедрения в другие ресурсы компьютера.

В отличие от червей, вирусы не используют сетевых сервисов для проникновения на другие компьютеры.

Типы компьютерных вирусов различаются между собой по следующим основным признакам:

- среда обитания;
- способ заражения.

К ним относятся:

1) Среда обитания.

По среде обитания вирусы можно разделить на:

- файловые;
- загрузочные;
- макро;
- скриптовые.

2) Способ заражения.

- Файловые вирусы:

По способу заражения файлов вирусы делятся на:

Перезаписывающие (overwriting).

Паразитические (parasitic):

внедрение вируса в начало файла,
внедрение вируса в конец файла,
внедрение вируса в середину файла,
вирусы без точки входа.

Вирусы-компаньоны (companion).

Вирусы-ссылки (link).

Вирусы, заражающие объектные модули (OBJ).

Вирусы, заражающие библиотеки компиляторов (LIB).

Вирусы, заражающие исходные тексты программ.

- Загрузочные вирусы:

Вирусы заражают загрузочный (boot) сектор гибкого диска.

Вирусы заражают boot-сектор винчестера.

Вирусы заражают Master Boot Record (MBR) винчестера.

- Макро-вирусы:

В вирусе присутствует авто-макрос (авто-функция).

В вирусе переопределен один из стандартных системных макросов (ассоциированный с каким-либо пунктом меню).

В вирусе макрос вируса вызывается автоматически при нажатии на какую-либо клавишу или комбинацию клавиш.

- *Скрипт-вирусы*

Вирусы заражают другие скрипт-программы (командные и служебные файлы MS Windows или Linux).

Вирусы являются частями многокомпонентных вирусов.

Вирусы заражают файлы других форматов (например, HTML), если в них возможно выполнение скриптов.

3. Троянские программы

В данную категорию входят программы, осуществляющие различные несанкционированные пользователем действия: сбор информации и ее передачу злоумышленнику, ее разрушение или злонамеренную модификацию, нарушение работоспособности компьютера, использование ресурсов компьютера в неблагоприятных целях.

Троянская программа (также — троян, троянец, троянский конь, трой) — вредоносная программа, проникающая на компьютер под видом безвредной — кодека, скринсейвера, хакерского ПО и т. д. «Троянские кони» не имеют собственного механизма распространения, и этим отличаются от вирусов, которые распространяются, прикрепляя себя к безобидному ПО или документам, и «червей», которые копируют себя по сети. Впрочем, троянская программа может нести вирусное тело — тогда запустивший троянца превращается в очаг «заразы».

Троянские программы крайне просты в написании: простейшие из них состоят из нескольких десятков строк кода на Visual Basic или C++.

Название «троянская программа» происходит от названия «троянский конь» — деревянный конь, по легенде, подаренный древними греками жителям Трои, внутри которого прятались воины, впоследствии открывшие завоевателям ворота города. Такое название, прежде всего, отражает скрытность и потенциальное коварство истинных замыслов разработчика программы.

Троянские программы помещаются злоумышленником на открытые ресурсы (файл-серверы, открытые для записи накопители самого компьютера), носители информации или присылаются с помощью служб обмена сообщениями (например, электронной почтой) из расчета на их запуск на конкретном, входящем в определенный круг или произвольном «целевом» компьютере.

Иногда использование троянцев является лишь частью спланированной многоступенчатой атаки на определенные компьютеры, сети или ресурсы (в том числе, третьи).

Троянские программы различаются между собой по тем действиям, которые они производят на зараженном компьютере.

К ним относятся:

- Backdoor - троянские утилиты удаленного администрирования
- Trojan-PSW - воровство паролей
- Trojan-Clicker - интернет-кликеры
- Trojan-Downloader - доставка прочих вредоносных программ
- Trojan-Dropper - инсталляторы прочих вредоносных программ
- Trojan-Proxy - троянские прокси-сервера
- Trojan-Spy - шпионские программы
- Trojan - прочие троянские программы
- Rootkit - сокрытие присутствия в операционной системе
- ArcBomb - «бомбы» в архивах
- Trojan-Notifier - оповещение об успешной атаке

4. Хакерские утилиты и прочие вредоносные программы

К данной категории относятся:

- утилиты автоматизации создания вирусов, червей и троянских программ (конструкторы);
- программные библиотеки, разработанные для создания вредоносного ПО;
- хакерские утилиты скрытия кода зараженных файлов от антивирусной проверки (шифровальщики файлов);
- «злые шутки», затрудняющие работу с компьютером;
- программы, сообщающие пользователю заведомо ложную информацию о своих действиях в системе;
- прочие программы, тем или иным способом намеренно наносящие прямой или косвенный ущерб данному или удаленным компьютерам.

К ним относятся:

- DoS, DDoS - сетевые атаки
- Exploit, HackTool - взломщики удаленных компьютеров
- Flooder - "замусоривание" сети
- Constructor - конструкторы вирусов и троянских программ

- Nuker - фатальные сетевые атаки
- Bad-Joke, Ноах - злые шутки, введение пользователя в заблуждение
- FileCryptor, PolyCryptor - скрытие от антивирусных программ
- PolyEngine - полиморфные генераторы
- VirTool - утилиты, предназначенные для облегчения написания компьютерных вирусов.

Антивирусная программа

Антивирусная программа (антивирус) — программа для обнаружения компьютерных вирусов и лечения инфицированных файлов, а также для профилактики — предотвращения заражения файлов или операционной системы вредоносным кодом (например, с помощью вакцинации). Многие современные антивирусы расширяют набор своих функций, позволяя обнаруживать и удалять также троянские и прочие вредоносные программы. Идёт и процесс интеграции антивирусных функций в другие программы — например, файрволы.

Первые наиболее простые антивирусные программы появились почти сразу после появления вирусов. Сейчас разработкой антивирусов занимаются крупные компании. Как и у создателей вирусов, в этой сфере также сформировались оригинальные приёмы — но уже для поиска и борьбы с вирусами. Современные антивирусные программы могут обнаруживать сотни тысяч вирусов, но ни одна из них не даст 100 % защиты.

Антивирусное программное обеспечение состоит из подпрограмм, которые пытаются обнаружить, предотвратить размножение и удалить компьютерные вирусы и другие вредоносные программы.

Классификация антивирусов

Евгений Касперский в 1992 году использовал следующую классификацию антивирусов в зависимости от их принципа действия (определяющего функциональность):

Сканеры (устаревший вариант — «полифаги») — определяют наличие вируса по базе сигнатур, хранящей сигнатуры (или их контрольные суммы) вирусов. Их эффективность определяется актуальностью вирусной базы и наличием эвристического

Ревизоры (класс, близкий к IDS) — запоминают состояние файловой системы, что делает в дальнейшем возможным анализ изменений.

Сторожа (мониторы) — отслеживают потенциально опасные операции, выдавая пользователю соответствующий запрос на разрешение/запрещение операции.

Вакцины — изменяют прививаемый файл таким образом, чтобы вирус, против которого делается прививка, уже считал файл заражённым..

Современные антивирусы сочетают все вышесказанные функции.

Часто используемые Антивирусные программы:

Антивирус Касперского (KAV, KIS)— Россия

Dr.Web — Россия

Eset NOD32 — Словакия

Panda Software — Испания

Symantec (Norton Internet Security, Norton Personal Firewall) — США

AVG (AVG Antivirus Free Edition) (GriSoft) — Чехия

Avira (AntiVir Personal Edition - Free Antivirus)— Германия .