

МИНИСТЕРСТВО ОБЩЕГО И ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИЯ
РОССИЙСКОЙ ФЕДЕРАЦИИ
САМАРСКИЙ ГОСУДАРСТВЕННЫЙ АЭРОКОСМИЧЕСКИЙ УНИВЕРСИТЕТ
имени академика С. П. КОРОЛЕВА

*А. В. Баландин Л. С. Зеленко
О. П. Солдатова*

КУРС ЛЕКЦИЙ ПО ИНФОРМАТИКЕ

для студентов заочной формы обучения

Самара 1999

УДК 681.3

Курс лекций по информатике/ А. В. Баландин, Л. С. Зеленко, О. П. Солдатова. Самарский гос. аэрокосмический ун-т. Самара, 1999, 58 с.

ISBN 5-7217-0028-9

Предназначен для студентов заочной формы обучения по специальности 22.02 - "Автоматизированные системы обработки информации и управления" - и включает в себя разделы, связанные с методами сбора, обработки, хранения и управления информацией; рассмотрены основные структурные элементы персонального компьютера и определены их аппаратные, информационные и логические взаимосвязи; подробно рассмотрены вопросы, связанные с решением прикладных задач; дана классификация программных продуктов.

Данный курс лекций будет также полезен и студентам других специальностей, изучающих курс информатики и обучающимся как по очной, так и заочной форме обучения. Разработан на кафедре информационных систем и технологий.

Печатается по решению редакционно-издательского совета Самарского государственного аэрокосмического университета имени академика С. П. Королёва

Рецензент канд. техн. наук, доц. Л. А. Ж а р и н о в а

ISBN 5-7217-0028-9

© Самарский государственный аэрокосмический университет, 1999

Оглавление

1. ПРЕДМЕТ ИНФОРМАТИКИ И ЕЁ СВЯЗЬ С ДРУГИМИ НАУКАМИ	4
1.1. Информация: виды и свойства.....	5
1.2. Семiotические аспекты информации. Информация и данные.....	6
1.3. Интеллектуальная деятельность с позиции информатики.....	8
2. ЭЛЕМЕНТЫ МОДЕЛИРОВАНИЯ И КОДИРОВАНИЯ ИНФОРМАЦИИ. 10	
2.1. Информационная модель источника информации. Основные определения и понятия.....	10
2.2. Кодирование информации.....	13
2.2.1. Практически значимые системы кодирования.....	14
2.2.2. Системы счисления.....	18
2.2.3. Правила вычислений в двоичной системе счисления.....	22
2.3. Элементы алгебры логики.....	24
3. МЕТОДЫ РАБОТЫ С ПРОЦЕДУРНОЙ ИНФОРМАЦИЕЙ.....	26
3.1. Основные этапы решения прикладных задач.....	26
3.2. Основные формы представления алгоритмов.....	28
3.3. Основные характеристики программных продуктов.....	31
3.3.1. Классификация программных продуктов.....	35
4. КОМПЬЮТЕР КАК УНИВЕРСАЛЬНАЯ ИНФОРМАЦИОННАЯ СИСТЕМА	38
4.1. Структурная схема персонального компьютера.....	39
4.2. Элементы конструкции персонального компьютера.....	44
4.3. Устройства отображения информации.....	47
4.4. Устройства хранения информации.....	50

1. Предмет информатики и её связь с другими науками

Термин *информация* происходит от латинского *informatio*, что означает разъяснение, осведомление, изложение. С позиции материалистической философии информация есть отражение реального мира с помощью сведений (сообщений). *Сообщение* - это определенная *форма* представления информации. Например, сообщение может быть представлено в виде речи, текста, изображения, графиков, таблиц и т.п. Понятие информации тесно связано с понятием *информационный процесс*. Информационным называется процесс, возникающий в результате установления связи между двумя объектами материального мира, один из которых выступает в качестве источника (носителя) информации, а другой - в качестве её приёмника (потребителя).

При всем многообразии видов информационных процессов все они содержат три основные составляющие:

- 1) *восприятие информации* (рецепция, получение и сохранение) от источника информации;
- 2) *интерпретация информации* (преобразование, моделирование) в соответствии с потребностями и/или возможностями приёмника и построение адекватной информационной модели источника информации;
- 3) *коммуникация информации* (транспортировка) - пространственно-временная передача информации от источника к приёмнику.

Прокомментируем приведенные составляющие информационного процесса на примере следа, оставляемого на песке ступней ноги. В качестве источника информации выступает ступня, а в качестве приёмника - песок. Восприятие информации песком осуществляется в результате механического воздействия ступни на песок. Интерпретация информации заключается в приобретении песком рельефа ступни (информационная модель ступни). Отметим, что информационная модель ступни выражается по отношению к оригиналу в виде негативного изображения, если пользоваться фотографической терминологией. При этом

возможности интерпретации у мокрого песка и у сухого разные. В этой связи разной будет и степень адекватности информационной модели ступни (следа) своему оригиналу. След на сухом песке менее похож на оригинал, чем на мокром. Что касается коммуникации информации в данном примере, то она происходит в процессе приближения стопы к песку.

Методологической основой информатики является всеобщее свойство материи, называемое *отражением*. В материальных системах это свойство проявляется в различных формах в зависимости от сложности и уровня развития материальной системы. Высшими формами отражения являются *ощущение и сознание*. Наиболее простыми формами отражения являются процессы отражения в неорганической природе (результаты физического или химического воздействия).

Предметом информатики являются *свойства* объектов природы (материальных систем), обеспечивающие их способность быть носителями информации, источниками или приёмниками информации, а также *информационные процессы*, понимаемые в широком смысле как процессы получения, хранения и накопления, преобразования, транспортировки и представления информации, взятые по отдельности или в совокупности.

1.1. Информация: виды и свойства

С позиции информатики любая материальная система проявляется как информационный объект или через свою структуру (форму) или посредством своего действия (поведения, функционирования). В этой связи различают два вида информации - фактографическую и процедурную.

Фактографическая информация связана со свойствами материальной системы, отличающими ее от других материальных систем (информация о весе, росте, цвете, материале, рельефе, температуре и т.п.). *Процедурная информация* связана с поведением или функционированием материальной системы во времени (например, механическая система может выполнять во времени следующие действия: включилась, набирает обороты, движется к пункту А и т.п.).

Материальная система, рассматриваемая как информационный объект, играет роль, так называемого, *носителя информации*. Это связано с тем, что информация обладает таким важным свойством, как относительная независимость от материальной системы (среды), являющейся её носителем. Это свойство выражается в возможности *переноса (копирования)* информации с одного носителя на другой (коммуникация информации). Например, текст на бумаге можно отобразить в память компьютера, а оттуда - на экран видеотерминала. И на бумаге, и в памяти компьютера, и на экране будет одна и та же информация. Если перенос информации происходит на новый носитель с иными материальными свойствами, то в общем случае происходит искажение и/или потеря информации. Например, изображение на телевизионном экране соответствует изображению на исходном носителе с точностью до разрешающей способности телевизионного экрана.

1.2. Семиотические аспекты информации. Информация и данные

При реализации информационных процессов большую роль играют знаки и знаковые системы. Под *знаком* понимается материальный предмет (явление, событие), который служит представителем, заместителем другого предмета и используется для восприятия, хранения и передачи информации о последнем. Выбор формы знаков зависит от свойств материального носителя информации, который используется для восприятия информации. Например, на бумаге можно располагать письменную речь, состоящую из графических символов (знаков).

Знаки делятся на *символы* - которые отображают значение знака (указатель направления - стрелка, геральдический знак (герб)) и *диокритики* - которые не имеют прямой зависимости между формой и значением или такая связь утрачена в результате изменения формы знака или переноса его значения на другие сущности. Это - буквы русского алфавита и арабские цифры, знаки математических операций. Если проследить историю букв, то можно определить связь их современного начертания с определенным значением (у древних финикийцев, изобретателей фонетической системы письма, на основании которой созданы латинский алфавит

и кириллица, форма буквы - это стилизованное изображение предмета, название которого начинается с этой буквы). Часто знаки являются и символами, и диакритиками одновременно (знаки дорожного движения).

Определение отношений между знаками и тем, что они обозначают, представляют, замещают, относится к *семиотическим аспектам информации*. Даже в простейшем случае между знаком и его обозначением стоит субъективная смысловая интерпретация знака (семантика знака). Например, знак 5 можно интерпретировать как номер 5, 5 лет, 5 км, 5 книг и т.д.

В семиотике выделяют три аспекта: синтаксис, семантику и прагматику. *Синтаксис* знаковых систем занимается изучением самих знаков. *Семантика* изучает отношение между знаком и тем, что он представляет. Даже в простейшем случае между знаком и его обозначением есть субъект, который интерпретирует этот знак. С точки зрения семантики каждое сочетание знаков является *осмысленным* или *бессмысленным*, причем осмысленность может допускать и неправильные с точки зрения синтаксиса построения (детская речь). Однако истинность или ложность того или иного высказывания, то есть достоверность отражения действительности знаковой системой, относится к области прагматики. *Прагматика* изучает законы функционирования знаковой системы как средства коммуникации между субъектами. Семиотические аспекты информатики очевидны. Конкретные приложения семиотики находят свою реализацию в знаковых системах кодирования информации, в алгоритмических языках программирования, в автоматическом анализе и переводе текстов в системах искусственного интеллекта и т.д.

Знаковые системы, обладающие семантикой, называются *языками*. Языки лежат в основе реализации любого информационного процесса. Языки предназначены для восприятия и интерпретации информации и реализации свойств относительной независимости информации от её носителя. Так как различные носители информации приспособлены под различные знаковые системы, то в общем случае возникает необходимость перевода информации из одной знаковой системы в

другую (с одного языка в другой). Например, ноты языка звуков переносятся на бумагу и фиксируются на ней в виде специальных символов, нотной грамоты.

Для реализации информационных процессов особенно важны искусственные знаковые системы, которые созданы на формальной основе (например, алгоритмические языки программирования). Основные отличия искусственных языков от естественных языков:

1. Отсутствие многозначности, каждая лексическая единица имеет ровно один смысл и наоборот.
2. Практическое отсутствие перефразирования, то есть изменения формы высказывания при полном сохранении смысла.
3. Ориентир, в основном, на письменное представление.
4. Отсутствие избыточности, что является причиной отсутствия многофункциональности.
5. Выявление связи между лексической единицей языка и смыслом, который она представляет (оператор сравнения, оператор действия).

Информация, выраженная посредством знаковых систем, называется *данными*. Данные отображают в себе информацию, позволяя работать с ней независимо от источника информации. Для обеспечения адекватности информации, выраженной в данных, своему оригиналу данные должны периодически обновляться.

1.3. Интеллектуальная деятельность с позиции информатики

Интеллектуальная деятельность не может осуществляться без сбора, накопления и осмысления информации. Чем более эффективными информационными средствами владеет специалист, тем более успешно складывается его деятельность.

Изучение истории общества показывает, что проблема сбора, накопления, обработки и использования различных сведений и данных стояла перед человеком на всех этапах его развития. В течение длительного времени основными

механизмами ее решения были мозг, язык и уши человека (т.е. умственная деятельность человека, речь, слух). В истории развития цивилизации произошло несколько информационных революций - т.е. преобразований общественных отношений из-за кардинальных изменений в сфере обработки информации:

1. *Первая революция* связана с изобретением письменности, что привело к гигантским качественному и количественному скачкам. Появилась возможность передачи знаний между поколениями.

2. *Вторая революция* (середина XVI в.) вызвана изобретением книгопечатания, которое радикально изменило индивидуальное общество, культуру, организацию деятельности.

3. *Третья* (конец XIX в.) обусловлена изобретением электричества, благодаря которому появились телеграф, телефон, радио, позволяющие оперативно передавать информацию в любом объеме.

4. *Четвертая* (начало 50-х гг. XX в.) связана с появлением ЭВМ и другой информационной техники. Появление ЭВМ открыло эру механизации и автоматизации умственного труда, позволило хранить и обрабатывать огромные массы информации.

5. *Пятая* (70-е гг. XX в.) связана с изобретением микропроцессорной техники и появлением ПК. На микропроцессорах и интегральных схемах создаются компьютеры, компьютерные сети, системы передачи данных (информационные коммуникации). Изобретение ПК вызвало переход к широкому применению ЭВМ в информационных процессах в масштабах всего общества.

Процесс компьютеризации современного общества приводит к появлению так называемого информационного общества, в котором людям обеспечивается доступ к надежным источникам информации, обеспечивается высокий уровень автоматизации обработки информации в производственной и социальной сферах. Движущей силой развития общества становится производство информационного продукта наравне с материальным. По сравнению с индустриальным обществом, где всё направлено на производство и потребление материальных благ, в

информационном обществе все большее значение приобретают информационные продукты, заменяющие собой материальные. Например, появились такие услуги, как электронная почта, компьютерные игры, компьютерные обучающие программы, виртуальное моделирование реального мира и многое другое.

2. Элементы моделирования и кодирования информации

Одна и та же материальная система, рассматриваемая с различных точек зрения, является источником различной информации. Например, химика морская вода интересует с точки зрения её химического состава, а отдыхающего интересует температура воды, то есть каждый потребитель информации воспринимает и интерпретирует информацию от одного и того же источника по-разному. Каждый потребитель в общем случае определяет свою информационную модель источника информации, используя для этого пригодные знаковые системы.

2.1. Информационная модель источника информации. Основные определения и понятия

С позиции моделирования информации источник называют *предметной областью* (ПрО), а используемую знаковую систему (способ абстрагирования предметной области) - *моделью данных*. Описание предметной области в терминах выбранной модели данных называют *концептуальной схемой* ПрО. Для описания ПрО используют три основных конструктивных элемента - сущность, атрибут и связь.

Сущность - это обобщённое понятие для обозначения множества однородных объектов ПрО, информацию о которых необходимо собирать и хранить в информационной системе. Сущность определяется своим уникальным именем и перечнем атрибутов, характеризующих свойства сущности. Сущность можно интерпретировать как поименованный "ящик", содержащий в себе множество однородных объектов. Конкретный объект множества называют *экземпляром* сущности. Каждый экземпляр имеет свой уникальный набор значений атрибутов сущности.

Атрибут - это поименованная характеристика сущности, которая принимает значения из некоторого множества допустимых значений. Атрибуты моделируют свойства сущности. Например, при описании ПрО, связанной с учебным процессом, можно выделить сущность **студент**, а для описания свойств сущности **студент** могут быть использованы атрибуты: **НОМЕР_ЗАЧЕТНОЙ_КНИЖКИ**; **ФАМИЛИЯ_ИНИЦИАЛЫ**; **НОМЕР_ГРУППЫ**. Чтобы задать атрибут в модели, необходимо присвоить ему наименование, определить множество его допустимых значений и указать, для чего он используется. Имя атрибута связывают с его смысловым описанием. В этой связи совокупность имен атрибутов описывает свойства сущности. В приведенном выше примере сущность **студент** обладает свойством иметь уникальный шифр (**НОМЕР_ЗАЧЕТНОЙ_КНИЖКИ**), фамилию и инициалы, а также входить в состав некоторой учебной группы. Эти свойства закреплены за именами соответствующих атрибутов.

В отличие от имён значения атрибутов предназначены для идентификации экземпляров сущности, или указания количественных значений соответствующих свойств экземпляров сущности, или для выражения отношений между различными сущностями. Например, атрибут **НОМЕР_ЗАЧЕТНОЙ_КНИЖКИ** позволяет выделить (идентифицировать) конкретного студента из множества экземпляров сущности **студент**. Если добавить к сущности **студент** атрибут **ВОЗРАСТ**, то конкретное значение этого атрибута будет давать количественное значение свойства "иметь возраст", но идентифицировать по этому атрибуту конкретного студента в общем случае нельзя. Если ввести в рассмотрение сущность **группа**, одним из атрибутов которой был бы атрибут **НОМЕР_ГРУППЫ**, то значения атрибута **НОМЕР_ГРУППЫ** сущности **студент** задают связь между экземплярами сущностей **студент** и **группа**.

Связь - это обобщённое понятие, предназначенное для обозначения выделенного в ПрО отношения между двумя или более сущностями. Как и сущности, каждая категория пользователей выделяет связи в соответствии со своей концепцией ПрО. Например, в ПрО *учебный процесс* между сущностями **студент**, **преподаватель** и **дисциплина** существует связь **экзамен**, которую можно интерпретировать как абстрактную ссылку, указывающую одновременно на один

экземпляр сущности **студент**, один экземпляр сущности **преподаватель** и один экземпляр сущности **дисциплина**, которые вместе характеризуют обобщённое понятие экзамен. Перечень всех троек *<студент, преподаватель, дисциплина>*, составленных из экземпляров соответствующих сущностей, определяет набор экземпляров связи **экзамен**.

Связь, как и сущность, может иметь атрибуты. Связь **экзамен** характеризуется, например, атрибутами **ОЦЕНКА** и **ДАТА_ПРОВЕДЕНИЯ**.

Распространённой моделью данных является устная или письменная речь. Концептуальная схема ПрО в этом случае выражается в виде совокупности текстов и/или фонограмм. Однако тексты и фонограммы плохо приспособлены, для моделирования информации, имеющей сложную структуру. В этом случае используют модели данных, располагающие удобными средствами описания информационной структуры. Существуют различные модели данных для описания информационных структур. Каждая из них имеет свои достоинства и недостатки. Чтобы получить представление о подобных моделях, рассмотрим табличную форму представления данных, которая часто используется на практике для построения концептуальной схемы ПрО при разработке систем, предназначенных для хранения и обработки информации, получаемой из этой предметной области.

Табличная модель предлагает представлять концептуальную схему, интерпретируя как сущности, так и связи в виде таблиц. С таблицами люди привыкли работать давно, поэтому табличная модель данных получила широкое распространение при информационном моделировании ПрО.

Одним из положительных свойств табличной модели является её однородность. Все данные рассматриваются как хранимые в таблицах, в которых каждая строка имеет один и тот же формат. Сущность представляется совокупностью атрибутов, которые характеризуют свойства сущности, а связь - совокупностью атрибутов, являющихся идентифицирующими атрибутами (ключевыми атрибутами) в сущностях, между которыми она устанавливается. Кроме того, связь может представляться атрибутами, которые характеризуют

свойства самой связи. Каждая строка в таблице представляет собой либо некоторый экземпляр сущности, либо некоторый экземпляр связи между соответствующими экземплярами связываемых сущностей.

Например, сущность **студент** в табличной модели может иметь вид:

НОМЕР_ЗАЧЕТНОЙ_КНИЖКИ	ФАМИЛИЯ, ИНИЦИАЛЫ	ВОЗРАСТ
...	...	
...	...	

Связь **экзамен** в табличной форме будет выглядеть как

ОЦЕНКА	ДАТА_ПРОВЕДЕНИЯ
...	...
...	...

Примером атрибута связи, характеризующим свойства самой связи, является атрибут **ОЦЕНКА** в рассмотренной выше связи **экзамен**.

В табличной модели строки таблиц содержат экземпляры сущностей или связей. Каждый экземпляр характеризуется конкретными значениями атрибутов. Заметим, что табличная форма моделирования **ПрО** порождает информационную модель, которая достаточно ясно представляется в виде данных. Для этого необходимо только задать способ знаковой интерпретации значений атрибутов. Решением подобных проблем занимается раздел информатики, который называется «кодирование информации».

2.2. Кодирование информации

Построенная информационная модель предметной области (источника информации) задает структурный вид информации. Информационные элементы этой структуры, которым должны быть поставлены в соответствие данные, непосредственно на источнике информации (**ПрО**) могут быть представлены в различном виде. Например, один элемент характеризуется яркостью свечения, другой - цветом, третий - температурой, четвертый - текстом и т. п. Носитель же

информации, который используется для восприятия информации в виде данных, имеет ограниченные возможности по фиксации на себе данных в том или ином виде. Например, бумага не может звучать, как скрипка, но она может фиксировать звуки в виде нот. Следовательно, информационное сообщение, в общем случае, должно быть преобразовано в форму, удобную для целей хранения и обработки данных на выбранном носителе. Процедура подобного преобразования информации называется *кодированием* (по существу, под кодированием понимается однозначное отображение символов одной системы символами другой знаковой системы).

Проблему кодирования информации поясним на примере рассмотренной выше табличной модели, используемой для информационного моделирования источника информации. Так как и сущности, и связи в табличной модели представляются своими именами и наборами атрибутов, то именно они и должны быть представлены в виде данных. Чтобы сопоставить, например, некоторому значению атрибута данное, требуется выбрать удобную знаковую систему и на её основе сформировать соответствующий знак, который и будет играть роль данного. В рамках проблемы кодирования информации подобные знаковые системы называют *системами кодирования*, а получаемые в качестве данных знаки называют *кодами*. Образованная после кодирования комбинация называется *кодовым словом*.

Для выполнения кодирования используется *система кодирования* - совокупность правил кодового обозначения информационных объектов. Код строится на базе алфавита, состоящего из букв, цифр и других символов. Код характеризуется длиной - числом позиций в коде, и структурой - порядком расположения в коде символов. Характерным примером систем кодирования информации являются системы счисления.

2.2.1. Практически значимые системы кодирования

В зависимости от назначения и применения различают

- внутренние коды (для представления данных в ЭВМ);

- коды, предназначенные для обмена данными и их передачи по каналам связи;
- коды для специальных применений.

Внутренние коды базируются на использовании позиционных систем счисления с заданным основанием P (см. параграф 2.1.2). В общем случае любое число X_p может быть представлено в виде:

$$X_p = A_N * P^N + A_{N-1} * P^{N-1} + \dots + A_1 * P^1 + A_0 * P^0, \text{ где} \quad (2.1)$$

N – количество разрядов, с помощью которых записывается число;

A_0, A_1, \dots, A_N – разрядные коэффициенты, которые могут принимать значения цифр, соответствующей системы счисления (изменяются в диапазоне $[0, P-1]$);

i – номер разряда числа.

Одни и те же данные могут быть представлены в различных системах счисления. В ЭВМ используются десятичная, двоичная, восьмеричная и шестнадцатеричная системы счисления, однако основной является двоичная система счисления, остальные используются либо для удобства представления информации (более привычная система счисления - десятичная), либо для более компактного представления информации в ЭВМ (восьмеричная и шестнадцатеричная).

Система счисления	Основание	Символы для кодирования A_i
Двоичная	2	0, 1
Восьмеричная	8	0,1,2,3,4,5,6,7
Десятичная	10	0,1,2,3,4,5,6,7,8,9
Шестнадцатеричная	16	0,1,2,3,4,5,6,7,8,9, A, B, C, D, E, F

10 11 12 13 14 15

К кодам, предназначенным для обмена данными, относится код ASCII (*American Standard Code for Information Interchange* - американский стандартный код для обмена информацией), предназначенный для хранения символьных данных в ЭВМ.

Основной стандарт (семиразрядный код) используется для кодирования управляющих символов, цифр и букв латинского алфавита. В расширенном варианте (восьмиразрядный код) - символы псевдографики и буквы национального алфавита.

Число в 10-ой системе счисления	2 система счисления (число)		ASCII-код (символ)		Символ
	0000	0000	0011	0000	
0	0000	0000	0011	0000	'0'
1	0000	0001	0011	0001	'1'
2	0000	0010	0011	0010	'2'
3	0000	0011	0011	0011	'3'
4	0000	0100	0011	0100	'4'
5	0000	0101	0011	0101	'5'
6	0000	0110	0011	0110	'6'
7	0000	0111	0011	0111	'7'
8	0000	1000	0011	1000	'8'
9	0000	1001	0011	1001	'9'

Символы латинского алфавита	Десятичный код символа	ASCII-код символа	
'A'	65	0100	0001
'B'	66	0100	0010
'C'	67	0100	0011
'D'	68	0100	0110
•••			
'a'	97	0110	0001
'b'	98	0110	0010
'c'	99	0110	0011
'd'	100	0110	0100
•••			

Коды для специальных применений разработаны для решения задач передачи и обработки данных в автоматических телефонных станциях, программном управлении металлорежущих станков, полиграфических наборных машинах и т.д.

Например, код для коммутации каналов связи «1 из n», n- разрядность двоичного числа, которое преобразуется в соответствующее значение позиционного десятичного кода или обратно. Для n = 10:

9	8	7	6	5	4	3	2	1	0
0	0	0	0	0	0	0	0	0	1
								1	
							1		
						1			
				1					
			1						
		1							
	1								
И	т.	д.							

Код Грея используется в аналогово-цифровых преобразователях (в нем два соседних числа отличаются на 1).

10 с/с	2 с/с (4 разрядный код)				Код Грея			
					(4 разрядный)			
0	0	0	0	0	0	0	0	0
1	0	0	0	1	0	0	0	1
2	0	0	1	0	0	0	1	1
3	0	0	1	1	0	0	1	0
4	0	1	0	0	0	1	1	0
5	0	1	0	1	0	1	1	1
6	0	1	1	0	0	1	0	1
7	0	1	1	1	0	1	0	0
8	1	0	0	0	1	1	0	0
9	1	0	0	1	1	1	0	1

2.2.2. Системы счисления

Исторически системы счисления возникали как системы кодирования информации о количестве чего-либо. Всем известны *арабская десятичная* и *римская* системы счисления. В римской системе счисления каждому количеству ставится в соответствие число (код), представляющее собой специфическое сочетание символов (XIV, CXXVII и т.п.). Недостатком римской системы счисления является то, что она не даёт общего правила получения уникального кода для любого количества. Арабская десятичная система счисления лишена этого недостатка. Это связано с тем, что арабская система счисления является позиционной системой счисления, то есть каждый знак в зависимости от того, на каком месте (позиции) он стоит, даёт различные количественные значения в общем результате. В *позиционной системе счисления* определяется исходный (базовый) набор знаков, которые называют цифрами. Количество (P) различных цифр в наборе называется *основанием системы счисления*. Цифры – это уникальные знаки, которые обозначают количественные значения в пределах от 0 до $P-1$. Для системы счисления с основанием 10 (десятичной системы счисления) соответствие между количественными значениями и цифрами следующее:

пусто	0
•	- 1
• •	- 2
• • •	- 3
• • • •	- 4
• • • • •	- 5
• • • • • •	- 6
• • • • • • •	- 7
• • • • • • • •	- 8
• • • • • • • • •	- 9

Для количества большего $P-1$ строится составной знак (число). Число представляет собой последовательность значащих и незначащих разрядов (позиций). Разряды располагаются справа налево. Самый правый разряд называется младшим разрядом. В каждом разряде располагается цифра. Число как минимум состоит из одного разряда. Чтобы сопоставить заданному количеству соответствующее число, его необходимо сосчитать. Счёт выполняется следующим образом:

1. К младшему разряду прибавляем по единице, взятой из заданного количества, при этом цифра в разряде заменяется на цифру со значением на 1 большим.
2. Когда количество в разряде превысит $P-1$, то в младший разряд помещается 0, при этом к соседнему разряду прибавляется 1
3. Действуя указанным способом до тех пор, пока не пересчитаем все заданное количество. При этом подсчет переносимых в соседние разряды единиц следует производить по тому же правилу, что и в младшем разряде.

Для дробных значений кодирование выполняется по такому же принципу, только подсчитываются доли, на которые делится целое.

Рассмотрим примеры преобразования информации из одной системы счисления в другие. Для этого рассмотрим запись числа в десятичной системе счисления. Например, четырехзначное число $Z_{10}=1999$ в десятичной системе счисления может быть представлено в виде:

$Z_{10}=1*10^3+9*10^2+9*10^1+9*10^0$, т. е. в виде суммы заданного количества единиц, десятков, сотен тысяч и т.д. (алгоритм перевода построен на двух операциях – сложении и умножении).

По образу и подобию проводится перевод из любой заданной системы счисления (как с меньшим, так и с большим основанием) в десятичную (в соответствии с формулой (2.1)):

$$1110011_2=1*2^6+1*2^5+1*2^4+0*2^3+0*2^2+1*2^1+1*2^0=64+32+16+2+1=115_{10}$$

$$1110011_4=1*4^6+1*4^5+1*4^4+0*4^3+0*4^2+1*4^1+1*4^0=4096+1024+256+4+1=5381_{10}$$

$$1110011_8=1*8^6+1*8^5+1*8^4+0*8^3+0*8^2+1*8^1+1*8^0=262144+32768+4096+8+1=299017_{10}$$

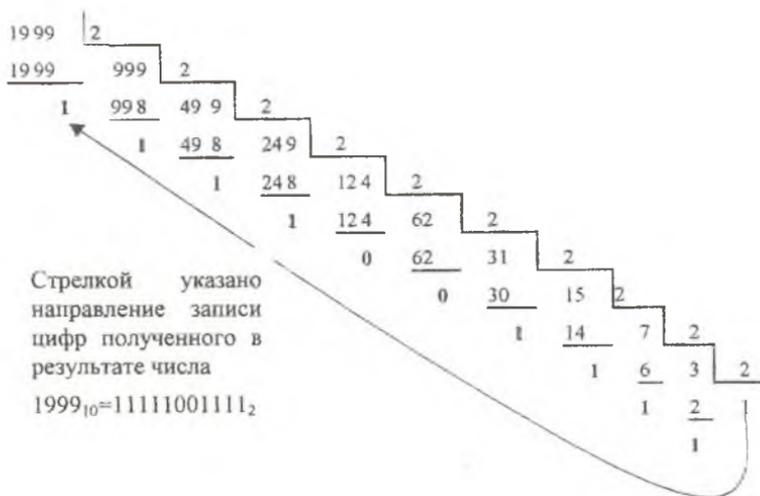
$$264_8=2*8^2+6*8^1+4*8^0=128+48+4=180_{10}$$

$$1110011_{16}=1*16^6+1*16^5+1*16^4+0*16^3+0*16^2+1*16^1+1*16^0=...$$

$$A1F_{16}=10*16^2+1*16^1+15*16^0=256+16+15=287_{10}$$

Перевод целых чисел из десятичной системы счисления в другую, наоборот, построен на операциях деления и вычитания и осуществляется последовательным делением десятичного числа на основание новой системы (например, на 2, если перевод осуществляется в двоичную систему счисления). После первого деления в остатке окажется количество единиц первого (младшего) разряда, от деления полученного частного на основание новой системы счисления в остатке окажется количество единиц следующего разряда и так далее. Деление необходимо продолжать до тех пор, пока в остатке не получится число, по значению меньшее основания системы счисления, в которую осуществляется перевод.

Рассмотрим алгоритм перевода на примере:



Как видно из примера, такой перевод достаточно длительный особенно для больших чисел. Поэтому возможен другой подход к решению этой задачи, основанный на разложении числа по степеням основания системы счисления:

1. Ищется ближайшее целое, меньшее исходного числа, которое является степенью основания системы счисления;

2. Данное число записывается в виде слагаемого (которое определяет присутствие разряда с заданным весом в числе), далее находится разность между исходным и найденным числами и процесс 1-2 итерационно повторяется до тех пор, пока последнее слагаемое в сумме не окажется равным степени заданного основания системы счисления.

$$\begin{aligned}
 1999_{10} &= 1024 + 975 = 1024 + 512 + 463 = 1024 + 512 + 256 + 207 = 1024 + 512 + 256 + 128 + 64 + 8 + 4 + 2 + 1 \\
 &= 1 * 1024 + 1 * 512 + 1 * 256 + 1 * 128 + 1 * 64 + 0 * 32 + 0 * 16 + 1 * 8 + 1 * 4 + 1 * 2 + 1 * 1 = \\
 & \quad 2^{10} \quad 2^9 \quad 2^8 \quad 2^7 \quad 2^6 \quad 2^5 \quad 2^4 \quad 2^3 \quad 2^2 \quad 2^1 \quad 2^0 \\
 & = 11111001111_2
 \end{aligned}$$

Таким образом, ненулевые слагаемые дают значимые разряды в записи результирующего числа.

Как было сказано ранее, назначение восьмеричной и шестнадцатеричной систем счисления – компактное представление чисел, записанных в двоичной систем счисления. Здесь перевод чисел осуществляется особым образом: путем объединения двоичных разрядов по *триадам* (для восьмеричной с/с) или *тетрадам* (для шестнадцатеричной с/с) в соответствии с таблицей кодирования цифр от 0 до 7 (или 0 до 15). Триады и тетрады выделяются справа налево, недостающие (старшие) разряды заполняются нулями.

Перевод из двоичной с/с в старшие с/с и наоборот

$$\underbrace{1\ 1\ 1\ 0\ 0\ 0\ 1\ 1}_{=E3_{16}} = 343_8$$

$$\underbrace{0\ 0\ 0\ 1\ 1\ 1\ 1\ 0\ 0\ 1\ 1\ 0}_{=1E5_{16}} = 746_8$$

$$AF_{16} = 1010\ 1111\ 0001_2$$

$$325_8 = 011\ 010\ 101_2$$

$$10151_8 = 001\ 000\ 001\ 101\ 001_2$$

$$325_{16} = 0011\ 0010\ 0101_2$$

2.2.3. Правила вычислений в двоичной системе счисления.

Для выполнения операций над двоичными числами используются как арифметические, так и логические операторы.

Арифметические операторы представлены в таблице.

Сложение	Вычитание	Умножение
0+0=0	0 - 0=0	0*0=0
0+1=1	1 - 0=1	0*1=0
1+0=1	1 - 1=0	1*0=0
1+1=10	10 - 1=1	1*1=1

Приоритеты (порядок выполнения) операторов точно такие же, как и при выполнении действий в десятичной арифметике.

Сложение

$$\begin{array}{r}
 + 1\ 0\ 1\ 1\ 0\ 0\ 1 \\
 \underline{\quad 1\ 1\ 1\ 1\ 0\ 1} \\
 1\ 0\ 0\ 1\ 0\ 1\ 1\ 0
 \end{array}$$

Умножение

$$\begin{array}{r}
 \quad \quad \quad 1\ 1\ 1\ 0\ 1 \\
 \quad \quad \quad \times \quad \quad 1\ 0\ 1 \\
 \hline
 \quad \quad 1\ 1\ 1\ 0\ 1 \\
 + \quad 0\ 0\ 0\ 0\ 0 \\
 \hline
 + 1\ 1\ 1\ 0\ 1 \\
 \hline
 1\ 0\ 0\ 1\ 0\ 0\ 0\ 1
 \end{array}$$

Вычитание

$$\begin{array}{r}
 - 1\ 0\ 1\ 1\ 0\ 0\ 1 \\
 \underline{\quad 1\ 1\ 1\ 1\ 0\ 1} \\
 \quad 1\ 1\ 1\ 0\ 0
 \end{array}$$

Деление

$$\begin{array}{r}
 1\ 0\ 1\ 1\ 0\ 1 \quad | \quad 1\ 0\ 0\ 1 \\
 \underline{1\ 0\ 0\ 1} \quad \quad \quad 1\ 0\ 1 \\
 \quad 1\ 0\ 0\ 1 \\
 \quad \quad \underline{1\ 0\ 0\ 1} \\
 \quad \quad \quad 0
 \end{array}$$

Логические операторы представлены в таблице

Отрицание НЕ (<i>NOT</i>)	Логическое сложение ИЛИ (<i>OR</i>)			Логическое умножение И (<i>AND</i>)		
	X1	X2	X1 Or X2	X1	X2	X1 And X2
Not 0=1	X1	X2	X1 Or X2	X1	X2	X1 And X2
Not 1=0	0	0	0	0	0	0
	0	1	1	0	1	0
	1	0	1	1	0	0
	1	1	1	1	1	1

Переменные, объединенные знаками логических операций, составляют *логические выражения*. При вычислении логических выражений определены следующие приоритеты выполнения логических операций:

- Отрицание – первый (высший) приоритет
- логическое умножение – второй приоритет
- Логическое сложение – третий (низший) приоритет.

Порядок выполнения операций может быть изменен введением скобок.

2.3. Элементы алгебры логики

В основу выполнения логических операторов положена *алгебра логики* или булева алгебра (Дж. Буль – английский математик 19 века). Булева алгебра оперирует с логическими переменными, которые могут принимать только два значения: *истина* и *ложь*, обозначаемые соответственно 1 и 0. Таким образом, одни и те же устройства ЭВМ могут применяться для обработки и хранения как числовой информации в двоичной системе счисления, так и логических переменных. Совокупность значений логических переменных x_1, x_2, \dots, x_n называется *набором переменных*. Набор логических переменных удобно изображать в виде n -разрядного двоичного числа, каждый разряд которого равен значению одной из переменных. Количество наборов логических переменных в n двоичных разрядах равно 2^n .

Логической функцией от набора логических переменных $f(x_1, x_2, \dots, x_n)$ называется функция, которая может также принимать только два значения: истина или ложь. Логические функции выражаются через элементарные логические функции, которые легко задаются таблично. Как правило, это функции от одной или двух переменных. Совокупность логических функций, с помощью которых можно выразить логическую функцию любой сложности, называется функционально полными системами логических функций.

Наиболее часто используемая система логических функций: инверсия (\bar{x} , отрицание), конъюнкция (\wedge , логическое умножение), дизъюнкция (\vee , логическое сложение)

x	\bar{x}
0	1
1	0

x1	x2	$x1 \wedge x2$
0	0	0
0	1	0
1	0	0
1	1	1

x1	x2	$x1 \vee x2$
0	0	0
0	1	1
1	0	1
1	1	1

Для упрощения логических функций в алгебре логики используются следующие основные законы, позволяющие производить тождественные преобразования логических выражений:

1. Коммутативный закон (переместительный):

$$x_1 \vee x_2 = x_2 \vee x_1 \qquad x_1 \wedge x_2 = x_2 \wedge x_1$$

2. Ассоциативный закон (сочетательный):

$$x_1 \vee (x_2 \vee x_3) = (x_1 \vee x_2) \vee x_3 \qquad x_1 \wedge (x_2 \wedge x_3) = (x_1 \wedge x_2) \wedge x_3$$

3. Дистрибутивный закон (распределительный):

$$x_1 \wedge (x_2 \vee x_3) = (x_1 \wedge x_2) \vee (x_1 \wedge x_3) \qquad x_1 \vee (x_2 \wedge x_3) = (x_1 \vee x_2) \wedge (x_1 \vee x_3)$$

4. Правила де Моргана (отрицание отрицания):

$$\overline{x_1 \vee x_2} = \overline{x_1} \wedge \overline{x_2} \qquad \overline{x_1 \wedge x_2} = \overline{x_1} \vee \overline{x_2}$$

5. Правила операций с константами:

$$\overline{0} = 1; \quad \overline{1} = 0; \quad x \wedge 1 = x; \quad x \wedge 0 = 0; \quad x \vee 0 = x; \quad x \vee 1 = 1.$$

6. Правила операций переменной с ее инверсией:

$$x \vee \overline{x} = 1; \qquad x \wedge \overline{x} = 0;$$

7. Закон поглощения:

$$x_1 \vee (x_1 \wedge x_2) = x_1; \qquad x_1 \wedge (x_1 \vee x_2) = x_1.$$

8. Закон идемпотентности:

$$x \vee x = x; \qquad x \wedge x = x;$$

9. Закон двойного отрицания:

$$\overline{\overline{x}} = x.$$

Схемные реализации операций, выполняемых элементарными логическими функциями, называются логическими элементами. С их помощью реализуются функции управления процессом обработки информации, о которых будет рассказано ниже.

3. Методы работы с процедурной информацией

До сих пор понятие информации рассматривалось как совокупность фактических данных. Однако информацию можно рассматривать и как динамическую величину, то есть *информацию действия*. В аспекте информация необходима для решения некоторой задачи. Различают технологические и функциональные задачи.

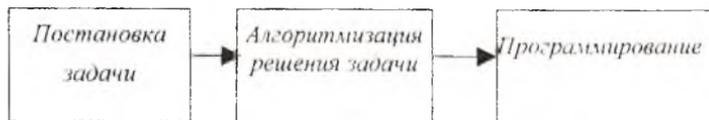
Технологические задачи ставятся и решаются при организации технологического процесса обработки информации на компьютере. Технологические задачи являются основой для сервисных систем программного обеспечения в виде операционных систем (ОС), систем программирования, систем управления базами данных (БД), табличных процессоров, текстовых процессоров и так далее, то есть при разработке системного программного обеспечения.

Функциональные задачи ставятся при реализации функций управления в рамках информационных систем предметных областей. Например, управление деятельностью предприятия или фирмы, управление поставками и перевозками продукции и т.д. Функциональные задачи в совокупности образуют предметную область и полностью определяют ее специфику. Функциональные задачи решаются при разработке прикладного программного обеспечения.

Предметная (прикладная) область – совокупность связанных между собой функций, задач управления, с помощью которых достигается выполнение поставленных целей.

3.1. Основные этапы решения прикладных задач

Процесс решения задач на компьютере можно представить как последовательность следующих действий:



Постановка задачи – это точная формулировка решения задачи на компьютере с описанием входной и выходной информации.

Постановка задачи связана с конкретизацией основных параметров ее реализации, определением источников и структурой входной и выходной информации, востребованной пользователем.

К *основным характеристикам* функциональных задач, уточняемым в процессе ее формализованной постановки, относятся:

- цель и (или) назначение задачи, ее место и связи с другими задачами;
- условия решения задачи с использованием средств вычислительной техники;
- содержание функций обработки входной информации при решении задач;
- требования к периодичности решения задачи;
- ограничения по точности выходной информации;
- состав и форма представления выходной информации;
- источники входной информации для решения задачи;
- пользователи задачи (кто ее решает и пользуется результатами).

Входная информация по задаче определяется как данные, поступающие на вход задачи и используемые для ее решения. Входной информацией служат рукописные документы, файлы базы данных, нормативно-справочная информация, входные сигналы от датчиков устройств и т.д.

Выходная информация по задаче может быть представлена в виде документов, выданных на принтер, файла базы данных, графика на экране дисплея, управляющего сигнала и т.д.

Алгоритм – система точно сформулированных правил, однозначно определяющая процесс преобразования допустимых исходных данных (входной информации) в желаемый результат (выходную информацию) за конечное число шагов.

Алгоритм должен обладать следующими свойствами:

- *дискретностью* – разбиением процесса обработки информации на более простые этапы (шаги), выполнение которых компьютером или человеком не вызывает затруднений;
- *определенностью* (детерминированностью) – однозначностью получаемого результата при одних и тех же исходных данных;
- *результативностью* – обязательным получением желаемого результата за конечное число шагов при допустимых исходных данных;
- *массовостью* – применимостью алгоритма для решения определенного класса задач.

3.2. Основные формы представления алгоритмов

В алгоритме отражаются логика и способ формирования результатов решения с указанием необходимых расчетных формул, логических условий, соотношений для контроля достоверности выходных результатов. В алгоритме обязательно должны быть предусмотрены все ситуации (в том числе и «критические»), которые могут возникнуть в процессе решения задачи, особенно при некорректно заданных входных данных (задача контроля ввода данных). В зависимости от степени детализации, поставленных целей, методов и технических средств решения задачи используются различные способы представления алгоритмов. На практике наиболее распространены следующие формы представления алгоритмов:

- содержательная (текстуальная) форма;
- графическая форма (блок-схема);
- программная (на языках программирования компьютера).

Рассмотрим перечисленные формы представления алгоритмов на примере решения следующей задачи. Пусть заданы три числа A_1 , A_2 , A_3 . Требуется определить максимальное по абсолютной величине число A_{max} из заданного набора.

Содержательная форма представления алгоритма:

Шаг 1. Определить абсолютные значения чисел A_1, A_2, A_3 .

Шаг 2. Сравнить абсолютные значения чисел A_1 и A_2 . Если $|A_1| > |A_2|$, то числу A_{max} присвоить значение $|A_1|$, иначе числу A_{max} присвоить значение $|A_2|$.

Шаг 3. Сравнить абсолютные значения чисел A_3 и A_{max} . Если $|A_3| > |A_{max}|$, то числу A_{max} присвоить значение $|A_3|$.

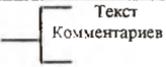
Шаг 4. Конец алгоритма.

Приведенное описание алгоритма является достаточно строгим и позволяет однозначно решить поставленную задачу. Однако для достаточно сложных алгоритмов такое описание становится слишком громоздким и ненаглядным. Поэтому содержательную форму используют на начальных стадиях разработки алгоритмов, когда определяются основные этапы решения поставленной задачи.

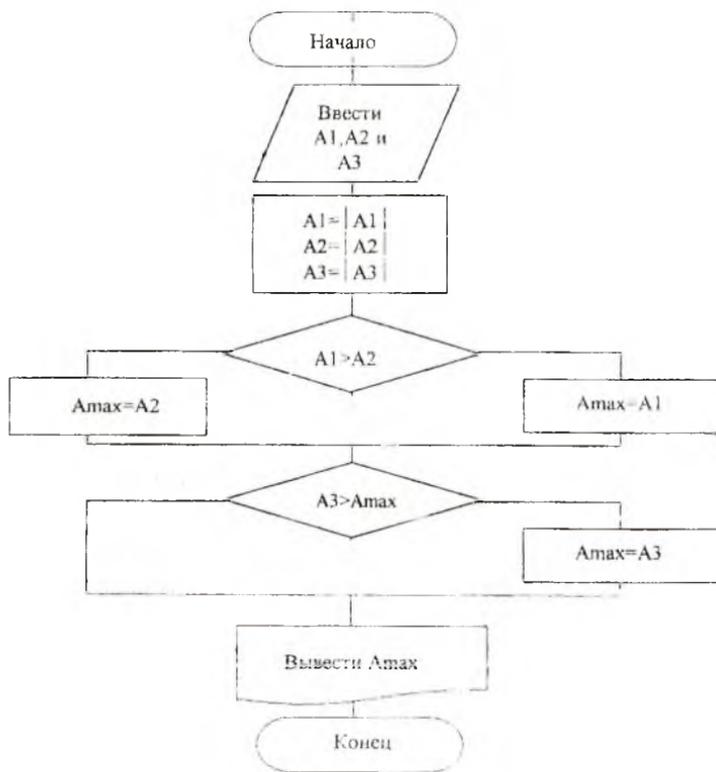
Графическая форма представления алгоритмов является более компактной и наглядной по сравнению с текстуальной формой. Алгоритм изображается в виде последовательности связанных между собой функциональных блоков, каждый из которых соответствует выполнению одного или нескольких действий. Такое графическое представление называется блок-схемой алгоритма.

Условные обозначения функциональных блоков, принятые в схемах алгоритмов

Графическое обозначение	Наименование	Пояснения
	Процесс	Операция, в результате которой изменяется значение данных.
	Решение	Разветвление алгоритма в зависимости от некоторых условий.
	Модификация	Начало циклической операции.
	Ввод-вывод данных	Ввод-вывод без указаний конкретного носителя.
	Документ	Ввод-вывод данных, носителем которых является бумага.

	Дисплей	Ввод-вывод данных на дисплей.
	Магнитный диск	Ввод-вывод данных на магнитный диск.
	Пуск-останов	Начало, конец, прерывание процесса обработки данных.
	Комментарий	Связь между элементом схемы и пояснением к нему.
	Соединители	Связь между прерванными линиями на одной странице, связь между прерванными частями схем на разных страницах

Рассмотрим блок-схему рассмотренного алгоритма



Программная форма представления алгоритма

Алгоритм и его программная реализация тесно взаимосвязаны. *Программа* – упорядоченная последовательность инструкций компьютера (команд) для решения задачи. *Программное обеспечение* – совокупность программ обработки данных и необходимых для их эксплуатации документов.

Программирование – теоретическая и практическая деятельность, связанная с созданием программ. Программирование является собирательным понятием и может рассматриваться и как наука, и как искусство; на этом основан научно-практический подход к разработке программ.

Программа – результат интеллектуального труда, для которого характерно творчество, поэтому в любой программе присутствует индивидуальность ее разработчика, программа отражает определенную степень искусства программиста. Вместе с тем программирование предполагает и рутинные работы, которые могут и должны иметь строгий регламент выполнения и соответствовать стандартам.

Программирование базируется на комплексе научных дисциплин, направленных на исследование, разработку и применение методов и средств разработки программ. При разработке программ используются ресурсоемкие и наукоемкие технологии, высококвалифицированный интеллектуальный труд. По некоторым данным в середине 90-х годов в мире было занято программированием $\approx 2\%$ трудоспособного населения. Совокупный оборот в сфере создания программных средств достигает нескольких сот миллиардов долларов в год. В связи с этим весьма актуальным становится вопрос о разработке и применении эффективных технологий программирования.

Основная категория специалистов, занятых разработкой программ, – это *программисты*. Программисты неоднородны по уровню квалификации, а также по характеру своей деятельности. Наиболее часто программисты делятся на системных и прикладных. Системный программист занимается разработкой, эксплуатацией и сопровождением системного программного обеспечения, поддерживающего работоспособность компьютера и создающего сферу для

выполнения программ, обеспечивающих реализацию функциональных задач. Прикладной программист осуществляет разработку и отладку программ для решения функциональных задач.

В процессе создания программ на начальной стадии работ участвуют и специалисты – *постановщики задач*. При создании больших по масштабам и функциям обработок программ (например, операционных систем) – нужен *программист-аналитик* для анализа и проектирования комплекса взаимосвязанных программ. Большинство информационных систем основано на работе с базами данных (БД). Если БД является сложной и используется многими программистами, то возникает проблема организационной поддержки БД, которая выполняется *администратором БД*. Основным потребителем программ служит конечный пользователь, который относится к категории пользователей-непрограммистов, которые должны иметь элементарные знания и навыки работы с компьютером.

Схема взаимодействия специалистов, связанных с созданием и эксплуатацией программ



3.3. Основные характеристики программных продуктов

Все программы по характеру использования и категориям пользователей можно разделить на два класса: утилитарные программы и программные продукты (изделия).

- *Утилитарные программы* («программы для себя») предназначены для использования самими разработчиками. Чаще всего это программные решения функциональных задач, не предназначенных для широкого распространения, или учебные программы.
- *Программные продукты* предназначены для широкого распространения и продажи.

Программный продукт должен быть соответствующим образом подготовлен к эксплуатации, иметь необходимую техническую документацию, представлять сервис и гарантию надежной работы программы, иметь товарный знак изготовителя, а также желательно наличие кода государственной регистрации.

Программный продукт – комплекс взаимосвязанных программ для решения определенной проблемы (задачи) массового спроса, подготовленный к реализации как любой вид промышленной продукции.

Путь от «программ для себя» до программных продуктов достаточно долгий, он связан с изменениями технической и программной среды разработки и эксплуатации программ, с появлением и развитием самостоятельной отрасли – информационного бизнеса, для которого характерны разделение труда фирм-разработчиков программ, их дальнейшая специализация, формирование рынка программных средств и информационных услуг.

Программный продукт разрабатывается на основе промышленной технологии выполнения проектных работ с применением современных инструментальных средств программирования. Специфика заключается в уникальности процесса разработки алгоритмов и программ, зависящего от характера обработки информации и используемых инструментальных средств.

Основные характеристики программ:

1. Алгоритмическая сложность.
2. Состав и глубина проработки реализованных функций.
3. Полнота и системность функций.
4. Объем файлов программ.
5. Требования к операционной системе и техническим средствам со стороны программы.
6. Объем дисковой памяти.
7. Размер оперативной памяти.
8. Тип процессора.
9. Версия операционной системы.
10. Наличие вычислительной сети и др.

Показатели качества программного продукта (ПП)

1. *Мобильность* – независимость ПП от технических средств обработки информации, ОС, сетевой технологии. Мобильный ПП пригоден для массового использования без каких-либо изменений.
2. *Надежность* – бесперебойность и устойчивость в работе, возможность диагностики возникающих ошибок.
3. *Эффективность* – минимально возможный расход вычислительных ресурсов и максимально возможное быстроедействие.
4. *Модифицируемость* - простота внесения изменений.
5. *Коммуникативность* – свойство интеграции с другими программами, обеспечения обмена данными в общих форматах представления.

6. Учет человеческого фактора – обеспечение дружественного интерфейса, наличие контекстно-зависимой подсказки или обучающей системы, хорошей документации.

3.3.1. Классификация программных продуктов

Выделяют три основных класса ПП:

1. Системное программное обеспечение (ПО).
2. Пакеты прикладных программ (ППП).
3. Инструментарий технологии программирования (ИТП).

Системное программное обеспечение – совокупность программ и программных комплексов для обеспечения работы компьютера и сетей ЭВМ. К ПП данного класса применяются высокие требования по надежности и технологичности работы, удобству и эффективности использования.



Данный класс ПП ориентирован, в основном, на профессиональных пользователей: системного программиста, администратора сети, прикладного программиста, оператора. Однако и конечный пользователь должен знать базовую технологию работы с этим классом ПП.

Операционная система (ОС) – совокупность программных средств, обеспечивающая управление аппаратной частью компьютера и прикладными программами, а также их взаимодействие между собой и пользователем. ОС предназначена для управления выполнением пользовательских программ, планирования и управления вычислительными ресурсами ЭВМ. Доход от продаж ОС в среднем превышает 20 млрд долларов в год (IBM, Microsoft, Novell, UNISYS).

ОС для персональных компьютеров делятся на:

- одно– и многозадачные (в зависимости от числа параллельно выполняемых прикладных процессов);
- одно– и многопользовательские (в зависимости от числа пользователей, одновременно работающих с ОС);
- непереносимые и переносимые на другие типы компьютеров;
- несетевые и сетевые, обеспечивающие работу в сетях ЭВМ.

ОС MS DOS фирмы Microsoft появилась в 1981г. В настоящее время существуют версии 6.22 и 7.0 (в составе Windows 95), а также ее разновидности других фирм-разработчиков (DR DOS, PC DOS). MS DOS – однозадачная, однопользовательская ОС. Обеспечивает командный интерфейс пользователя.

Операционные оболочки – специальные программы предназначенные для облегчения общения пользователя с командами ОС. Операционные оболочки имеют текстовый и графический варианты интерфейса пользователя. Наиболее популярны текстовые оболочки для MS DOS: Norton Commander (Symantec); DOS Navigator. Эти программы существенно упрощают задание команды ОС для выполнения.

Большую популярность получили графические оболочки Windows 95 - Windows 98. Наиболее популярен следующий комплект утилит: Norton Utilities, Symantec. Наиболее известные антивирусные программы AIDStest, DrWeb.

Пакет прикладных программ (ППП) – комплекс взаимосвязанных программ для решения задач определенного класса конкретного программного обеспечения.

В данный класс ПП входят пакеты, выполняющие обработку информации в различных предметных областях. Это самый многочисленный класс ПП.

Все ППП разделяются на :

- ППП общего назначения:
- Проблемно-ориентированные ППП (бухучет, отдел кадров, управление производством и т.д.).
- ППП автоматизированного проектирования (для поддержки работы конструкторов и технологов, связанной с разработкой чертежей, схем и т.п.).
- Методоориентированные ППП (математические, статистические и другие методы решения задач, независимо от сферы их применения).
- Офисные ППП (органайзеры, программы-переводчики, программы проверки орфографии, коммуникационные: электронная почта, браузеры для Internet и т.д.).
- Настольные издательские системы.
- Программные средства мультимедиа (процесс обучения, организация досуга).
- Системы искусственного интеллекта (экспертные системы, оболочки для их создания, системы управления базами знаний, системы анализа и распознавания речи и другие).

ППП общего назначения поддерживают информационные технологии конечных пользователей:

- настольные системы управления базами данных (СУБД);

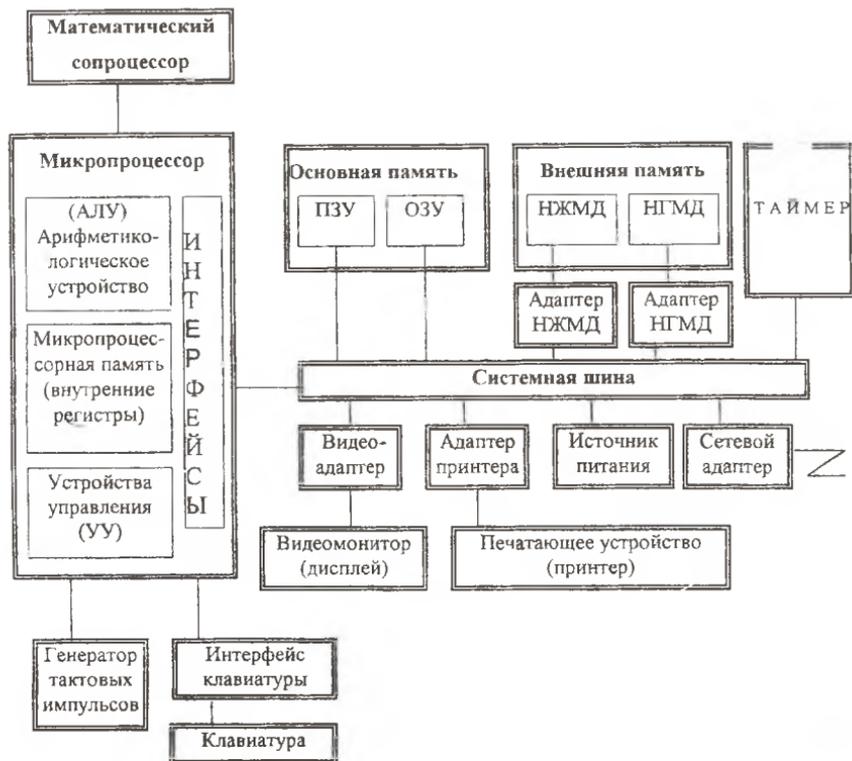
- серверы БД (для создания и работы в сети интегрированных БД);
- генераторы отчетов (серверы) – реализация запросов и генерация отчетов в условиях сети с архитектурой клиент–сервер;
- текстовые и табличные процессоры;
- средства презентационной графики – для подготовки слайд–фильмов, мультфильмов, видеофильмов, их редактирования;
- интегрированные пакеты – набор нескольких ПП, функционально дополняющие друг друга и поддерживающие единые информационные технологии.

Инструментарий технологии программирования – совокупность программ и программных комплексов, обеспечивающих технологию разработки, отладки и внедрения создаваемых ПП. Включают языки программирования, системы программирования, инструментальные среды пользователя.

4. Компьютер как универсальная информационная система

Понятие информационной системы было дано выше. Информационная система в первую очередь обеспечивает хранение и доступ к информации. В рамках информационной системы реализуется некоторая совокупность информационных процессов. Для построения любой информационной системы требуется выбрать носитель информации и знаковые системы, которые позволяли бы наиболее эффективно работать с информацией. В настоящее время в качестве такого носителя информации в широком смысле слова выступает компьютер.

4.1. Структурная схема персонального компьютера.



Рассмотрим основные составные части персонального компьютера (ПК).

Микропроцессор (МП) – центральный блок ПК, предназначенный для управления работой всех блоков машины и для преобразования информации, т.е. выполнения арифметических и логических операций над информацией. В состав МП входят следующие устройства:

арифметико-логическое устройство (АЛУ), которое предназначено для выполнения всех арифметических и логических операций над числовой и символьной информацией, для ускорения операций с вещественными числами. К АЛУ подключается математический сопроцессор. Математический сопроцессор используется для ускоренного выполнения операций над числами с плавающей

запятой, имеет свою систему команд и работает параллельно с основным МП, но под его управлением. Последние модели МП, начиная с МП 80486DX, включают сопроцессор в свой состав в качестве устройства с плавающей точкой.;

внутренние регистры, которые служат для кратковременного хранения, записи и выдачи информации, непосредственно используемой в вычислениях в ближайшее время работы ПК. Регистры – быстродействующие ячейки памяти различной длины (8+64 разрядов);

устройство управления (УУ), которое формирует и подает на все блоки машины управляющие символы, формирует адреса ячеек памяти и передает их для выполнения операции;

интерфейс МП, который реализует сопряжение и связь с другими устройствами ПК; включает в себя связь между устройствами МП, буферные регистры, схемы управления портами ввода-вывода и системной шиной. Интерфейс – совокупность средств сопряжения и связи устройств компьютера, обеспечивающая их эффективное взаимодействие.

Генератор тактовых импульсов генерирует последовательность электрических импульсов, синхронизирующих работу всех устройств ПК; частота генерируемых импульсов определяет тактовую частоту ПК, которая является одной из основных характеристик ПК.

Системная шина – основная интерфейсная система ПК. Системная шина включает в себя:

- *шину данных*, содержащую провода и схемы сопряжения для параллельной передачи всех разрядов числового кода операнда;
- *шину адреса*, содержащую провода и схемы сопряжения для параллельной передачи всех разрядов числового кода адреса ячейки основной памяти или порта ввода-вывода внешнего устройства;
- *шину управления*, содержащую провода и схемы сопряжения для передачи управляющих сигналов во все блоки ПК;

- *шину питания*, имеющую провода и схемы сопряжения для подключения блоков ПК к системе электропитания.

Системная шина обеспечивает три направления передачи информации:

- 1) между МП и ОП;
- 2) между МК и портами ввода-вывода внешних устройств;
- 3) между ОП и портами ввода-вывода внешних устройств (в режиме прямого доступа к памяти).

Все блоки ПК (точнее их порты ввода-вывода) через соответствующие унифицированные разъемы подключаются к шине единообразно: непосредственно или через адаптеры.

Управление системной шиной осуществляется либо непосредственно, либо через контроллер шины. Обмен информацией между внешними устройствами и системной шиной выполняется с использованием ASCII-кодов.

Функции хранения информации выполняет память внутренняя и внешняя.

Основная память (ОП) предназначена для хранения и оперативного обмена информацией с другими блоками ПК. Основная память состоит из постоянного запоминающего устройства (ПЗУ) и оперативного запоминающего устройства (ОЗУ).

ПЗУ служит для хранения постоянной программной и справочной информации, позволяет оперативно только считывать хранящуюся в нем информацию (изменить информацию в ПЗУ нельзя).

ОЗУ предназначено для оперативной записи, хранения и считывания информации (программ и данных), непосредственно участвующей в информационно-вычислительном процессе, выполняемом ПК в текущий период времени.

Внешняя память используется для долговременного хранения любой информации, которая может когда-либо потребоваться. Во внешней памяти хранится все программное обеспечение ПК.

Источник питания – блок, содержащий системы автономного и сетевого энергопитания ПК.

Таймер – внутримашинные электронные часы, обеспечивающие автоматический съем значения текущего момента времени (год, месяц, день, часы, минуты, секунды и доли секунды). Таймер подключается к автономному источнику питания – аккумулятору, и при отключении ПК от сети продолжает работать.

Внешние устройства обеспечивают взаимодействие ПК с окружающей средой: пользователями, объектами управления и другими ЭВМ. Они разделяются на следующие классы:

- внешние запоминающие устройства (ВЗУ);
- устройства ввода информации;
- устройства вывода информации;
- средства связи и телекоммуникации.

Устройства ввода информации:

- клавиатура;
- графические планшеты (диджитайзеры) – ручной ввод графической информации путем перемещения по планшету специального указателя (пера);
- сканеры;
- манипуляторы (джойстик – рычаг, мышь, трекбол – шар в оправе, световое перо – для ввода графической информации на экран дисплея путем управления движением курсора по экрану);
- сенсорные экраны – для ввода отдельных элементов изображения, программ или команд в ПК;
- устройство речевого ввода-вывода – быстроразвивающиеся средства мультимедиа;
- видеомонитор – устройство для отображения вводимой и выводимой информации.

Устройства вывода информации:

- принтеры – печатающие устройства для регистрации информации на бумажный носитель.

- графопостроители (плоттеры) – устройства для вывода графической информации из ПК на бумажный носитель. Векторные плоттеры вычерчивают изображение при помощи пера. Растровые бывают электростатические, струйные и лазерные. По конструкции плоттеры подразделяются на планшетные и барабанные.

Устройства связи и телекоммуникации используются для связи с другими приборами и подключения ПК к каналам связи и другим ЭВМ и вычислительным сетям (сетевые интерфейсные платы, мультиплексоры передачи данных, модемы).

Сетевой адаптер служит для подключения ПК к каналу связи для работы в составе вычислительной сети. В глобальных сетях функции сетевого адаптера выполняет модулятор-демодулятор (модем).

Средства мультимедиа (multimedia – многосредовость) – комплекс аппаратных и программных средств, позволяющих человеку общаться с ПК, используя самые разные, естественные для себя среды: звук, видео, графику, тексты, анимацию и др. К средствам мультимедиа относятся: устройства речевого ввода-вывода, сканеры, высококачественные видео- и звуковые платы, платы видеозахвата (снимают изображения с видеомаягнитофона или видеокамеры и вводят его в ПК), высококачественные акустические и видеовоспроизводящие системы с усилителями, звуковыми колонками, большими видеозэкранами. К средствам мультимедиа относят также внешние запоминающие устройства большой емкости на оптических дисках, часто используемые для записи звуковой и видеоинформации.

Дополнительные схемы:

Контроллер прямого доступа к памяти освобождает МП от прямого управления ИМД, что существенно повышает эффективность ПК.

Сопроцессор ввода-вывода за счет параллельной работы с МП значительно ускоряет выполнение процедур ввода-вывода при обслуживании нескольких внешних устройств, освобождая МП от обработки процедур ввода-вывода, в том числе реализует и режим прямого доступа к памяти.

Контроллер прерываний обслуживает процедуры прерывания, принимает запрос на прерывание от ВУ, определяет уровень приоритета этого запроса и выдает сигнал прерывания в МП. МП, получив этот сигнал, приостанавливает выполнение текущей программы и переходит к выполнению специальной программы обслуживания того прерывания, которое запросило ВУ. После завершения программы обслуживания восстанавливается выполнение прерванной программы. Контроллер прерываний является программируемым.

4.2. Элементы конструкции персонального компьютера

Конструктивно ПК выполнены в виде центрального системного блока, к которому через разъемы подключаются ВУ: дополнительная память, клавиатура, дисплей, принтер и т.д.

Системный блок обычно включает в себя системную плату (материнскую плату), блок питания, накопители на дисках, разъемы для дополнительных устройств и платы расширения с контроллерами-адаптерами ВУ.

На системной плате размещаются: микропроцессор; математический сопроцессор; генератор тактовых импульсов; блоки (микросхемы) ОЗУ и ПЗУ; адаптеры клавиатуры, НЖМД, НГМД ; контроллер прерываний; таймер.

Микропроцессор (центральный процессор или CPU)— функционально законченное программно-управляемое устройство обработки информации, выполненное в виде одной или нескольких больших (БИС) или сверхбольших (СБИС) интегральных схем.

Основные функции МП:

- чтение и дешифрация команд из ОП;
- чтение данных из ОП и регистров адаптеров ВУ;

- прием и обработка запросов и команд от адаптеров на обслуживание ВУ;
- обработка данных и их запись в ОП и регистры адаптеров ВУ;
- выработка управляющих сигналов для всех узлов и блоков ПК.

Разрядность шины данных МП определяет разрядность ПК в целом; разрядность шины адреса МП – его адресное пространство. *Адресное пространство* – это максимальное количество ячеек ОП, которое может быть непосредственно адресовано микропроцессором.

Первый МП был выпущен в США в 1971 году фирмой Intel – МП 4004. В настоящее время МП фирмы Intel и Intel-подобные являются наиболее популярными и распространенными.

Все МП можно разделить на три группы:

1. МП типа CISC (с полным набором команд);
2. МП типа RISC (с сокращенным набором команд);
3. МП типа MISC (с минимальным набором команд и весьма высоким быстродействием) – находятся в стадии разработки.

Большинство современных ПК типа IBM PC используют МП типа CISC.

Модель МП	Разрядность, бит		Тактовая частота, МГц	Адресное пространство, байт	Число команд	Число элементов	Год выпуска
	Данных	Адреса					
4004	4	4	4,77	4×10^3	45	2300	1971
8080	8	8	4,77	64×10^3		10000	1974
8086	16	16	4,77 и 8	10^6	134	70000	1982
8088	8, 16	16	4,77 и 8	10^6	134	70000	1981
80186	16	20	8 и 10	10^6		140000	1984
80286	16	24 (20)	10-33	4×10^6 (4×10^9)		180000	1985
80386	32	32	25-50	16×10^6	240	275000	1987

				(4×10^9)			
80486	32	32	33-100	16×10^6 (4×10^9)	240	$1,2 \times 10^6$	1989
Pentium	64	32	50-150	4×10^9	240	$3,1 \times 10^6$	1993
Pentium Pro	64	32	66-200	4×10^9	240	$5,5 \times 10^6$	1995

Особенности некоторых процессоров.

Начиная с МП 80286, имеется возможность многозадачной работы и работы МП в вычислительной сети. Начиная с МП 80386, используется конвейерное выполнение команд, что позволило увеличить быстродействие ПК в 2–3 раза, а также обеспечивается поддержка системы виртуальных машин, при которой в одном МП моделируется как бы несколько ПК, работающих параллельно и имеющих разные ОС. МП Pentium – это МП 80586(P5), товарная марка Pentium запатентована фирмой Intel. Эти МП имеют пятиступенчатую конвейерную структуру и КЭШ-буфер для команд условной передачи управления, позволяющие предсказать направление ветвления программы; по эффективному быстродействию они приближаются к RISC МП. Обмен данными с системой может выполняться со скоростью 1 Гбайт/с. У всех МП Pentium имеется встроенная КЭШ -память, отдельно для команд, отдельно для данных; имеются специализированные конвейерные аппаратные блоки сложения, умножения и деления, значительно ускоряющие выполнение операций с плавающей запятой. Микропроцессоры Pentium Pro – это МП 80686 (P6). Эти МП имеют 14-ступенчатую суперконвейерную структура, предсказание ветвления программы при условных передачах управления и исполнение команд по предполагаемому пути ветвления. В Pentium Pro вероятность правильного предсказания ветвления $\approx 0,9$, а в Pentium $\approx 0,8$. Внутренняя КЭШ – память емкостью 256-512 Кбайт – у Pentium Pro, а у Pentium – внутренняя КЭШ – память – 16 Кбайт, остальная (до 256-512 Кбайт) – внешняя по отношению к МП. Начиная с МП 80486, внутренняя частота процессора в несколько раз выше тактовой частоты. Для МП 80486 DX2 – в 2 раза, DX4 – в 3 раза выше.

4.3. Устройства отображения информации

Основным устройством отображения информации в ПК является видеотерминал. *Видеотерминал* состоит из видеомонитора (дисплея) и видеоадаптера. Видеоадаптеры входят в состав системного блока ПК (находятся на видеокарте, устанавливаемой в разъем материнской платы), а видеомониторы – это внешние устройства ПК.

Видеомонитор – устройство отображения текстовой и графической информации на экране (в стационарных ПК – на экране электронно-лучевой трубки (ЭЛТ), в портативных ПК – на жидкокристаллическом плоском экране). В зависимости от вида управляющего лучом сигнала мониторы бывают аналоговые и цифровые. Аналоговые мониторы позволяют более качественно, с большим количеством полутонов и цветовых оттенков формировать изображение на экране.

Размер экрана задается величиной его диагонали в дюймах: от 10” до 21”, обычно – 14”.

Важной характеристикой монитора является частота его кадровой развертки. У современных мониторов смена изображений (кадров) происходит с частотой ≈ 60 Гц (уже при 25 Гц глаз воспринимает изображение как непрерывное, но сильно устает от мерцания экрана).

Видеомониторы обычно могут работать в двух режимах: текстовом и графическом.

В *текстовом режиме* изображение на экране состоит из символов расширенного набора ASCII-кодов, формируемых знакогенератором, который находится на плате видеоадаптера. Разрешающая способность экрана в текстовом режиме обычно составляет 80 столбцов \times 25 строк. Возможен вывод текста, примитивных рисунков, рамки, таблицы.

В графическом режиме выводятся более сложные изображения и тексты с различными шрифтами и размерами букв, формируемых из отдельных мозаичных элементов – пикселей (pixel-picture element). Разрешающая

способность монитора в графическом режиме зависит от типа монитора и от характеристик видеоадаптера.

Разрешающие способности цветных графических режимов чаще всего бывают: 320×200, 640×480, 800×600, 1024×768, 1600×1200.

Важной характеристикой монитора, определяющей четкость изображения на экране, является размер зерна (точки) люминофора экрана. Чем меньше зерно, тем выше четкость изображения. У мониторов с большим зерном не может быть достигнута большая разрешающая способность. Совместно с ПК IBM PC могут использоваться различные типы мониторов, как монохромные, так и цветные.

Обычно используются цветные RGB-мониторы – они самые качественные, обладают высокой разрешающей способностью и графики и цвета (три цветовых сигнала Red-Green-Blue идут по разным проводам). RGB-мониторы работают совместно с цветным графическим видеоадаптером.

Монохромные композитные мониторы обеспечивают качественное отображение символической и графической информации, применяются, в основном, в издательских системах. Черно-белые монохромные мониторы типа «rarep white», работающие с видеоадаптером SVGA обеспечивают разрешающую способность 1280×1024 пикселей.

Среди прочих характеристик мониторов следует отметить:

- 1) наличие плоского или выпуклого экрана (1 вариант лучше);
- 2) уровень высокочастотного радиоизлучения (мониторы с низким уровнем излучения помечаются Low Radiation);
- 3) наличие защиты экрана от электростатических полей (мониторы типа Anti Static (AS));
- 4) наличие системы энергосбережения (типа Green).

Видеоадаптеры являются внутрисистемными устройствами, непосредственно управляют мониторами и выводом информации на экран.

Видеоадаптер содержит схему управления ЭЛТ, видеопамять, матрицы знаков и порты ввода-вывода.

Основные характеристики видеоадаптеров: режим работы (текстовый или графический), режим воспроизведения цвета (монохромный или цветной), число цветов (полутонов), разрешающая способность, емкость видеопамяти и число страниц в ней (страница – текстовый экран, который может быть выведен на отображение в мониторе), размер матрицы символа (количество пикселей в строке и столбце матрицы, формирующей символ на экране).

Емкость видеопамяти определяет количество хранимых в памяти пикселей и их атрибутов. Разрядность атрибута пикселя определяет максимальное число цветов на экране (1 байт - 256 цветов, 2 байта – 65536 цветов, 3 байта \approx 16,7 миллионов цветов). Необходимую емкость видеопамяти можно сосчитать, умножив размер атрибута на число пикселей на экране.

Видеоадаптеры IBM PC

Параметр	MGA	CGA	EGA	VGA	SVGA
Разрешающая способность	720×350	640×200 320×200	640×350 720×350	720×350 640×480	800×600 1024×768
Число цветов		2 16	16	16 256	16 256
Число строк × столбцов в текстовом режиме	25×80 80×25	25×80 80×25	25×80	25×80 (50×80)	25×80 (50×80)
Емкость видеобуфера (Кб)	64	128	128/512	256/512	512/1024
Число страниц в буфере	1	4	4-8	8	8
Размер матрицы символа, пиксели	14×9	8×8	8×8 14×8	8×8 14×8	8×8 14×8
Частота кадров, Гц	50	60	60	60	60

SVGA с видеокартой типа VESA с V=1+2 Мб обеспечивают максимальную разрешающую способность 1280×1024; видеокарта Twin Turbo-128 M2 имеет видеопамять 2 Мб (возможно до 4 Мб), цветов 256×65536

4.4. Устройства хранения информации

К устройствам хранения информации в ПК относятся внутренняя память и внешняя память. Внутренняя память: ПЗУ, ОЗУ, КЭШ, FLASH. ОРЕ, строится на микросхемах динамического типа (DRAM), а кэш-память строится на микросхемах статического типа (SRAM). Статический тип памяти более высокопроизводительный, но и более дорогой, чем динамический. ОЗУ энергозависимая память, т.е. информация в ней исчезает при отключении питания.

КЭШ-память – высокоскоростная память сравнительно небольшой емкости, являющаяся буфером между ОП и МП и позволяющая увеличить скорость выполнения операций. Регистры КЭШ-памяти недоступны для пользователя, отсюда и название КЭШ (Cache) - тайник (англ.). В КЭШ-памяти хранятся данные, которые МП получил и будет использовать в ближайшие такты своей работы. Быстрый доступ к этим данным и позволяет сохранить время выполнения очередных команд программы. При выполнении программы данные, считанные из ОП с небольшим опережением, записываются в КЭШ-память. По принципу записи результатов различают два типа КЭШ-памяти:

1. «С обратной записью» - результаты операции записываются сначала в КЭШ-память, а затем контроллер КЭШ перезаписывает их в ОП;
2. «Со сквозной записью» - результаты операций одновременно параллельно записываются и в КЭШ-память, и в ОП. КЭШ-память – энергозависимая память.

Запись в ПЗУ производится путем прожига (обычно лазером) на специальных устройствах. Объем ПЗУ ≤ несколько сот килобайтов. ПЗУ - энергонезависимое ЗУ.

В последние годы на ПК стали использоваться полупостоянные ЗУ (перепрограммируемые ЗУ) - FLASH-память. Модули или карты FLASH-памяти могут устанавливаться прямо в разъемы материнской платы : объем - 32 Кб÷4 Мб, время считывания ≈ 0,06мкс, время записи 1 байта ≈ 10 мкс. FLASH-память - энергонезависимое ЗУ. Перепрограммирование FLASH-памяти можно сделать с

дискеты или клавиатуры ПК при наличии специального контроллера или внешнего программатора, подключенного к ПК.

Структурно-основная память состоит из миллионов ячеек емкостью 1 байт каждая. Любая ячейка имеет свой уникальный адрес. Размер ОП от 1 Мб до 256 Мб. Из них ПЗУ – это 128+256 Кб, остальное ОЗУ.

Адресация ячеек памяти может быть реализована по-разному; существуют четыре типа адресации ОП (зависит от процессора и ОП):

1. Линейное адресное пространство - каждый байт пронумерован подряд - это и есть адрес.

2. Сегментированное адресное пространство, вся память разделена на блоки (сегменты) переменной длины, а сегмент на байт.

3. Страничное адресное пространство - вся память разделена на блоки (страницы) постоянной длины, а страницы - на байты.

4. Сегментно-страничное адресное пространство - вся память разделена на сегменты, которые разделены на страницы, которые содержат пронумерованные байты.

Микропроцессоры, начиная с 80486, имеют свою встроенную КЭШ-память (КЭШ 1 уровня), что обуславливает их высокую производительность. Микропроцессоры Pentium и Pentium Pro имеют КЭШ-память отдельно для данных и отдельно для команд, у Pentium $V_{кэш}=8Кб+8Кб$, у Pentium Pro $V_{кэш}=256+512Кб$.

У всех микропроцессоров существует дополнительная (внешняя) КЭШ-память 2 уровня, размещаемая на материнской плате вне МП, $V_{кэш}$ 2 уровня может достигать несколько Мб.

Логическая структура ОП

Каждая ячейка памяти имеет свой уникальный адрес. ОП имеет для ОЗУ и ПЗУ единое адресное пространство.

Адресное пространство определяет максимально возможное количество непосредственно адресуемых ячеек основной памяти. Адресное пространство зависит от разрядности адресных шин и способа организации адресного пространства.

Непосредственно адресуемая память		Расширенная память	
Стандартная память (СМА – Conventional Memory Area)	Верхняя память (блоки UMA – Upper Memory Area)	Высокая память (НМА)	Дополнительная память
640Кб	384Кб и видеобуферы ПЗУ	64Кб	

1024Кб

1088Кб

4Гб

Доступ к расширенной памяти осуществляется при использовании драйверов. *Драйвер* – это специальная программа, управляющая работой памяти или внешними устройствами ЭВМ и организующая обмен информацией между МП, ОП и внешними устройствами ЭВМ. Доступ к расширенной памяти непосредственно осуществляется только в защищенном режиме.

В реальном режиме доступ может быть реализован только при помощи драйверов:

- XMS (extended memory area) – драйвер ХММ.
- EMS (EM – Expanded Memory) – драйвер ЕММ 386.EXE или пакет QEMM.

Часто память XMS называют дополнительной, учитывая что в первых моделях ПК эта память размещалась на отдельных дополнительных платах, хотя термин Extended почти идентичен термину Expanded и переводится как расширенный или увеличенный.

Спецификация EMS является более ранней. Согласно этой спецификации доступ реализуется путем отображения отдельных полей Expanded Memory в

определенную область верхней памяти. При этом хранится не информация, а адреса, обеспечивающие доступ к этой информации. EMS называют отображаемой памятью.

64Кб высокой памяти может адресоваться при использовании драйвера HIMEM.SYS в соответствии со спецификацией XMS. Обычно используется для программ и данных ОС.

Внешняя память.

Внешние запоминающие устройства классифицируют по целому ряду признаков: по виду носителя, типу конструкции, по принципу записи и считывания информации, методу доступа и т.д.

Магнитные ленты – носители с последовательным доступом к информации, *магнитные диски* – информации с прямым доступом. «Прямой доступ» означает, что ПК может обратиться к любому месту на диске, независимо от того, где находится головка чтения/записи.

Накопители на дисках:

- 1) НГМД – флоппи-диски или дискеты;
- 2) НЖМД – «винчестер»;
- 3) Накопители на сменных жестких магнитных дисках, используя эффект Бернулли;
- 4) Накопители на флоптических дисках, floptical – накопители;
- 5) Накопители сверхвысокой плотности записи, иначе VHD – накопители;
- 6) Накопители на оптических компакт-дисках – CD-ROM (только чтение);
- 7) Накопители на оптических дисках типа CC WORM (однократная запись, многократное чтение);
- 8) Накопители на магнитооптических дисках (НМОД).

Физическая и логическая структура диска

Накопители на МД (НМД) являются наиболее распространенными внешними запоминающими устройствами в ПК. Диски бывают жесткими и гибкими, сменными и встроенными в ПК. Устройство для чтения и записи информации на МД называется дисководом. Все диски характеризуются своим диаметром или, иначе, *форм-фактором*. Наиболее распространены диски 3,5" (89 мм), 5,25" (133 мм). Диски 3,5" при меньших габаритах имеют большую емкость, меньшее время доступа и более высокую скорость чтения данных подряд, более высокую надежность и долговечность. Информация на МД записывается и считывается магнитными головками вдоль концентрических окружностей – *дорожек*. Количество дорожек и их информационная емкость зависят от типа МД, конструкции НМД, качества магнитных головок и магнитного покрытия.

Каждая дорожка разбита на *сектора*. Один сектор обычно имеет размер 512 байт; могут быть секторы размером 128 байт, 256 байт, 1024 байт. *Кластер* – минимальная единица выделения памяти на диске, кластер – это 1, 2, 4, 8 или более секторов, расположенных подряд. Размер кластера – это всегда степень 2, зависит от емкости МД. Данные на дисках хранятся в файлах.

Файл – это именованная область внешней памяти, выделенная для хранения набора однотипных данных. *Цилиндр* – совокупность дорожек МД, находящихся на одинаковом расстоянии от его центра.

Накопители на гибких МД диаметром 5,25" помещаются в плотный гибкий конверт, а диаметром 3,5" – в пластмассовую кассету. Дискета имеет 2 прорези: центральное отверстие для соединения с дисководом и смещенное от центра небольшое (скрытое футляром), определяющее радиус-вектор начала всех дорожек на ГМД. Футляр тоже имеет несколько отверстий: центральное (чуть больше, чем на дискете), широкое окно для головок чтения/записи, и боковое отверстие для защиты от записи. В последние годы появились дискеты с тефлоновым покрытием (обычно покрыты только ферролаком), которое

предохраняет магнитное покрытие и информацию от грязи, пыли, воды, жира, отпечатков пальцев и даже ацетона. Возможная емкость 3,5" – 2, 88 Мб. Существуют и дискеты под названием «Вездеход». Стойкие к пыли, температуре и влажности. Каждую новую дискету в начале работы следует отформатировать.

Форматирование дискеты – это создание структуры записи информации на ее поверхности: разметка дорожек, секторов, записи маркеров и другой служебной информации. Возможный вариант форматирования зависит от типа дискеты (на ее конверте):

- SS/SD – односторонняя, одинарной плотности;
- SS/DD – односторонняя, двойной плотности;
- DS/SD – двухсторонняя, одинарной плотности;
- DS/DD – двухсторонняя, двойной плотности;
- DS/HD – двухсторонняя, высокой плотности, обеспечивающая максимальные емкости.

Накопители на ЖД. Наиболее широкое распространение получили НЖМД типа «винчестер». Термин возник из жаргонного названия первой модели ЖД 16 Кб, у которой было 30 дорожек по 30 секторов, что случайно совпало с калибром «30/30» известного охотничьего ружья. В этих накопителях 1 или несколько ЖД вместе с блоком головок чтения/записи помещены в герметически закрытый корпус. Емкость достигает до 5Гб (1995г) благодаря очень плотной записи. Наиболее распространены ЖД 3,5", но есть и 5,25" и 1,6" (45 мм). Высота корпуса дисководов 25 мм у настольных ПК, 41 мм – у машин-серверов, 12 мм – у портативных ПК. Для того, чтобы ЖД разделить дорожки и сектора, над ним следует выполнить процедуру низкоуровневого (физического) форматирования, которую выполняет, как правило, изготовитель ЖД. В ПК имеется обычно один, реже два ЖД. MS DOS позволяет программными средствами разделить 1 физический диск на несколько «логических» дисков, имитируя несколько МД на одном накопителе (время доступа – 6мс, трансфер – 11 Мбайт/с).

Накопители на оптических дисках. Не перезаписываемые лазерно-оптические диски называют компакт-дисками (CD-ROM). Эти диски поставляются фирмой-изготовителем с уже записанной на них информацией. Запись информации на них производится при помощи лазерного луча в лабораторных условиях. Так создается первичный «мастер-диск». Процесс массового тиражирования CD-ROM по «мастер-диску» выполняется путем литья под давлением. В оптическом дисковом ПК дорожка читается лазерным лучом существенно меньшей мощности. CD-ROM имеют емкость от 250 Мб до 1,5 Гб; время доступа от 30 до 300 мс, скорость чтения информации 150-1500 Кб/с. Перезаписываемые лазерно-оптические диски с однократной (CD-R) и многократной (CD-E) записью появились в 1996г. На этих CD лазерный луч непосредственно в дисковом ПК прожигает микроскопические углубления на поверхности диска под защитным слоем; чтение записи выполняется лазерным лучом так же, как и у CD-ROM. Перезаписываемые магнитооптические диски (CC-E) используют лазерный луч для местного разогрева поверхности диска при записи информации магнитной головкой. Считывание информации выполняется лазерным лучом меньшей мощности. Емкость современных CC-E доходит до 2,6 Гб (ожидаются до 5,2 Гб), время доступа 15±150 мс, скорость считывания до 2000 Кбайт/с.

Список рекомендуемой литературы

1. Информатика: Учебник/ Под ред. проф. Н.В. Макаровой. – М.: Финансы и статистика, 1997. –768 с.
2. Информатика: Практикум по технологии работы на компьютере/ Под ред. проф. Н.В. Макаровой . – Финансы и статистика, 1997. – 384 с.
3. Лобов О.Ф., Смирнов Н.П., Чижухин Г.Н. Преобразование чисел в машинной арифметике. Учебное пособие – Пенза: Пензенский политехнический институт, 1979. – 99 с.