

## РАЗДЕЛ 1. ТЕОРЕТИЧЕСКИЕ ОСНОВЫ ИНФОРМАТИКИ И ЕЕ ОСНОВНЫЕ НАПРАВЛЕНИЯ РАЗВИТИЯ

### 1.1. Информатика сегодня: состояние и основные области исследования

В своей деятельности человек использует все большие массивы информации. Так, если с 1940 по 1950 годы объем информации удвоился примерно за 10 лет, то в настоящее время это удвоение уже происходит за 2-3 года. При работе с информацией приходится решать большое число вопросов, связанных с удобными и выгодными формами ее хранения, передачи, поиска, обработки. Кроме этого, возникают задачи, связанные с определением структуры информации. Необходимо также изучать общие свойства информации. Всем этим занимается ИНФОРМАТИКА.

В современной литературе встречается ряд определений термина «Информатика». Например, в соответствии со словарем под термином Информатика понимается: 1) научное направление, занимающееся изучением законов, методов и способов накопления, обработки и передачи информации с помощью ЭВМ и других технических средств; 2) группа дисциплин, занимающихся различными аспектами применения и разработки ЭВМ; прикладная математика, программирование, программное обеспечение, искусственный интеллект, архитектура ЭВМ, вычислительные сети; 3) научная дисциплина, изучающая структуру и общие свойства информации, а также закономерности всех процессов обмена информацией при непосредственном устном и письменном общении специалистов до формальных процессов обмена посредством различных носителей информации.

В настоящее время наиболее распространенным является представление об информатике, как о технической науке, изучающей методы автоматизации информационных процессов при помощи средств вычислительной техники и связи. Именно поэтому в ряде стран американского континента (США, Канада, страны Латинской Америки)



для обозначения этой дисциплины наиболее часто используют термин «computer science», что подчеркивает компьютерную ориентацию данной области исследований. В то же время в странах Восточной и Западной Европы более употребительным является термин французского происхождения «informatique» (информатика), который получил широкое распространение в России и в странах СНГ. Этот термин представляется более удачным, так как он подчеркивает информационную сущность данной научной дисциплины. Ведь именно информация и информационные процессы становятся главными объектами ее исследований.

По определению академиков В.М. Глушкова и С.С. Михалевича, **информатика** - это комплексная научная и инженерная дисциплина, изучающая все аспекты проектирования, разработки, создания, оценки, функционирования компьютеризированных систем переработки информации, их применение и воздействие на различные области социальной практики.

Информатика как самостоятельная дисциплина опирается на возможность и полезность выделения сведений об окружающей действительности в виде модели и физическое воплощение ее с помощью различных устройств и процессов. Формы существования информационной модели могут иметь различные физические процессы, которые могут отличаться от оригинала. Вычислительная техника является важной формой существования моделей.

Для более эффективного использования возможностей вычислительной техники при любой форме взаимодействия необходимо владеть определенным стилем мышления, определенными навыками умственных действий. Одним из наиболее существенных является умение строить информационные структуры для описания объектов и систем. Необходимо отметить, что основные научные направления информатики

образуют такие дисциплины, как теория передачи и преобразования информации, программирование и ряд других.

О.М. Белоцерковский определяет информатику как «единство трех составляющих»: hardware, software, brainware. Понятие hardware связывают с техническим обеспечением, software - с программным (рис.1.1). Особое внимание уделяется brainware (от слова «мозг», «рассудок») - проблеме, связанной с формализацией сложного процесса или явления, его математическим описанием, "...может быть самой существенной проблемой информатики».

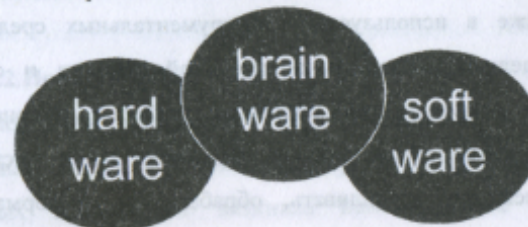


Рис. 1.1. Основные составляющие информатики

Дисциплину «информатика» можно рассматривать как "основу для унификации обозначений и терминов, методов изложения, классификации доказательства во всех учебных курсах" [8].

В настоящее время в информатике получили наибольшее развитие три следующие основные области исследований:

- теоретическая информатика (теория информации, методы информационного моделирования, компьютерная лингвистика, теория систем искусственного интеллекта, теория информационных систем);
- техническая информатика (теория и методология создания и использования технических систем сбора, хранения, обработки и передачи информации, ориентированных, в основном, на использование средств ЭВТ и связи);
- прикладная информатика (теория и методология создания и использования информационных технологий, комплексов и систем в различных сферах социальной практики).



Под теоретической (научной) информатикой будем понимать науку, изучающую структуру и общие свойства научной информации, а также закономерности всех процессов научной коммуникации.

Прикладная информатика - научное направление, объединяющее информатику, ВТ и автоматизацию.

Зародившись в недрах кибернетики - науки об общих законах получения, хранения, передачи и переработки информации в сложных системах, информатика сохраняет с ней много общего. Эта общность проявляется как в терминологическом аппарате, так и в методах исследования, а также в используемых инструментальных средствах (математические модели, средства вычислительной техники и связи, системы электронных коммуникаций). Основное свойство кибернетики заключается в том, что она пригодна для исследования любой системы, которая может записывать, накапливать, обрабатывать информацию, благодаря чему ее можно использовать в целях управления.

Однако не следует забывать, что у информатики, как и у любой фундаментальной науки, имеются свои, отличные от кибернетики, объекты исследования - информация и информационные процессы, а также свои методы исследования, одним из которых является информационный подход к изучению явлений в природе и обществе.

Основными понятиями информатики являются структуры данных различных типов, алгоритмы их переработки и компьютер как основной "переработчик". Объектом информатики являются данные (информация).

Основная задача информатики заключается в определении общих закономерностей, в соответствии с которыми происходит создание научной информации, ее преобразование, передача и использование в различных сферах деятельности человека. Прикладные задачи заключаются в разработке более эффективных методов и средств осуществления информационных процессов, в определении способов оптимальной научной коммуникации с широким применением технических средств.

В информатике, как в научном направлении, можно выделить три уровня:

- нижний (физический), представляющий собой программно-аппаратные средства ВТ и техники связи;
- средний (логический), объединяющий информационные технологии, представляющие собой совокупность моделей, методов и средств решения различных задач пользователя;
- верхний (пользовательский), на котором разработанные информационные технологии применяются для проектирования различных систем.

## 1.2. Понятие информации

Развитие человеческого общества практически на всех этапах проходило на основе технического прогресса. В процессе формирования трудовых коллективов возникла необходимость обмена знаниями. Первоначально знания передавались в устной форме из поколения в поколение, появление письменности позволило по-новому показать накопленные знания - представить их в виде информации. Предполагают, что между первыми инструментами обработки материальных объектов и средствами отображения информационных образов существовал временной интервал около миллиона лет. Таким образом, появление информации является естественным следствием развития человеческого общества.

Термин «информация» происходит от латинского слова «informatio» - разъяснение, изложение, осведомленность. К.Шеннон понимал под информацией «меру неопределенности (энтропия) события». В.М. Глушков считал, что «информация как свойство материи создает представление о ее природе и структуре, упорядоченности, разнообразии». Приведем еще ряд определений: «Информация - обозначение содержания, полученного из внешнего мира» (Н. Винер), «Информация - это передача разнообразия» (Эшли). Яглом полагал, что информация - это вероятность



выбора, Бриллюэн определил информацию как «отрицание энтропии». Энтропийная оценка оказалась перспективной.

Можно считать, что этот термин в начальном представлении является общим понятием, означающим некоторые сведения, совокупность данных, знаний и т.д. Сейчас смысл, вкладываемый в это понятие, сильно изменился и расширился, появилась особая математическая дисциплина – теория информации. Хотя в теории информации и вводится несколько строгих определений информации, они не охватывают всего богатства этого понятия. Поэтому большинство ученых в наши дни отказываются от попыток дать строгое определение и склоняются к тому, чтобы рассматривать **информацию** как первичное, неопределенное понятие, подобное понятию множества в математике.

### 1.3. Основные концепции информации

Существование множества определений информации обусловлено сложностью, специфичностью и многообразием подходов к толкованию сущности этого понятия [4, 6, 9]. Выделим три наиболее распространенные концепции информации, каждая из которых по-своему объясняет ее сущность.

Первая концепция (концепция К.Шеннона), отражая количественно-информационный подход, определяет **информацию как меру неопределенности (энтропию) события**. Под энтропией понимается математическое ожидание случайной величины собственной информации:

$$H(X_0) = \sum_{j=1}^M P(X_{0j}) I(X_{0j}),$$

где  $M$  – множество сообщений в ансамбле  $X_0$ ,  $H(X_0)$  – показывает количество двоичных единиц информации, которая содержится в любом сообщении из множества  $X_0$ .

Эта концепция основана на статическом подходе. Имеет место источник информации и потребитель, возникает процесс передачи информации. Источник выдает сообщение, потребитель, принимая

сообщение, получает при этом информацию о состоянии источника. В статической теории не изучается содержание информации. Предполагается, что до получения информации имела место некоторая неопределенность. С получением информации эта неопределенность снимается. Таким образом, статическая количественная характеристика информации – это мера снимаемой в процессе получения информации неопределенности системы. Количество информации в том или ином случае зависит от вероятности его получения: чем более вероятным является сообщение, тем меньше информации содержится в нем [6].

Рассмотрим пример, источник формирует  $M$  сообщений, каждое из которых передается неизбыточным кодом длины  $n$ . Определим число сведений  $f$  в некотором сообщении.  $f = kn$ , где  $k$  – коэффициент пропорциональности. Сделав ряд допущений, получим число передаваемых сообщений  $M = k^n$ .

За единицу количества информации примем число сведений, которые передаются двумя равновероятными сообщениями. Назовем эту единицу двоичной единицей информации. Тогда  $f=1$ ,  $M=2$ , получим  $I = k_0 \log_a 2$ . Отсюда следует, что  $a=2$ ,  $k_0=1$  и, следовательно, количество информации в сообщении составит  $I = \log_2 M$ . Приведенная формула называется формулой Хартли. Из нее следует, что для равновероятных дискретных сообщений количество информации зависит лишь от числа передаваемых сообщений.

Этот подход, хоть и не учитывает смысловую сторону информации, оказался весьма полезным в технике связи и вычислительной технике, послужил основой для измерения информации и оптимального кодирования сообщений. Кроме того, он представляет удобным для иллюстрации такого важного свойства информации, как новизна, неожиданность сообщений. При таком понимании информация – это снятая неопределенность, или результат выбора из набора возможных альтернатив.



Вторая концепция рассматривает информацию как свойство (атрибут) материи. Ее появление связано с развитием кибернетики и основано на утверждении, что информацию содержат любые сообщения, воспринимаемые человеком или приборами. Наиболее ярко и образно эта концепция информации выражена академиком В.М.Глушковым. Он писал, что "информацию несут не только испещренные буквами листы книги или человеческая речь, но и солнечный свет, складки горного хребта, шум водопада, шелест травы". Иными словами, информация как свойство материи создает представление о ее природе и структуре, упорядоченности, разнообразии и т.д. Она не может существовать вне материи, а значит, она существовала и будет существовать вечно, ее можно накапливать, хранить, перерабатывать.

Третья концепция основана на логико-семантическом (семантика - изучение текста с точки зрения смысла) подходе, при котором информация трактуется как знание, причем не любое знание, а та его часть, которая используется для активного действия, для управления и самоуправления. Иными словами, информация - это действующая, полезная, "работающая" часть знаний.

В.Г.Афанасьев, развивая логико-семантический подход, дает определение социальной информации: "Информация, циркулирующая в обществе, используемая в управлении социальными процессами, является социальной информацией. Она представляет собой знания, сообщения, сведения о социальной форме движения материи и о всех других формах в той мере, в какой она используется обществом...". Социальная информация - многоуровневое знание. Она характеризует:

- общественные процессы в целом - экономические, политические, социальные, демографические, культурно-духовные и т.д.;
- конкретные процессы, происходящие в различных ячейках общества, на предприятиях, в кооперативах, семьях и т.д.;

- интересы и стремления различных социальных групп - рабочего класса, молодежи, пенсионеров, женщин и др.

В самом общем смысле под социальной информацией понимают знания, сообщения, сведения о социальной форме движения материи и о всех других ее формах в той мере, в какой они используются обществом, вовлеченными в орбиту общественной жизни. Другими словами, информация есть содержание логического мышления, которое, воспринимаясь с помощью слышимого или видимого слова, может быть использована людьми в их деятельности [7].

#### 1.4. Виды и свойства информации

Рассмотренные подходы освещают различные стороны сущности понятия информации и облегчают тем самым систематизацию ее основных свойств. Из множества определений информации наиболее целесообразным представляется следующее: **информация** - это сведения, снимающие неопределенность об окружающем мире, которые являются объектом хранения, преобразования, передачи и использования.

**Сведения** - это знания, выраженные в сигналах, сообщениях, известиях, уведомлениях и т.д.

Понятие информации должно быть связано с определенным объектом, свойства которого она отражает. Кроме того, наблюдается относительная независимость информации от ее носителя, поскольку возможны ее преобразование и передача по различным физическим средам с помощью разнообразных физических сигналов безотносительно к ее содержанию, т.е. к семантике, что и явилось центральным вопросом многих исследований, в том числе и в философской науке. Информация о любом материальном объекте может быть получена путем наблюдения, натурного либо вычислительного эксперимента, а также на основе логического вывода. Поэтому говорят о доопытной, или **априорной**, информации и послеопытной, т.е. **апостериорной**, полученной в итоге эксперимента.



Для человека любое восприятие реальных объектов окружающей действительности происходит через ощущения. Органы чувств человека и высшая нервная система позволяют ему воспринимать объекты. При обмене информацией имеют место источник в виде объекта материального мира и приемник - человек либо какой-то материальный объект. Информация возникает за счет отражения, которое является свойством всей материи, любой материальной системы. У человека получают развитие более сложные формы отражения: познавательная и творческая. Эти формы носят сознательный характер и позволяют человеку активно воздействовать на окружающий мир.

Особенность информации в том, что, будучи материальным явлением, она не является ни материей, ни энергией. В кибернетическом смысле информация - это отражение одного объекта в другом, используемое для формирования управленческих воздействий. Говоря об информации, важно, чтобы она для потребителя имела смысл. Потребитель информации может ее оценивать в зависимости от того, где и для какой конкретной задачи информация используется. Использование информации в управлении и самоуправлении опирается на наличие связи между объектами системы, источниками информации и ее получателями. Структура сообщений, их смысл и практическая ценность всегда ориентированны на определенного получателя. Поэтому выделяют такие аспекты информации, как **прагматический, семантический и синтаксический**.

Прагматический аспект связан с возможностью достижения поставленной цели с использованием получаемой информации. Этот аспект влияет на поведение потребителя. Если информация была эффективной, то поведение потребителя меняется в желаемом направлении. Таким образом, этот аспект характеризует поведенческую сторону проблемы.

Семантический аспект позволяет оценить смысл передаваемой информации, соотнося ее с информацией, хранящейся до появления данной. Семантические связи между словами или другими смысловыми элементами отражает словарь - **тезаурус**. Он состоит из двух частей: списка слов и устойчивых словосочетаний, которые сгруппированы по смыслу, и некоторого ключа, т.е. алфавитного словаря, позволяющего расположить слова и словосочетания в определенном порядке. Тезаурус имеет особое значение в системах хранения информации, что позволяет на логическом уровне осуществлять организацию информации в виде отдельных записей, массивов и их комплексов. Семантический подход к информации базируется на анализе.

Синтаксический аспект информации связан со способом ее представления. В зависимости от реального информационного процесса, в котором участвует информация, она представляется в виде специальных символов, знаков.

Обмен информацией совершается не вообще между любыми объектами, а только между теми из них, которые представляют собой систему, обладающую каким-то минимумом организованности. В целом возникновение и развитие теории информации, а также кибернетики и информатики, явилось научным подтверждением теории отражения и способствовало ее дальнейшему развитию.

Основываясь на положении, что информация - результат и форма проявления отражения, а объектом социального отражения может быть любой фрагмент действительности (предметы, факты, явления, процессы), информация в системе подразделяется на внешнюю - порожденную другими системами управления, и внутреннюю - возникающую непосредственно в процессе функционирования самой системы как результат отражения ею соответствующих социальных явлений. Последняя есть часть социальной информации и поэтому имеет ту же природу. Особенности внутренней информации определяются объектами



отражения и теми признаками, которыми эти объекты характеризуются. Выбор объектов отражения и их признаков зависит от специфичности задач, решаемых данной социальной системой.

Предприятия ГА в своей практической деятельности сталкиваются с самыми разнообразными объектами действительности, и поэтому циркулирующая в них информация носит самый различный по содержанию характер. В связи с этим очень трудно дать такое определение информации в отрасли ГА, которое бы полностью отражало ее специфику. Ввиду многообразия информации в ГА практически невозможно дать ей такое определение, которое было бы одновременно универсально, содержательно и конкретно.

В своей повседневной деятельности сотрудники ГА широко используют различные виды информационной техники: радиопередатчики, телевидение, магнитофоны, телеграфию, вычислительную технику. Сигналы, передаваемые по радио и телевидению, а также используемые в магнитной записи, имеют форму непрерывных быстро изменяющихся во времени кривых линий. Такие сигналы называются непрерывными или аналоговыми сигналами. В противоположность этому в телеграфии и вычислительной технике сигналы имеют импульсную форму и именуется дискретными сигналами.

Исходя из сказанного можно сделать вывод, что информация передается в двух формах:

- дискретная форма представления информации - это последовательность символов, характеризующая прерывистую, изменяющуюся величину (количество авиационно-транспортных происшествий, количество рейсов авиакомпаний и т.п.);

- аналоговая или непрерывная форма представления информации - это величина, характеризующая процесс, не имеющий перерывов или промежутков (температура тела человека, скорости воздушного судна на определенном участке трассы и т.п.).

Все многообразие окружающей нас информации можно сгруппировать по различным признакам:

- по признаку "область возникновения" информация, отражающая процессы, явления неодушевленной природы называется элементарной или механической, процессы животного и растительного мира - биологической, человеческого общества - социальной;

- различают виды информации по способу передачи и восприятия. Информацию, передаваемую видимыми образами и символами, называют визуальной, звуками - аудиальной, ощущениями - тактильной, запахами и вкусами - органолептической. Информацию, выдаваемую и воспринимаемую средствами вычислительной техники, - машинной;

- информацию, создаваемую и используемую человеком, по общественному назначению можно разбить на три вида: личная, массовая и специальная. Название классов используемой информации раскрывает и их содержание, так личная информация предназначается для конкретного человека, массовая - предназначается для любого желающего ее пользоваться (общественно-политическая, научно-популярная и т.д.), специальная - предназначена для использования узким кругом лиц занимающихся решением сложных специальных задач в области науки, техники, экономики и т.п.

Часть информации, занесенная на бумажный носитель, получила название **документальной** информации. Наряду с научной информацией в сфере техники используется **техническая** информация. Она сопровождает разработки новых изделий. Научную и техническую информацию объединяют термином научно-техническая информация.

Разнообразие источников и потребителей информации привело к существованию различных **форм ее представления: символьной, текстовой и графической.** Символьная форма основана на использовании символов - букв, цифр, знаков и т.д., является наиболее простой, но она практически применяется только для передачи несложных сигналов о



различных событиях. Примером может служить зеленый свет уличного светофора, который сообщает о возможности начала движения пешеходам или водителям автотранспорта, звонок будильника дает информацию о действиях, которые должен предпринять человек, услышавший этот звуковой сигнал.

Более сложной является текстовая форма представления информации. Здесь так же, как и в предыдущей форме, используются символы: буквы, цифры, математические знаки. Однако информация заложена не только в этих символах, но и в их сочетании, порядке следования. Так, слова МИР и РИМ имеют одинаковые буквы, но содержат различную информацию. Благодаря взаимосвязи символов и отображению речи человека текстовая информация чрезвычайно удобна и широко используется в деятельности человека: книги, брошюры, журналы, различного рода документы, аудиозаписи и т.д.

Наиболее емкой и сложной является графическая форма представления информации. К этой форме относятся фотографии, схемы, чертежи, рисунки, играющие большое значение в деятельности человека вообще и сотрудника ГА, в частности.

Информация выступает как свойство объектов и явлений (процессов) порождать многообразие состояний, которые посредством отражения передаются от одного объекта к другому и запечатляются в его структуре (возможно, в измененном виде). **Свойства информации** можно рассматривать в трех аспектах: техническом - это точность, надежность, скорость передачи сигналов и т.д.; семантическом - это передача смысла текста с помощью кодов; и прагматическом - это насколько эффективно информация влияет на поведение объекта.

Целевая функция информации (ее прагматический аспект) характеризуется способностью влиять на процессы управления, на соответствующее целям управления поведение людей. В этом, по существу, и состоит полезность или ценность информации. В определенных случаях

ценность информации становится отрицательной, полезность сменяется вредностью, а сама информация становится дезинформацией. Ее источником служат субъективные факторы (мнения, взгляды, оценки), а также преднамеренные искажения информации с какой-либо целью. Если эта цель вызвана общественными интересами, то дезинформация может быть полезной (широко, например, используется дезинформация в военном деле. Чтобы ввести в заблуждение противника применяются ложные сигналы, сообщения, обозначения и т.д.). В общем же дезинформация - это ложь, искажение фактов, нарушение принципа адекватности информации, объективного освещения явлений общественной жизни.

### 1.5. Информация как ресурс

Информация охватывает все сферы, все отрасли общественной жизни, прочно входит в жизнь каждого человека, воздействует на его образ мышления и поведение. Она обслуживает общение людей, социальных групп, классов, наций и государств, помогает людям овладеть научным мировоззрением, разбираться в многообразных явлениях и процессах общественной жизни, повышать уровень своей культуры и образованности, усваивать и соблюдать законы и нравственные принципы. Огромную роль выполняет информация в управленческой деятельности. По существу, без информации не может быть и речи о любом виде управления, о целенаправленной деятельности взаимосвязанных объектов и систем.

В настоящее время разнообразная по своему значению информация, зафиксированная на специальных носителях, стала национальным богатством нового типа - **информационным ресурсом** государства. Являясь предметом купли-продажи во все времена, информация имеет свои специфические особенности: при обмене информацией ее количество увеличивается. «Если у Вас есть по яблоку, и Вы обменяетесь ими, у Вас опять будет по яблоку, но если у Вас есть по идее и мы обменяемся, то у



каждого их будет по две» [3]. Общение людей, информирование друг друга приводит к их сближению, повышению интеллектуального потенциала. У информационных ресурсов есть еще уникальное свойство - они не убывают от интенсивного использования. Более того, в процессе применения они постоянно развиваются и совершенствуются, избавляясь от ошибок и уточняя свои параметры.

В определении практической ценности информации нет каких-либо точных количественных параметров, поскольку ценность зависит от полезности информации для множества конкретных людей, ее получателей и пользователей. Ценность информации принято определять величиной тех потерь, которые эта информация предотвращает, или величиной затрат на добывание этой информации.

Анализируя информацию, мы сталкиваемся с необходимостью оценки качества и определения количества получения информации. Определить качество информации чрезвычайно сложно, а часто и вообще невозможно. Какие-либо сведения, например исторические, могут десятилетиями считаться ненужными и вдруг их ценность может резко возрасти. Вместе с этим определить количество информации не только нужно, но и можно. Это прежде всего необходимо для того, чтобы сравнить друг с другом массивы информации, определить, какие размеры должны иметь материальные объекты (бумага, магнитная лента и т.д.), хранящие эту информацию.

### 1.6. Количественная оценка информации

Для определения количества информации необходимо представить любую ее форму (символьную, текстовую, графическую) в едином виде, т.е. преобразовать эти формы информации так, чтобы она получила единый стандартный вид. Таким видом стала так называемая двоичная форма представления информации. Она заключается в записи любой информации в виде последовательности только двух символов.

Эти символы могут на бумаге обозначаться любым способом: буквами А, Б; словами ДА, НЕТ. Однако ради простоты записи взяты цифры 1 и 0. В электронном аппарате, хранящем либо обрабатывающем информацию, рассматриваемые символы могут также обозначаться по-разному: один из них - наличием в рассматриваемой точке электрического тока либо магнитного поля, второй - отсутствием в этой точке электрического тока либо магнитного поля.

Рассмотрим простейший случай получения информации. Вы задаете только один вопрос: "Летная ли погода?". При этом условимся, что с одинаковой вероятностью ожидаете ответ: "ДА" или "НЕТ". Легко увидеть, что любой из этих ответов несет самую малую порцию информации. Эта порция определяет единицу измерения информации, называемую **битом**. Благодаря введению понятия единицы информации появилась возможность определения размера любой информации числом битов. Образно говоря, если, например, налет воздушного судна определяют в часах, то объем информации - в битах.

Условимся каждый положительный ответ представлять цифрой 1, а отрицательный - цифрой 0. Тогда запись всех ответов образует многозначную последовательность цифр, состоящую из нулей и единиц, например 100110. Процесс получения двоичной информации об объектах исследования называют **кодированием информации**.

В информационных документах широко используются не только русские, но и латинские буквы, цифры, математические знаки и другие специальные знаки всего примерно 200-250 символов. Поэтому для кодировки всех указанных символов используется восьмиразрядная последовательность цифр 0 и 1. Следует отметить, что указанный способ кодирования используется тогда, когда к нему не предъявляются дополнительные требования, например, необходимо указать на возникшую ошибку, исправление ошибки, обеспечить секретность информации. В



этих случаях применяют специальное кодирование, при использовании которого коды получаются длиннее.

Для представления графической информации в двоичной форме используется так называемый точечный способ. На первом этапе вертикальными и горизонтальными линиями делят изображение. Чем больше при этом получилось квадратов, тем точнее будет передана информация о картинке.

Информация с точки зрения ее возникновения и совершенствования проходит следующий путь: человек наблюдает некоторый факт окружающей действительности. Этот факт отражается в виде совокупности данных, при последующем структурировании в соответствии с конкретной предметной областью данные превращаются в знания. Таким образом, верхним уровнем информации как результата отражения окружающей действительности (результата мышления) являются знания. Знания возникают как итог теоретической и практической деятельности. Информация в виде знаний отличается высокой структуризацией. На основе структуризации информации формируется информационная модель объекта. По мере развития общества информация как совокупность научно-технических данных и знаний превращается в базу системы информационного обслуживания научно-технической деятельности общества.

### 1.7. Основные направления развития информатики

В настоящее время можно выделить несколько направлений развития информатики:

1) *теоретическая информатика* - центральное место в этом направлении занимает общая теория информации.

2) *теория информационного моделирования* явлений природы, объектов и систем техносферы, социальных процессов и систем биосферы - быстрое развитие методов информационного моделирования, особенно в сочетании с методами глобального информационного мониторинга, могут

обеспечить уже сегодня возможность прогнозирования многих кризисных ситуаций в регионах повышенной социальной и политической напряженности, в районах экологического бедствия, в местах природных катастроф и крупных технологических аварий. Следует отметить перспективное научное направление в области методологии моделирования сложных систем, которое развивает группа российских ученых во главе с профессором И.Л. Букатовой. Это направление, названное авторами эвоинформатикой, имеет целью создание новых интеллектуальных информационных метатехнологий для решения задач глобального моделирования сложных природных явлений и технических систем на основе эволюционного подхода (например, для решения задачи исследования устойчивости и живучести биосферы нашей планеты с учетом все возрастающего антропогенного воздействия на нее в различных регионах земного шара).

3) *синтеллектика* - теория интеллектуальных информационных систем. Ее главными компонентами будут методы представления и использования в информационных системах нечетких знаний, интеграции и кооперативного использования разнородных информационных ресурсов, а также методы создания обучаемых информационных систем, основанных на принципах когнитивной графики и интегрированного использования различных форм представления информации (образов, текстов, музыки, речи).

4) *инструментальные средства технической реализации* информационных процессов - главное внимание здесь должно быть уделено средствам сбора, хранения и обработки информации, построенным на новых физических, архитектурных и алгоритмических принципах (например, оптоэлектронные, голографические и нейроподобные информационно-вычислительные системы).

5) *социальная информатика: информационная медиэкология* - наука об изучении возможностей человека по адаптации ко все более



быстро возрастающим информационным нагрузкам; социальная когнитология - наука о закономерностях формирования и использования совокупного интеллектуального потенциала общества; информдинамика - наука о закономерностях изменения социальных структур общества под воздействием процессов его информатизации и т.д. Многие из перечисленных новых научных направлений находятся в настоящее время в начальной стадии своего формирования.

6) биологическая информатика, изучающая общие закономерности и особенности протекания информационных процессов в объектах биосферы- живых организмах и растениях. Результаты ряда исследований, проводимых в этой области профессором Д.С.Чернавским с использованием математического моделирования и методов динамической теории информации, убедительно свидетельствуют, что информация и информационные процессы, несомненно, играют ключевую роль в эволюции живого вещества. Данное направление должно помочь вскрыть и понять некоторые общие закономерности информационных процессов в природе, которые затем можно будет использовать при изучении информационных процессов в социальной сфере, а также при конструировании и исследовании различного рода знаковых, языковых, технических и других искусственно создаваемых человеком информационных систем и информационных технологий.

7) минероинформатика - исследования информационных процессов в неживой природе. Такие процессы имеют место не только в естественных минералах (особенно кристаллической структуры), но также и в металлах, и жидкостях. Некоторые свойства металлов и жидкостей запоминать и регенерировать полученную информацию уже используют при конструировании отдельных технических устройств и технологических процессов.

8) энионика - исследование информационных процессов, происходящих вне пределов уже доступного измерения при помощи современной

научной аппаратуры материального мира. В настоящее время накоплено довольно большое количество документально подтвержденных фактов, свидетельствующих о том, что процессы такого рода объектно существуют, хотя, последствия их проявления воспринимаются в основном не с помощью научных приборов, а через посредство людей, обладающих экстрасенсорными способностями. Такие явления, как телепатия, телекинез, ясновидение и лечение на расстоянии, уже давно не являются секретом и практически используются. Поэтому проведение научных экспериментов в этой области, накопление и систематизация результатов их изучения научными методами сегодня не только уместно, но и необходимо. На основании этого будет создана теория паранормальных энергоинформационных процессов (энионика).

#### 1.8. Основные положения концепции информатизации общества, необходимость информатизации ГА

Последние три десятилетия проходят под знаком бурно развивающегося глобального процесса информатизации общества. Этот процесс охватил практически все страны мира и позволяет создать интеллектуальный фундамент для решения многих глобальных проблем современности, содействует преодолению энергетического и экологического кризисов.

Одним из важнейших механизмов, затрагивающих все основные направления реформирования различных отраслей России, является ее информатизация, которая рассматривается как необходимое условие и важнейший этап информатизации России в целом. Основой перехода от индустриального этапа развития общества к информационному являются современные информационные технологии (ИТ).

Вместе с тем известно, что в нашей стране наблюдается заметное отставание в области ИТ, особенно в плане обеспечения материально-технической базой.



В целях устранения создавшегося положения в стране была разработана государственная программа, базирующаяся на концепции информатизации общества. Основные положения этой концепции предусматривают:

1. Достижение всеобщей компьютерной грамотности.
2. Создание материально-технической базы, обеспечивающей широкое использование компьютерной техники и технологий на уровне мировых стандартов.
3. Разработка и внедрение единого фонда программного обеспечения.
4. Эффективное использование потенциала информационно-вычислительных систем во всех сферах жизни общества.

"Информационный взрыв" охватил и деятельность служб ГА. Все возрастающая сложность и масштабность стоящих перед ними задач объективно потребовали использование больших массивов информации для выработки, принятия и организации исполнения оптимального управленческого решения.

Практика убедительно подтверждает, что отсутствие достоверной, своевременной и полной информации порождает субъективизм, необоснованность решений и действий, несовместимых с научным управлением, что особенно недопустимо в деятельности предприятий ГА, к числу которых относится и МГТУ ГА. Поэтому совершенствование деятельности ГА непосредственно зависит от использования современной информационной технологии и техники. Для этого службы ГА должны решить следующие задачи:

- подготовка специалистов в области применения компьютерной техники, в том числе овладение соответствующими знаниями и навыками всеми сотрудниками;
- внедрение автоматизированных информационно-поисковых систем;
- создание общих автоматизированных банков данных;

- создание локальных, республиканских и международных информационных сетей;

- обеспечение всех подразделений ГА компьютерной техникой, создание автоматизированных рабочих мест;

- подготовка прикладных программ по всем направлениям деятельности служб ГА;

- создание и внедрение дистанционных систем обучения.

Несомненно, что успешное решение этих задач позволит поднять на качественно более высокий уровень деятельность всех подразделений и служб ГА. Использование в ГА достижений информационной технологии в большей степени зависит от перестройки ВУЗовского образования. Современный специалист должен не только иметь представление о возможностях, которые дает ему информатизированное общество, но и эффективно использовать их.

С этой точки зрения существенны три образовательных элемента информатики:

- аппаратная реализация электронно-вычислительных машин, т.е. знакомство с ее архитектурой и программным обеспечением (общая предварительная и обязательная подготовка к работе с компьютером);

- специальные компьютерные системы, используемые в ГА;

- основы системного анализа, теории информации и управления применительно к ГА.

Актуальной задачей становится повышение производительности управленческого труда, способности работников перерабатывать большие объемы информации. Традиционные методы переработки информации сегодня не срабатывают. Практика деятельности ГА в нашей стране и за рубежом показывает, что решение этой задачи может быть осуществлено только на основе новых технологий, на внедрении информационно-аналитического обеспечения, которое подразумевает внедрение компьютеров.



Анализ современного состояния информатики и направлений перспективного развития научных исследований в области формирующегося в настоящее время комплекса наук об информации позволяет сделать следующие основные выводы:

- информатика сегодня - это быстро прогрессирующая и стратегически важная область научного знания. Это одна из «точек роста» мировой науки, из которой в ближайшие годы будет сформирован новый комплекс наук об информации. Именно он станет научной базой наступающего постиндустриального информационного общества;

- основными факторами, стимулирующими развитие информатики, являются, с одной стороны, - процесс информатизации общества, который в настоящее время принял глобальный характер и является стержнем научно-технического, экономического и социального развития практически для всех стран мирового сообщества, а, с другой стороны, - проблема необходимости разрешения кризиса в самой науке, где сегодня происходит пересмотр многих сложившихся ранее понятий и представлений о законах природы и общества.

#### Контрольные вопросы

- 1) Что понимается под информацией?
- 2) Как осуществляется количественная оценка информации?
- 3) Что является верхним уровнем информации как результата отражения окружающей действительности?
- 4) Как определить ценность информации?
- 5) Что понимается под информационным ресурсом?
- 6) Перечислите существующие аспекты информации.
- 7) Перечислите основные составляющие информатики.
- 8) Приведите примеры использования в ГА достижений информационных технологий.
- 9) Перечислите основные направления развития информатики.

## РАЗДЕЛ 2. ПРОГРАММНОЕ ОБЕСПЕЧЕНИЕ ЭЛЕКТРОННО-ВЫЧИСЛИТЕЛЬНЫХ МАШИН

Современный компьютер представляет собой сочетание двух важнейших компонент: аппаратуры, состоящей из целого ряда устройств, и программного обеспечения, представляющего собой совокупность программ по управлению работой компьютера и обработке поступающей в компьютер информации. Программное обеспечение для компьютера играет примерно такую же роль, как двигатель для автомобиля, без которого он просто не может двигаться. Между этими компонентами компьютера существует меняющееся со временем соотношение в стоимости «материального» (аппаратура) и «интеллектуального» (программы) продукта, причем не всегда в пользу первого. Как правило, это зависит от степени сложности процессов обработки информации на компьютере.

### 2.1. Состав программного обеспечения ЭВМ

**Программное обеспечение (Software)** - совокупность программ и правил, позволяющая использовать ЭВМ для решения различных задач.

По своему назначению программы, выполняемые компьютером, можно разделить на три основные группы:

- системные и сервисные программы;
- языки программирования (система программирования);
- прикладные программы.

**Прикладное программное обеспечение (ППО)** - совокупность программ, предназначенная для решения конкретных задач. Прикладное программное обеспечение разрабатывается самим пользователем в зависимости от интересующей его задачи. В некоторых источниках можно встретить деление прикладного программного обеспечения на кроссовое и тестовое.

Кроссовое программное обеспечение (КПО) - предназначено для



работы с соответствующей микро-ЭВМ и ее программами, но реализованное на вычислительных машинах других классов (больших или мини-ЭВМ) или на микро ЭВМ другого типа.

**Тестовое программное обеспечение (ТПО)** - совокупность предназначенных для проверки работоспособности устройств, входящих в состав микро ЭВМ на стадиях ее изготовления, эксплуатации и ремонта.

**Системное программное обеспечение (СПО)** - совокупность программ и языковых средств, предназначенных для поддержания функционирования ЭВМ и наиболее эффективного выполнения его целевого назначения. По функциональному назначению в системном программном обеспечении можно выделить две системы:

- операционную систему;
- систему программирования.

**Система программирования (СП)** - совокупность языка программирования и соответствующего ему языкового процессора, обеспечивающие автоматизацию отработки и отладки программ. Программные компоненты системы программирования работают под управлением операционной системы наравне с прикладными программами пользователя.

Системные программы можно разделить на две группы:

а) программы, записанные в постоянную память компьютера и составляющие базовую систему ввода/вывода (в дальнейшем БСВВ или BIOS);

б) программы, записанные во внешней памяти, основную часть которых составляет операционная система (в дальнейшем ОС или OS).

Системные программы в BIOS являются промежуточным звеном между программным обеспечением компьютера и его электронными компонентами. Эти программы обеспечивают выполнение всех операций ввода/вывода, соответствующих специфическим особенностям работы каждого из периферийных устройств данного компьютера.

**Операционная система (Operating system)** - 1) набор специальных программ, обеспечивающих работоспособность компьютерной системы, управление аппаратурой и прикладными программами, интерфейс с пользователем; 2) совокупность системных программ, записанных в память компьютера в процессе загрузки ОС, которые осуществляют общее планирование ресурсами и управление всеми действиями компьютера; 3) комплекс программ, предназначенный для обеспечения определенного уровня эффективности вычислительной системы за счет автоматизированного управления ее работой и предоставляемого пользователям услуг.

Основными функциями операционной системы являются:

- управление аппаратурой,
- управление процессами ввода/вывода и передачи данных на внешние устройства,
- поддержка файловой системы,
- управление работой прикладных программ,
- администрирование (защита информации, поддержка системы паролей в многопользовательских системах, учет использования ресурсов вычислительной системы),
- поддержка сетевого взаимодействия.

Иногда в функции операционной системы включается поддержка различных систем программирования (programming system). ОС не зависит, вообще говоря, от конструкции компьютера, однако она может использовать все "встроенные" специальные функции BIOS по набору предоставляемых возможностей. Кроме того, обращение к функциям OS гораздо проще для работы пользователя, так как представляют собой более высокий уровень программного обеспечения, чем обращение к функциям BIOS напрямую. Существует множество различных операционных систем. На IBM-совместимых персональных компьютерах распространены системы MS-DOS, MS Windows, OS/2, LINUX.



## 2.2. Принципы распространения и тестирования ПО

Существует несколько принципов распространения и тестирования программного обеспечения:

1) **freeware** — принцип свободного распространения программного обеспечения, включающий обычно следующие пункты:

- потребитель может использовать программу в любых целях и модифицировать ее, как сочтет нужным;
- изготовитель не несет никакой ответственности за правильность работы программы и последствия ее применения;
- изготовитель может (но не обязан) давать консультации по использованию программы или ее настройке для решения задач пользователя.

Распространением программ по принципу **freeware** обычно занимаются некоммерческие организации или программисты-энтузиасты. Примером воплощения такого принципа может служить проект GNU по разработке системы программирования для языка C++.

2) **shareware** — принцип распространения программных продуктов, который обычно включает в себя следующие условия:

- пользователь может бесплатно использовать данный продукт в течение некоторого времени для проверки его пригодности к решению своих задач;
- в течение указанного периода он не должен извлекать коммерческую выгоду из этого продукта (например, не может его продавать);
- по истечении этого периода пользователь должен либо прекратить использование программы, либо официально зарегистрироваться (т.е. купить ее).

Как правило, по принципу **shareware** распространяются демонстрационные версии программ, имеющие некоторые ограничения в

использовании. После регистрации пользователь получает программу со всеми правами, предоставляемыми производителем.

В процессе создания программного обеспечения, разрабатываемые программные продукты проходят несколько этапов тестирования:

- **alpha test** — тестирование первого уровня;
- **beta test** — тестирование второго уровня.

**Alpha test** — это первый этап полного тестирования программы, при котором обнаруживаются и устраняются ошибки, несоответствия с исходными спецификациями и т.п.;

**Beta test** — это второй этап полного тестирования программы. После этого этапа программа считается в целом готовой к использованию и распространению. Поскольку в больших программных продуктах, даже после полного тестирования, вероятность наличия ошибок остается достаточно велика, готовая программа, прошедшая бета-тестирование распространяется в виде так называемой бета-версии. Это означает, что фирма-производитель распространяет данный продукт, осознавая возможность некорректной работы данной программы в некоторых ситуациях. Обычно, после сбора информации об эксплуатации данной программы широким кругом пользователей в течение нескольких месяцев, фирма-производитель устраняет обнаруженные ошибки и выпускает уже окончательную версию программы.

## 2.3. Классификация операционных систем

Операционные системы могут различаться особенностями реализации внутренних алгоритмов управления основными ресурсами компьютера (процессорами, памятью, устройствами), особенностями использованных методов проектирования, типами аппаратных платформ, областями использования и многими другими свойствами.

Рассмотрим характеристики ОС, связанные с управлением только одним типом ресурсов — процессором. В табл.2.1 приведена классификация ОС по нескольким наиболее основным признакам.



Таблица 2.1

## Классификация ОС

Основные признаки	Классификация
особенности алгоритмов управления ресурсами	-многозадачные, однозадачные; -многопользовательские, однопользовательские - многопроцессорные, однопроцессорные - поддерживающие многоплатформенную обработку
особенности аппаратных платформ	- персональные компьютеры; - мини-компьютеры; - кластеры; - сети ЭВМ
особенности областей использования	- системы пакетной обработки; - системы разделения времени; - системы реального времени
особенности методов построения	- способы построения ядра системы; -объектно-ориентированный подход; -наличие нескольких прикладных сред; - распределенная организация

## 2.3.1 Особенности алгоритмов управления ресурсами

От эффективности алгоритмов управления локальными ресурсами компьютера во многом зависит эффективность всей операционной системы в целом. Поэтому, характеризуя ОС, часто приводят важнейшие особенности реализации функций ОС по управлению ресурсами (процессорами, памятью, внешними устройствами) автономного компьютера [5].

В зависимости от особенностей использованного алгоритма управления процессором, операционные системы делят на многозадачные и однозадачные, многопользовательские и однопользовательские, на многопроцессорные и однопроцессорные системы, на системы, поддерживающие многоплатформенную обработку и не поддерживающие ее.

По числу одновременно выполняемых задач операционные системы могут быть разделены на два класса: *однозадачные* (например, MS-DOS) и *многозадачные* (OS/2, UNIX, Windows9x).

Однозадачные ОС в основном выполняют функцию предоставления пользователю виртуальной машины, делая более простым и удобным процесс взаимодействия пользователя с компьютером. Однозадачные ОС включают средства управления периферийными устройствами, средства управления файлами, средства общения с пользователем.

Многозадачные ОС, кроме вышеперечисленных функций, управляют разделением совместно используемых ресурсов, таких как процессор, оперативная память, файлы и внешние устройства.

По числу одновременно работающих пользователей ОС делятся на *однопользовательские* (MS-DOS, Windows3.x, ранние версии OS/2); *многопользовательские* (UNIX, Windows NT). Главным отличием многопользовательских систем от однопользовательских является наличие средств защиты информации каждого пользователя от несанкционированного доступа других пользователей.

Важнейшим разделяемым ресурсом является процессорное время. Способ распределения процессорного времени между несколькими одновременно существующими в системе процессами (или нитями) во многом определяет специфику ОС. Среди множества существующих вариантов реализации многозадачности можно выделить две группы алгоритмов: *невывесняющая многозадачность* (NetWare, Windows 3.x) и *вывесняющая многозадачность* (Windows NT, OS/2, UNIX). Основным различием между вывесняющими и невывесняющими вариантами многозадачности является степень централизации механизма планирования процессов. В первом случае механизм планирования процессов целиком сосредоточен в операционной системе, а во втором - распределен между системой и прикладными программами. При невывесняющей многозадачности активный процесс выполняется до тех



пор, пока он сам, по собственной инициативе, не отдаст управление операционной системе для того, чтобы та выбрала из очереди другой готовый к выполнению процесс. При вытесняющей многозадачности решение о переключении процессора с одного процесса на другой принимается операционной системой, а не самим активным процессом.

Важным свойством операционных систем является возможность распараллеливания вычислений в рамках одной задачи. *Многонитевая ОС* разделяет процессорное время не между задачами, а между их отдельными ветвями (нитеями).

Другим важным свойством ОС является отсутствие или наличие в ней средств поддержки *многопроцессорной обработки* мультипроцессорирование. Мультипроцессорирование приводит к усложнению всех алгоритмов управления ресурсами. В наши дни становится общепринятым введение в ОС функций поддержки многопроцессорной обработки данных. Такие функции имеются в операционных системах Solaris 2.x фирмы Sun, Open Server 3.x компании Santa Crus Operations, OS/2 фирмы IBM, Windows NT/2000/XP фирмы Microsoft и NetWare 4.1 фирмы Novell.

Многопроцессорные ОС могут классифицироваться по способу организации вычислительного процесса в системе с многопроцессорной архитектурой: *асимметричные ОС* и *симметричные ОС*.

Асимметричная ОС целиком выполняется только на одном из процессоров системы, распределяя прикладные задачи по остальным процессорам.

Симметричная ОС полностью децентрализована и использует весь пул процессоров, разделяя их между системными и прикладными задачами.

Важное влияние на облик операционной системы в целом, на возможности ее использования в той или иной области, оказывают особенности и других подсистем управления локальными ресурсами –

подсистем управления памятью, файлами, устройствами ввода-вывода.

Специфика ОС проявляется и в том, каким образом она реализует сетевые функции: распознавание и перенаправление в сеть запросов к удаленным ресурсам, передача сообщений по сети, выполнение удаленных запросов. При реализации сетевых функций возникает комплекс задач, связанных с распределенным характером хранения и обработки данных в сети: ведение справочной информации о всех доступных в сети ресурсах и серверах, адресация взаимодействующих процессов, обеспечение прозрачности доступа, тиражирование данных, согласование копий, поддержка безопасности данных.

### 2.3.2. Особенности аппаратных платформ

На свойства операционной системы непосредственное влияние оказывают *аппаратные средства*, на которые она ориентирована. По типу аппаратуры различают операционные системы персональных компьютеров, мини-компьютеров, мейнфреймов, кластеров и сетей ЭВМ. Среди перечисленных типов компьютеров могут встречаться как однопроцессорные варианты, так и многопроцессорные. В любом случае специфика аппаратных средств, как правило, отражается на специфике операционных систем.

Очевидно, что ОС большой машины является более сложной и функциональной, чем ОС персонального компьютера. Так в ОС больших машин функции по планированию потока выполняемых задач, очевидно, реализуются путем использования сложных приоритетных дисциплин и требуют большей вычислительной мощности, чем в ОС персональных компьютеров. Аналогично обстоит дело и с другими функциями.

Сетевая ОС имеет в своем составе средства передачи сообщений между компьютерами по линиям связи, которые совершенно не нужны в автономной ОС. На основе этих сообщений сетевая ОС поддерживает разделение ресурсов компьютера между удаленными пользователями, подключенными к сети. Для поддержания функций передачи сообщений



сетевые ОС содержат специальные программные компоненты, реализующие популярные коммуникационные протоколы, такие как IP, IPX, Ethernet и другие.

Многопроцессорные системы требуют от ОС особой организации, с помощью которой сама операционная система, а также поддерживаемые ею приложения могли бы выполняться параллельно отдельными процессорами системы. Параллельная работа отдельных частей ОС создает дополнительные проблемы для разработчиков ОС, так как в этом случае гораздо сложнее обеспечить согласованный доступ отдельных процессов к общим системным таблицам, исключить эффект гонок и прочие нежелательные последствия асинхронного выполнения работ.

*Кластер* - слабо связанная совокупность нескольких вычислительных систем, работающих совместно для выполнения общих приложений, и представляющих пользователю единую систему. Наряду со специальной аппаратурой для функционирования кластерных систем необходима и программная поддержка со стороны операционной системы, которая сводится в основном к синхронизации доступа к разделяемым ресурсам, обнаружению отказов и динамической реконфигурации системы. Одной из первых разработок в области кластерных технологий были решения компании Digital Equipment на базе компьютеров VAX.

Наряду с ОС, ориентированными на совершенно определенный тип аппаратной платформы, существуют операционные системы, специально разработанные таким образом, чтобы они могли быть легко перенесены с компьютера одного типа на компьютер другого типа так называемые *мобильные ОС*. Наиболее ярким примером такой ОС является популярная система UNIX. В этих системах аппаратно-зависимые места тщательно локализованы, так что при переносе системы на новую платформу переписываются только они.

### 2.3.3. Особенности областей использования

Операционная система может выполнять запросы в разных режимах,

поэтому ее можно разделить на следующие типы:

- системы пакетной обработки (например, ОС ЕС),
- системы разделения времени (UNIX, VMS),
- системы реального времени (QNX, LynxOS, VxWorks).

Системы пакетной обработки предназначались для решения задач в основном вычислительного характера, не требующих быстрого получения результатов. Главной целью и критерием эффективности систем пакетной обработки является максимальная пропускная способность, то есть решение максимального числа задач в единицу времени. Для одновременного выполнения выбираются задачи, предъявляющие отличающиеся требования к ресурсам, так, чтобы обеспечивалась сбалансированная загрузка всех устройств вычислительной машины; так, например, в мультипрограммной смеси желательно одновременное присутствие вычислительных задач и задач с интенсивным вводом-выводом. Таким образом, выбор нового задания из пакета заданий зависит от внутренней ситуации, складывающейся в системе, то есть выбирается "выгодное" задание. Следовательно, в таких ОС невозможно гарантировать выполнение того или иного задания в течение определенного периода времени. В системах пакетной обработки переключение процессора с выполнения одной задачи на выполнение другой происходит только в случае, если активная задача сама отказывается от процессора, например, из-за необходимости выполнить операцию ввода-вывода. Взаимодействие пользователя с вычислительной машиной, на которой установлена система пакетной обработки, сводится к тому, что он приносит задание, отдает его диспетчеру-оператору, а в конце дня после выполнения всего пакета заданий получает результат. Очевидно, что такой порядок снижает эффективность работы пользователя.

Системы разделения времени обеспечивают одновременное обслуживание многих пользователей, позволяют любому пользователю взаимодействовать со своим заданием. Эффект одновременной работы



достигается разделением процессорного времени и других ресурсов между несколькими вычислительными процессами, которые заданы разными пользователями. Операционная система выстраивает очередь из поступающих заданий, выделяет квант времени для доступа к центральному процессору каждому заданию согласно очереди. Выполнив первое задание, операционная система отсылает его в конец очереди и переходит ко второму и т. д. Критерием эффективности систем разделения времени является не максимальная пропускная способность, а удобство и эффективность работы пользователя.

Одним из основных свойств операционных систем реального времени (ОС РВ) является их способность изолировать друг от друга приложения, поэтому если в программе возникает сбой или выполняются какие-то нелегальные операции, ОС может быстро блокировать программу, инициировать восстановление и защиту других программ либо самой системы от серий вредоносных команд. Та же самая защита предотвращает переполнение стеков памяти, вызываемое действиями любых программ. Так как для системы реального времени главным является ее способность вовремя отреагировать на внешние события, то такой параметр, как время реакции системы на прерывание, (interrupt latency) является ключевым.

Критерием эффективности для систем реального времени является их способность выдерживать заранее заданные интервалы времени между запуском программы и получением результата (управляющего воздействия).

К операционным системам реального времени предъявляются следующие основные требования [5]:

1. ОС РВ должна быть многонитевой и допускать вытеснение (preemptible).
2. Диспетчеризация должна осуществляться на базе приоритетов.
3. Механизм синхронизации нитей должен быть предсказуемым.

4. Должна существовать система наследования приоритетов.

5. Временные характеристики ОС должны быть предсказуемы и известны.

Различают два типа ОС реального времени: один из них характеризуют «жестким реальным временем», второй — «мягким». Если не выполняется обработка критических ситуаций либо она происходит недостаточно быстро, система жесткого реального времени прерывает операцию и блокирует ее, чтобы не пострадала надежность и готовность остальной части системы. Системы мягкого реального времени более «снисходительны» и «терпят» определенные, некритичные ошибки.

Выпускается довольно много операционных систем реального времени. К числу наиболее популярных продуктов относятся VxWorks и VxWorks AE, разработанные компанией Wind River Systems. Последняя представляет собой архитектуру высокой готовности с распределенной передачей сообщений и поддержкой отказоустойчивости. ОС позволяет программистам изолировать совместно используемые библиотеки, данные и системное программное обеспечение, а также приложения.

LynxOS — жесткая ОС реального времени, которая работает с Unix и Java. Операционная система QNX функционирует на процессорах архитектуры Intel и имеет микроядро размером всего 10 Кбайт.

К операционным системам реального времени, созданным в научных учреждениях, относится Chimera Университета Карнеги-Меллона, многопроцессорная многозадачная система реального времени, которая упрощает конфигурацию и повторное использование кода программистами. Chimera предназначена для роботов и автоматизированных систем. Операционная система, созданная в Университете штата Мэриленд, получила название Maruti. Она поддерживает как жесткое, так и мягкое реальное время.

В течение многих лет приложения на базе ОС реального времени использовались во встроенных системах специального назначения, а с



недавнего времени они стали применяться повсюду, от управляемого компьютером медицинского оборудования до кофеварок.

### 2.3.4. Особенности методов построения

При описании операционной системы часто указываются особенности ее структурной организации и основные концепции, положенные в ее основу. К таким базовым концепциям относятся:

1) *Способы построения ядра системы* - монолитное ядро или микроядерный подход. Большинство ОС использует монолитное ядро, которое компонуется как одна программа, работающая в привилегированном режиме и использующая быстрые переходы с одной процедуры на другую, не требующие переключения из привилегированного режима в пользовательский и наоборот. Альтернативой является построение ОС на базе микроядра, работающего также в привилегированном режиме и выполняющего только минимум функций по управлению аппаратурой, в то время как функции ОС более высокого уровня выполняют специализированные компоненты ОС - серверы, работающие в пользовательском режиме. При таком построении ОС работает более медленно, так как часто выполняются переходы между привилегированным режимом и пользовательским, зато система получается более гибкой - ее функции можно наращивать, модифицировать или сужать, добавляя, модифицируя или исключая серверы пользовательского режима. Кроме того, серверы хорошо защищены друг от друга, как и любые пользовательские процессы.

2) *Построение ОС на базе объектно-ориентированного подхода* дает возможность использовать все его достоинства, хорошо зарекомендовавшие себя на уровне приложений, внутри операционной системы, а именно: аккумуляцию удачных решений в форме стандартных объектов; возможность создания новых объектов на базе имеющихся с помощью механизма наследования; хорошую защиту данных за счет их инкапсуляции во внутренние структуры объекта, что делает данные

недоступными для несанкционированного использования извне; структурированность системы, состоящей из набора хорошо определенных объектов.

3) *Наличие нескольких прикладных сред* дает возможность в рамках одной ОС одновременно выполнять приложения, разработанные для нескольких ОС. Многие современные операционные системы поддерживают одновременно прикладные среды MS-DOS, Windows, UNIX (POSIX), OS/2 или хотя бы некоторого подмножества из этого популярного набора. Концепция множественных прикладных сред наиболее просто реализуется в ОС на базе микроядра, над которым работают различные серверы, часть которых реализуют прикладную среду той или иной операционной системы.

4) *Распределенная организация* операционной системы позволяет упростить работу пользователей и программистов в сетевых средах. В распределенной ОС реализованы механизмы, которые дают возможность пользователю представлять и воспринимать сеть в виде традиционного однопроцессорного компьютера. Характерными признаками распределенной организации ОС являются: наличие единой справочной службы разделяемых ресурсов единой службы времени, использование механизма вызова удаленных процедур (RPC) для прозрачного распределения программных процедур по машинам, многокритерной обработки, позволяющей распараллеливать вычисления в рамках одной задачи и выполнять эту задачу сразу на нескольких компьютерах сети, а также наличие других распределенных служб.

### 2.4. Требования к современным ОС

Главным требованием, предъявляемым к операционной системе, является способность выполнения основных функций: эффективного управления ресурсами и обеспечения удобного интерфейса для пользователя и прикладных программ. Современная операционная



система, кроме функциональных требований, должна соответствовать не менее важным рыночным требованиям. К этим требованиям относятся:

- *расширяемость*- код должен быть написан таким образом, чтобы можно было легко внести дополнения и изменения, не нарушив целостность системы;
- *переносимость* – код должен легко переноситься с процессора одного типа на процессор другого типа и с аппаратной платформы одного типа на аппаратную платформу другого типа;
- *надежность и отказоустойчивость*- система должна быть защищена как от внутренних, так и внешних ошибок, сбоев, отказов, ее действия должны быть всегда предсказуемыми;
- *совместимость* - ОС должна иметь средства для выполнения прикладных, написанных для других операционных систем. Пользовательский интерфейс должен быть совместим с существующими системами и стандартами;
- *безопасность* – операционная система должна обладать средствами защиты ресурсов одних пользователей от других;
- *производительность* – система должна обладать настолько хорошим быстродействием и временем реакции, насколько это позволяет аппаратная платформа.

## 2.5.Современные архитектуры файловых систем

Разработчики новых операционных систем стремятся обеспечить пользователя возможностью работать сразу с несколькими файловыми системами. В новом понимании файловая система состоит из многих составляющих, в число которых входят и файловые системы в традиционном понимании.

Новая файловая система имеет многоуровневую структуру, на верхнем уровне которой располагается так называемый переключатель файловых систем (например, в ОС Windows - устанавливаемый диспетчер файловой системы - installable filesystem manager, IFS). Он обеспечивает

интерфейс между запросами приложения и конкретной файловой системой, к которой обращается это приложение. Переключатель файловых систем преобразует запросы в формат, воспринимаемый следующим уровнем - уровнем файловых систем.

Каждый компонент уровня файловых систем выполнен в виде драйвера, соответствующей файловой системы и поддерживает определенную организацию файловой системы. Переключатель является единственным модулем, который может обращаться к драйверу файловой системы. Приложение не может обращаться к нему напрямую. Драйвер файловой системы может быть написан в виде реентерабельного кода, что позволяет сразу нескольким приложениям выполнять операции с файлами. Каждый драйвер файловой системы в процессе собственной инициализации регистрируется у переключателя, передавая ему таблицу точек входа, которые будут использоваться при последующих обращениях к файловой системе.

Для выполнения своих функций драйверы файловых систем обращаются к подсистеме ввода-вывода, образующей следующий слой файловой системы новой архитектуры. Подсистема ввода вывода - это составная часть файловой системы, которая отвечает за загрузку, инициализацию и управление всеми модулями низших уровней файловой системы. Обычно эти модули представляют собой драйверы портов, которые непосредственно занимаются работой с аппаратными средствами. Кроме этого подсистема ввода-вывода обеспечивает некоторый сервис драйверам файловой системы, что позволяет им осуществлять запросы к конкретным устройствам.

Большое число уровней архитектуры файловой системы обеспечивает большую гибкость - драйвер может получить управление на любом этапе выполнения запроса - от вызова приложением функции, которая занимается работой с файлами, до того момента, когда работающий на самом низком уровне драйвер устройства начинает



просматривать регистры контроллера. Многоуровневый механизм работы файловой системы реализован посредством цепочек вызова.

В ходе инициализации драйвер устройства может добавить себя к цепочке вызова некоторого устройства, определив при этом уровень последующего обращения. Подсистема ввода-вывода помещает адрес целевой функции в цепочку вызова устройства, используя заданный уровень для того, чтобы должным образом упорядочить цепочку. По мере выполнения запроса, подсистема ввода-вывода последовательно вызывает все функции, ранее помещенные в цепочку вызова.

На рис.2.1 представлена структура Windows NT, в которой каждое из приложений обращается к сервисным функциям (серверам) косвенно, через вызовы локальных процедур (LPC), реализованных в диспетчере LPC, являющемся частью NT Executive и работающем в привилегированном режиме. Приложения Win32 исполняются как отдельные многопоточные процессы. Программы Win16 могут запускаться как однопоточные процессы на общей виртуальной машине, или на собственной виртуальной машине, что обеспечивает им большую степень защищенности от других программ Win16. Приложения DOS выполняются как отдельные процессы на отдельных виртуальных DOS-машинах (VDM). Подсистемы OS/2 и POSIX обеспечивают работу соответствующих прикладных программ в текстовом режиме.

Windows NT Executive предоставляет сервисные функции ОС, необходимые для подсистем пользовательского режима и реализует внутренние механизмы системы, такие, например, как планирование нитей и управление памятью. Слой системных сервисных функций служит интерфейсом между программами пользовательского режима и NT Executive.

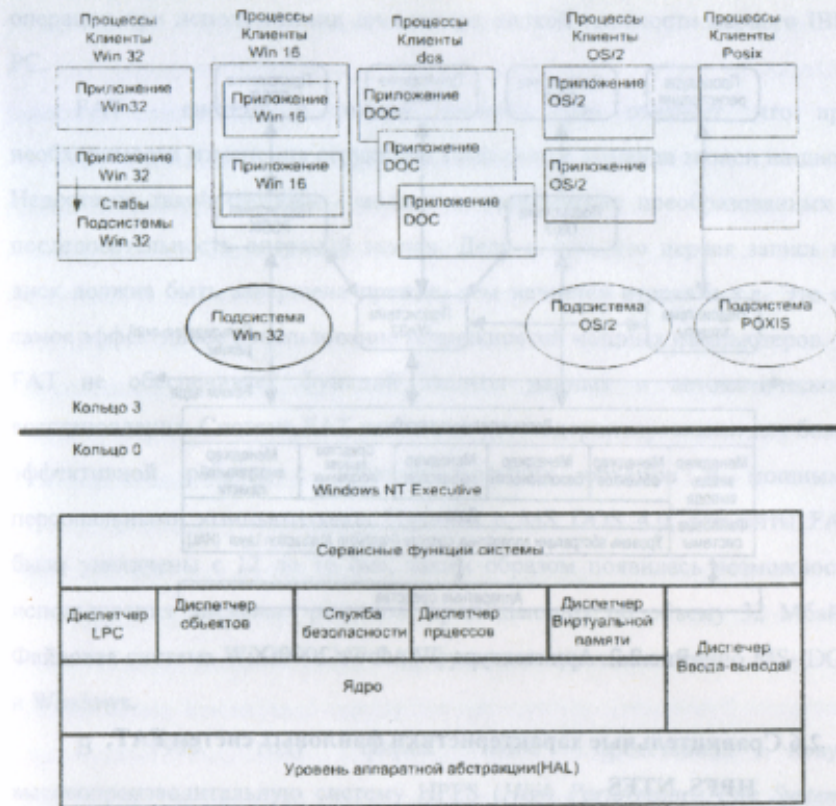


Рис.2.1. Архитектура ОС Windows NT

Ядро обрабатывает прерывания и исключительные ситуации, переключает нити, синхронизирует процессоры в многопроцессорных системах и выполняет другие низкоуровневые функции, используемые при работе NT Executive.

Разбиение версий Windows на два семейства - NT и 9x было вызвано скорее не стратегическими соображениями, а тактикой борьбы за пользователей, ввиду угрозы перетекания последних на более компактную и менее ресурсоемкую (по сравнению с Windows NT) OS/2. Очевидные слабости Windows 9x говорят о том, что долговременная стратегия Microsoft связана с линией Windows NT - Win2000/XP.

На рис.2.2 приведена архитектура Windows 2000/X.



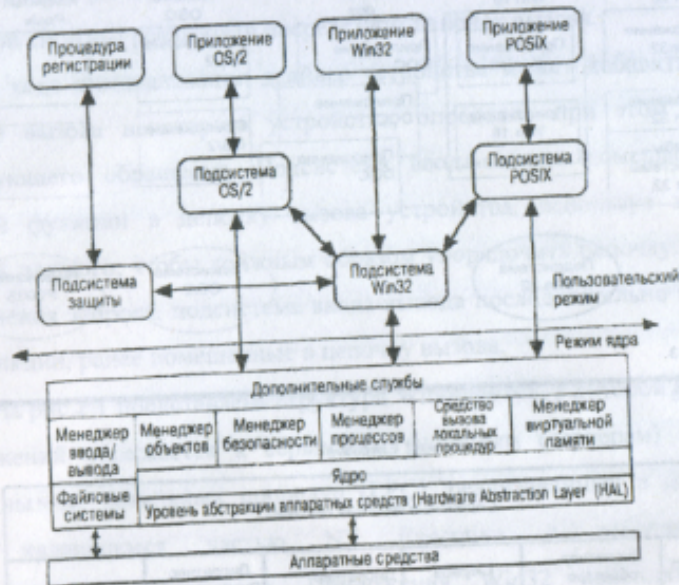


Рис.2.2. Архитектура Windows 2000/X

## 2.6. Сравнительные характеристики файловых систем FAT, HPFS, NTFS

Первый персональный компьютер, представленный фирмой IBM, работал под управлением MS DOS, и имел 16-битовый процессор 8088 и два дисководов гибких дисков низкой плотности. Файловая система MS DOS FAT (названная так из-за своей таблицы размещения файлов - *file allocation table*) предоставляла более чем достаточные возможности для форматирования дисков небольших объемов и поддержки иерархической структуры каталогов и файлов. Затем появились дисководы двойной, а позже - высокой плотности. Система FAT все еще оставалась основной файловой системой, в то время как мощности аппаратного и программного обеспечения возрастали с каждым годом. Однако из-за использования этой файловой системы на поиск файлов и считывание данных с больших жестких дисков уходило значительно больше времени, чем на аналогичные

операции при использовании дисководов низкой плотности первого IBM PC.

FAT - система с точной записью. Это означает, что при необходимости изменения структуры тома дается команда записи на диск. Недостаток такой системы - медленное выполнение преобразованных в последовательность операций записи. Дело в том, что первая запись на диск должна быть завершена прежде, чем начнется вторая и т.д. Это не самое эффективное использование возможностей мощных компьютеров. FAT не обеспечивает функций защиты данных и автоматического восстановления. Система FAT несколько раз модернизировалась для более эффективной работы с дисками больших объемов и мощными персональными компьютерами. Начиная с MS DOS 4.0, элементы FAT были увеличены с 12 до 16 бит, таким образом появилась возможность использования дисковых разделов, превышающих по объему 32 Мбайт. Файловая система Windows NT FAT функционирует аналогично MS-DOS и Windows.

В 1990 году фирма IBM представила новую высокопроизводительную систему HPFS (*High Performance File System*), вошедшую в состав ОС OS/2 версии 1.2. HPFS разрабатывалась специально для использования на дисках большого объема, подключенных к 16-битовым компьютерам. Через некоторое время после выхода HPFS в свет была представлена HPFS386. Эта файловая система стала частью MS LAN Manager и использовала все возможности 32-битовых процессоров 386.

HPFS имеет особенности, способствующие эффективному управлению большими объемами жесткого диска. HPFS поддерживает длинные имена файлов (до 255 символов). Когда том форматируется под HPFS, первые 18 секторов резервируются для блока начальной загрузки, суперблока и запасного блока. Эти структуры используются для загрузки



ОС, поддержки файловой системы и восстановления при возможных ошибках.

В HPFS резервируется пространство под два битовых массива объемом 2Кб для каждого дискового интервала в 16Мб. Каждый массив отводит по одному биту для каждого размещаемого блока (равного одному сектору) в полосе 8 Мб, показывая, какие размещаемые блоки используются. Битовые массивы поочередно размещаются в конце и начале каждой полосы, обеспечивая таким образом максимальное количество непрерывного пространства для данных (16 Мб). Кроме того, запись новых файлов планируется так, что между новым и существующим файлами оставался свободный участок, и каждый файл имел возможность расширения в непрерывном дисковом пространстве. Это свойство HPFS помогает осуществлять быстрый поиск данных и минимизировать фрагментацию файлов.

Другая особенность, объясняющая быстрый поиск в каталоге - технология B-tree. Эта древовидная структура с корнем и несколькими узлами содержит данные, организованные некоторым логическим способом. Корень содержит административную информацию, карту для остальной структуры и, возможно, некоторые данные. Большинство данных содержится в узлах. С большими каталогами технология B-tree работает гораздо эффективнее линейных списков, используемых FAT.

HPFS применяет B-tree для структуризации каждого файла и каталога. Каждый каталог указывает на структуры Fnode для содержащихся в нем файлов. Структура Fnode (ее размер 512 байтов) содержит заголовок, имя файла (усеченное до 15 символов), длину файла, расширенные атрибуты, список контроля доступа (ACL) и расположение данных файла.

Следует подчеркнуть, что HPFS эффективно работает на дисках объемом до 2 Гб. Однако есть у нее и слабые стороны. Например, если повреждена первая часть тома с информацией начальной загрузки и

указателем на корневой каталог, том использовать невозможно. Кроме того, HPFS предполагает применение 512-байтовых секторов, которые не очень годятся для больших томов.

HPFS - система с отложенной записью. Работа с данными производится через буфер ввода/вывода. Пока пользователь читает файлы или просматривает каталоги, необходимые для записи, данные накапливаются в кэше, следовательно, ждать окончания процесса записи не нужно. Запись данных на диск производится только в момент низкой загрузки ресурсов компьютера. Недостаток систем с отложенной записью в том, что в случае сбоя диска восстановление данных займет гораздо больше времени, чем в системе с точной записью.

Из-за высокой производительности и универсальности FAT и HPFS стали популярными файловыми системами и широко используются.

Новая файловая система Windows NT - NTFS- *Windows NT file system* - совместима с этими системами и имеет множество новых возможностей, таких как защита от сбоев, восстановление и защита данных от несанкционированного доступа.

NTFS разработана специально для максимального использования возможностей современных персональных компьютеров, в которых может находиться несколько мощных процессоров и несколько жестких дисков большого объема. NTFS обеспечивает сочетание эффективности, надежности и совместимости, невозможное в FAT или HPFS.

Она разработана для быстрого выполнения стандартных файловых операций вроде чтения, записи и поиска, а так же улучшенных операций, например, восстановления файловой системы на очень больших жестких дисках. Способ считывания данных зависит от метода хранения информации о расположении файлов, используемого системой. NTFS, подобно HPFS, для увеличения производительности использует двоичные деревья.



Каждый распределенный на томе NTFS сектор принадлежит некоторому файлу. Частью файла являются даже метаданные файловой системы (информация, описывающая непосредственно файловую систему). Каждый файл на томе NTFS представлен записью в специальном файле - главной файловой таблице (Master File Table — MFT). NTFS резервирует первые 16-ть записей таблицы для специальной информации. Первая запись таблицы описывает непосредственно главную файловую таблицу. За ней следует зеркальная запись MFT. Если первая запись MFT разрушена, NTFS читает вторую запись для отыскания зеркального файла MFT, первая запись которого идентична первой записи MFT. Местоположения сегментов данных MFT и зеркального файла MFT записаны в секторе начальной загрузки. Дубликат сектора начальной загрузки находится в логическом центре диска. Третья запись MFT - файл регистрации, применяемый для восстановления файлов. Семнадцатая и последующие записи главной файловой таблицы используются собственно файлами и каталогами на томе.

Файловая система NTFS — это восстанавливаемая файловая система, сочетающая быстрдействие файловой системы с отложенной записью и практически мгновенное восстановление. Каждая операция ввода/вывода, изменяющая файл на томе NTFS, рассматривается файловой системой как транзакция и может выполняться как неделимый блок. При модификации файла пользователем сервис файла регистрации фиксирует всю информацию, необходимую для повторения или отката транзакции. Если транзакция завершена успешно, производится модификация файла. Если нет, NTFS производит откат транзакции, следуя инструкциям в информации отмены. При обнаружении в транзакции ошибки, транзакция прокручивается обратно.

Важная особенность NTFS — отложенная передача (lazy commit) - позволяет минимизировать затраты на регистрацию транзакций и подобна отложенной записи.

NTFS поддерживает длинные имена файлов (до 255 символов). В имени файла используются символы UNICODE, что позволяет создавать файлы, содержащие, например, символы кириллицы.

## 2.7. Классификация системных программ

Системные программы подразделяются на: драйверы, программные оболочки, вспомогательные программы (утилиты) [1].

Драйверы устройств - это специальные резидентные программы, которые дополняют систему ввода/вывода и обеспечивают обслуживание дополнительных внешних устройств или нестандартное использование имеющихся устройств.

Программные оболочки - обеспечивают наиболее удобный и наглядный способ общения с компьютером, чем штатные средства ОС.

Вспомогательные программы (утилиты) - к системным программам можно отнести большое количество утилит, т.е. программ вспомогательного назначения. Чаще всего используются следующие типы утилит:

- программы резервирования - позволяют быстро скопировать нужную информацию, находящуюся на жестком диске компьютера;
- антивирусные программы - предназначены для предотвращения заражения ПК;
- программы упаковщики (архиваторы);
- программы русификаторы - приспособливают другие программы (обычно ОС) для работы с русскими буквами (текстами пользователя);
- программы для диагностики компьютера - позволяют проверить конфигурацию компьютера и работоспособность его устройств;
- программы кэши для диска - убыстряют доступ к информации на дисках путем организации в оперативной памяти кэш-буфера, содержащего наиболее часто используемые участки диска;



- программы для оптимизации диска - позволяют обеспечить более быстрый доступ к информации на диске за счет оптимизации размещения данных на диске;

- программы динамического сжатия дисков - создают псевдодиски, информация которых в сжатом виде хранится в виде файлов на обычных (настоящих дисках) ПК, что позволяет хранить больше данных (DriveSpace, DoubleSpace, Stacker);

- программы ограничения доступа позволяют защищать хранящиеся на компьютере данные от нежелательных или неквалифицированных пользователей.

Наиболее известный интегрированный набор программ Norton Utilities, позволяющих находить и устранять различные ошибки компьютера, повышать его быстродействие, выполнять диагностику и профилактику сбоев. Пакет программ, которые можно условно разделить на 4 группы:

#### 1) Recovery - утилиты для восстановления информации на дисках:

- *Disk Doctor (ndd.exe)* - автоматическое устранение некоторых дефектов на дисках, по-прежнему лучшая программа восстановления дисков;
- *Unerase (Unerase.exe), Smartcan* - восстановление удаленных файлов и каталогов;
- *Unformat (Unformat.exe)* - восстановление удаленных файлов и каталогов на диске после его форматирования;
- *Disk Editor* - мощная утилита использует улучшенный режим восстановления (Advanced Recovery Mode);

#### 2) SPEED- утилиты для повышения скорости чтения/записи на диске:

- *Speed Disk (Speeddisk.exe)* - дефрагментирование записей на дисках;
- *Norton Cache, Calibrate, Ncache2* - создание в ОП специальной области (кэша), повышающей скорость чтения/записи.

#### 3) SECURITY - утилиты для защиты информации от несанкционированного доступа:

- *Disk Monitor* - защищает диск от несанкционированной записи;

- *DiskReet* - позволяет выбрать разные методы защиты данных от любопытного взгляда;

- *WipeInfo* - для удаления конфиденциальной информации.

#### 4) TOOLS - утилиты общего назначения:

- *Disk Tools* - необходима для восстановления дисков, а также позволяет сделать диск загрузочным;
- *Image u Rescue* - создает диск спасения или страховочную копию данных;
- *SysInfo u Ndiag* - информация о конфигурации и производительности вашего компьютера;
- *DupDisk* - программа дублирования гибких дисков;
- *NCC* - помощь при настройке режимов работы аппаратуры компьютера;
- *FileFind* - в существующих файлах;
- *Safe Format* - обеспечивает быстрое и безопасное форматирование;
- *Batch Enhancer (be.exe)* - дает прекрасную возможность автоматизировать рутинные операции поиска, последовательного подключений и выполнения всевозможных программ, работая с ней, даже новичок с легкостью создаст удобное, красочное и звучащее окно-меню.

Кроме того, в состав пакета входит оригинальный командный процессор NDOS, выполняющий кроме всех команд, что дает определенные преимущества пользователю, сравнительное тестирование системы, дисков и мультимедиа, позволяя оценить быстродействие компьютера.

Технология LiveUpdate корпорации Symantec, включенная в Norton Utilities, позволяет своевременно обновлять Norton Utilities и файлы описания вирусов. LiveUpdate использует модем или подключение в Internet для автоматической загрузки обновлений непосредственно из Symantec.

#### 2.8. Общие принципы архивации. Классификация методов

Подавляющее большинство современных форматов записи данных содержат их в виде, удобном для быстрого манипулирования, для удобного прочтения пользователями. При этом данные занимают объем больший, чем это действительно требуется для их хранения. Алгоритмы,



которые устраняют избыточность записи данных, называются алгоритмами сжатия данных, или *алгоритмами архивации*. В настоящее время существует огромное множество программ для сжатия данных, основанных на нескольких основных способах.

Алгоритмы сжатия данных по определению имеют своей основной задачей устранение избыточности, то есть корреляций между данными во входном тексте. Все алгоритмы сжатия данных качественно делятся на:

1) алгоритмы сжатия без потерь, при использовании которых данные на приемной восстанавливаются без малейших изменений (используются в криптосистемах);

2) алгоритмы сжатия с потерями, которые удаляют из потока данных информацию, незначительно влияющую на суть данных, либо вообще не воспринимаемую человеком (такие алгоритмы сейчас разработаны только для аудио- и видео- изображений).

Существует два основных метода архивации без потерь:

- алгоритм Хаффмана (англ. Huffman), ориентированный на сжатие последовательностей байт, не связанных между собой;

- алгоритм Лемпеля-Зива (англ. Lempel, Ziv), ориентированный на сжатие любых видов текстов, то есть использующий факт неоднократного повторения "слов" – последовательностей байт.

Практически все популярные программы архивации без потерь (ARJ, RAR, ZIP и т.п.) используют объединение этих двух методов – алгоритм LZH.

### 2.8.1 Алгоритм Хаффмана

Данный алгоритм основан на том факте, что некоторые символы из стандартного 256-символьного набора в произвольном тексте могут встречаться чаще среднего периода повтора, а другие, соответственно, – реже. Следовательно, если для записи распространенных символов использовать короткие последовательности бит, длиной меньше 8, а для

записи редких символов – длинные, то суммарный объем файла уменьшится.

Хаффман предложил очень простой алгоритм определения того, какой символ необходимо кодировать каким кодом для получения файла с длиной, очень близкой к его энтропии (то есть информационной насыщенности). Пусть у нас имеется список всех символов, встречающихся в исходном тексте, причем известно количество появлений каждого символа в нем. Выпишем их вертикально в ряд в виде ячеек будущего графа по правому краю листа (рис.2.3а). Выберем два символа с наименьшим количеством повторений в тексте (если три или большее число символов имеют одинаковые значения, выбираем любые два из них). Проведем от них линии влево к новой вершине графа и запишем в нее значение, равное сумме частот повторения каждого из объединяемых символов (рис.2.3б). Далее не будем принимать во внимание при поиске наименьших частот повторения два объединенных узла (для этого сотрем числа в этих двух вершинах), но будем рассматривать новую вершину как полноценную ячейку с частотой появления, равной сумме частот появления двух соединившихся вершин. Будем повторять операцию объединения вершин до тех пор, пока не придем к одной вершине с числом (рис.2.3в и 2.3г).

Расставим на двух ребрах графа, исходящих из каждой вершины, биты 0 и 1 произвольно – например, на каждом верхнем ребре 0, а на каждом нижнем – 1. Теперь для определения кода каждой конкретной буквы необходимо просто пройти от вершины дерева до нее, выписывая нули и единицы по маршруту следования. Для рис.2.3 символ "А" получает код "000", символ "Б" – код "01", символ "К" – код "001", а символ "О" – код "1".

В теории кодирования информации показывается, что код Хаффмана является префиксным, то есть код никакого символа не является началом кода какого-либо другого символа. Проверьте это на нашем примере. А из



этого следует, что код Хаффмана однозначно восстановим получателем, даже если не сообщается длина кода каждого переданного символа. Получателю пересылают только дерево Хаффмана в компактном виде, а затем входная последовательность кодов символов декодируется им самостоятельно без какой-либо дополнительной информации. Например, при приеме "0100010100001" им сначала отделяется первый символ "Б" : "01-00010100001", затем снова начиная с вершины дерева – "А" "01-000-10100001".

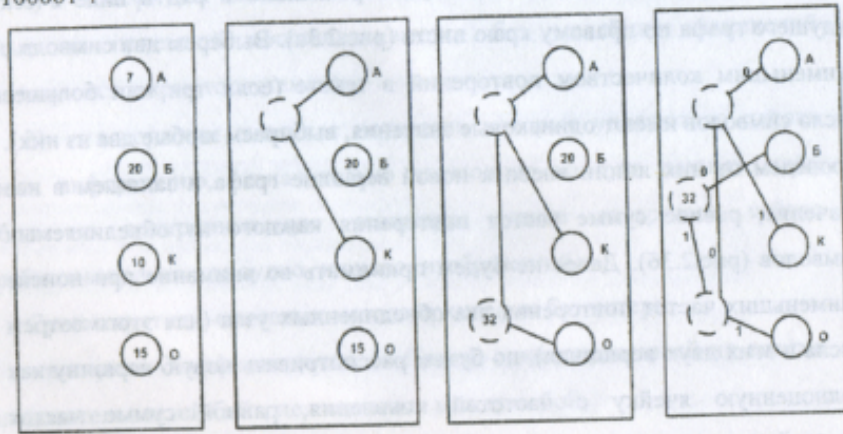


Рис.2.3.Алгоритм Хаффмана

### 2.8.2.Алгоритм Лемпеля-Зива

Классический алгоритм Лемпеля-Зива – LZ77, названный так по году своего опубликования, предельно прост. Он формулируется следующим образом: "Если в прошедшем ранее выходном потоке уже встречалась подобная последовательность байт, причем запись о ее длине и смещении от текущей позиции короче чем сама эта последовательность, то в выходной файл записывается ссылка (смещение, длина), а не сама последовательность". Так фраза "КОЛОКОЛ\_ОКОЛО\_КОЛОКОЛЬНИ" закодируется как "КОЛО(-4,3)\_(-5,4)О\_(-14,7)БНИ".

Распространенный метод сжатия RLE (Run Length Encoding), который заключается в записи вместо последовательности одинаковых

символов одного символа и их количества, является подклассом данного алгоритма. Рассмотрим, например, последовательность "AAAAAAA". С помощью алгоритма RLE она будет закодирована как "(A,7)", в то же время ее можно достаточно хорошо сжать и с помощью алгоритма LZ77 : "A(-1,6)". Действительно, степень сжатия именно такой последовательности им хуже (примерно на 30-40%), но сам по себе алгоритм LZ77 более универсален, и может намного лучше обрабатывать последовательности, вообще несжимаемые методом RLE.

### Контрольные вопросы

- 1) Что понимается под программным обеспечением?
- 2) Приведите классификацию программного обеспечения.
- 3) Что понимается под операционной системой?
- 4) Назовите ее основные функции.
- 5) Назовите отличия симметричных и асимметричных операционных систем.
- 6) Какие требования предъявляются к современным операционным системам?
- 7) Поясните особенности мобильных операционных систем.
- 8) Какие файловые системы являются системой с отложенной записью?
- 9) Поясните технологию B-tree.
- 10) Приведите классификацию системных программ.
- 11) Что понимается под архивацией?
- 12) Поясните общие принципы архивации.
- 13) Приведите пример работы алгоритма Хаффмана.



### РАЗДЕЛ 3. Архитектуры вычислительных машин и систем, перспективные типы процессоров ЭВМ

Рассматривая архитектуру ЭВМ, вычислительных систем, суперкомпьютеров и информационно-вычислительных сетей с общих позиций и абстрагируясь от деталей, можно воспользоваться классификацией Скиллиорна [2], построенной на следующих элементах-объектах:

- процессор команд (IP - Instruction Processor) - функциональное устройство, работающее как интерпретатор команд (может отсутствовать);
- процессор данных (DP - Data Processor) - функциональное устройство, работающее как преобразователь данных, в соответствии с арифметическими операциями;
- иерархия памяти (IM - Instruction Memory, DM - Data Memory) - запоминающее устройство, в котором хранятся данные и команды, пересылаемые между процессорами;
- переключатель - абстрактное устройство, обеспечивающее связь между процессорами и памятью.

Таким образом, имеются процессоры и блоки памяти - информационно-вычислительная среда и средства коммутации и коммуникации — коммуникационно-коммутационная среда. Все эти компоненты активно присутствуют как в ЭВМ, так и в вычислительных сетях и системах (суперЭВМ).

#### 3.1. Классификация Флинна

В 1966 г. М. Флинном (Flynn) был предложен следующий подход к классификации архитектур вычислительных систем. В основу было положено понятие потока, под которым понимается последовательность элементов, команд или данных, обрабатываемая процессором. Соответствующая система классификации основана на рассмотрении числа потоков инструкций и потоков данных и описывает четыре базовых класса:

1) *Архитектура ОКОД (SISD)* охватывает все однопроцессорные и одномашинные варианты систем, т.е. с одним вычислителем. Все ЭВМ классической структуры попадают в этот класс. Здесь параллелизм вычислений обеспечивается путем совмещения выполнения операций отдельными блоками арифметическо-логического устройства (АЛУ), а также параллельной работой устройств ввода-вывода информации и процессора. Закономерности организации вычислительного процесса в этих структурах достаточно хорошо изучены.

2) *Архитектура ОКМД (SIMD)* предполагает создание структур векторной или матричной обработки. Системы этого типа обычно строятся как однородные, т.е. процессорные элементы, входящие в систему, идентичны, и все они управляются одной и той же последовательностью команд. Однако каждый процессор обрабатывает свой поток данных. Под эту схему хорошо подходят задачи обработки матриц или векторов (массивов), задачи решения систем линейных и нелинейных, алгебраических и дифференциальных уравнений, задачи теории поля и др. В структурах данной архитектуры желательно обеспечивать соединения между процессорами, соответствующие реализуемым математическим зависимостям. Как правило, эти связи напоминают матрицу, в которой каждый процессорный элемент связан с соседними. По данной схеме строились системы: первая суперЭВМ — ILLIAC-IV, отечественные параллельные системы — ПС-2000, ПС-3000. Идея векторной обработки широко использовалась в таких известных суперЭВМ, как Cyber-205 и Gray-I, II, III. Узким местом подобных систем является необходимость изменения коммутации между процессорами, когда связь между ними отличается от матричной. Кроме того, класс задач, допускающих широкий матричный параллелизм, весьма узок. Структуры ВС этого типа, по существу, являются структурами специализированных суперЭВМ.

Элементы технологии SIMD реализованы в процессорах Intel, начиная с Pentium MMX (1997 г.).



3) *Третий тип архитектуры* — *МКОД (MISD)* предполагает построение своеобразного процессорного конвейера, в котором результаты обработки передаются от одного процессора к другому по цепочке. Выгоды такого вида обработки понятны. Прототипом таких вычислений может служить схема любого производственного конвейера. В современных ЭВМ по этому принципу реализована схема совмещения операций, в которой параллельно работают различные функциональные блоки, и каждый из них делает свою часть в общем цикле обработки команды. В ВС этого типа конвейеры должны образовывать группы процессоров. Однако при переходе на системный уровень очень трудно выявить подобный регулярный характер в универсальных вычислениях. Кроме того, на практике нельзя обеспечить и «большую длину» такого конвейера, при которой достигается наивысший эффект. Вместе с тем конвейерная схема нашла применение в так называемых скалярных процессорах суперЭВМ, в которых они применяются как специальные процессоры для поддержки векторной обработки.

4) *Архитектура МКМД (MIMD)* предполагает, что все процессоры системы работают по своим программам с собственным потоком команд. В простейшем случае они могут быть автономны и независимы. Такая схема использования ВС часто применяется на многих крупных вычислительных центрах для увеличения пропускной способности центра. Большой интерес представляет возможность согласованной работы ЭВМ (процессоров), когда каждый элемент делает часть общей задачи. Подобные системы могут быть многомашинными и многопроцессорными. Например, отечественный проект машины динамической архитектуры (МДА) - ЕС-2704, ЕС-2727 - предполагал одновременное использование сотни процессоров.

Наличие большого разнообразия систем, образующих класс МКМД (MIMD), делает классификацию Флинна не полностью адекватной. Действительно и 4-х процессорный SX-5 компании NEC и 1000-

процессорный Сгау ТЗЕ попадают в этот класс, и это заставляет искать другие подходы к классификации.

### 3.2. Классификация Джонсона

Е. Джонсон предложил проводить классификацию MIMD-архитектур на основе структуры памяти и реализации механизма взаимодействия и синхронизации между процессорами.

По структуре оперативной памяти существующие вычислительные системы делятся на две большие группы:

- системы с общей памятью, прямо адресуемой всеми процессорами;
- системы с распределенной памятью, каждая часть которой доступна только одному процессору.

Одновременно с этим и для межпроцессорного взаимодействия существуют две альтернативы: через разделяемые (общие) переменные или с помощью механизма передачи сообщений.

Основываясь на таком делении, Джонсон вводит следующие наименования для некоторых классов:

- *системы с разделяемой памятью* - вычислительные системы, использующие общую разделяемую память для межпроцессорного взаимодействия и синхронизации (класс 1);
- *архитектуры с передачей сообщений* - системы, в которых память распределена по процессорам, а для взаимодействия и синхронизации используется механизм передачи сообщений (класс 3);
- *гибридные архитектуры* - системы с распределенной памятью и синхронизацией через разделяемые переменные (класс 2).

### 3.3. Классификация Базу

По мнению А. Базу (A. Basu), любую параллельную вычислительную систему можно однозначно описать последовательностью решений, принятых на этапе ее проектирования, а сам процесс проектирования представить в виде дерева. Корень дерева — это вычислительная система, и



последующие ярусы дерева, фиксируя уровень параллелизма, метод реализации алгоритма, параллелизм инструкций и способ управления, последовательно дополняют друг друга, формируя описание системы.

На первом этапе определяется, какой уровень параллелизма использует вычислительная система. Одна и та же операция может одновременно выполняться над целым набором данных, определяя параллелизм на уровне данных. Способность выполнять более одной операции одновременно говорит о параллелизме на уровне команд. Если компьютер спроектирован так, что целые последовательности команд могут быть выполнены одновременно, то говорят о параллелизме на уровне задач.

Второй уровень в классификационном дереве фиксирует метод реализации алгоритма. С появлением сверхбольших интегральных схем (СБИС) стало возможным реализовывать аппаратно не только простые арифметические операции, но и алгоритмы целиком. Например, быстрое преобразование Фурье, перемножение матриц и другие относятся к классу тех алгоритмов, которые могут быть эффективно реализованы в СБИС. Данный уровень классификации разделяет системы с аппаратной реализацией алгоритмов и системы, использующие традиционный способ программной реализации.

Третий уровень конкретизирует тип параллелизма, используемого для обработки инструкций машины: конвейеризация инструкций или их независимое (параллельное) выполнение. В большей степени этот выбор относится к компьютерам с программной реализацией алгоритмов, так как аппаратная реализация всегда предполагает параллельное исполнение команд.

Четвертый уровень данной классификации определяет способ управления, принятый в вычислительной системе: синхронный или асинхронный. Если выполнение команд происходит в строгом порядке, определяемом только сигналами таймера и счетчиком команд, то говорят о синхронном способе управления. Если для инициации команды

определяющими являются такие факторы, как готовность данных, то машина попадает в класс с асинхронным управлением.

### 3.4. PVP-архитектура

PVP (Parallel Vector Process) - параллельная архитектура с векторными процессорами. Основным признаком PVP-систем является наличие векторно-конвейерных процессоров, в которых предусмотрены команды однотипной обработки векторов независимых данных, эффективно выполняющиеся на конвейерных функциональных устройствах. Как правило, несколько таких процессоров работают одновременно с общей памятью (аналогично SMP-Symmetric Multi-Processing) в рамках многопроцессорных конфигураций. Несколько таких узлов могут быть объединены с помощью коммутатора. Поскольку передача данных в векторном формате осуществляется намного быстрее, чем в скалярном (максимальная скорость может составлять 64 Гбайт/с, что на два порядка быстрее, чем в скалярных машинах), то проблема взаимодействия между потоками данных при распараллеливании становится несущественной. Таким образом, системы PVP-архитектуры могут являться машинами общего назначения (*general purpose systems*). Однако, поскольку векторные процессоры весьма дороги, эти машины не являются общедоступными.

Принципы программирования на PVP-системах предусматривают векторизацию циклов (для достижения оптимальной производительности одного процессора) и их распараллеливание (для одновременной загрузки нескольких процессоров одним приложением).

За счет большой физической памяти (доли терабайта), даже плохо векторизуемые задачи на PVP-системах решаются быстрее, чем на системах со скалярными процессорами.

### 3.5. Кластерная архитектура

Кластер представляет собой два или более компьютеров (часто называемых узлами), которые объединяются с помощью сетевых



технологий на базе шинной архитектуры или коммутатора и предоставляются пользователю в качестве единого информационно-вычислительного ресурса. В качестве узлов кластера могут выступать серверы, рабочие станции или обычные персональные компьютеры. Преимущество кластеризации для повышения работоспособности становится очевидным в случае сбоя какого-либо узла, при этом другой узел кластера может взять на себя нагрузку неисправного узла, и пользователи не заметят прерывания в доступе. Возможности масштабируемости кластеров позволяют многократно увеличивать производительность приложений для большого числа пользователей технологий (Fast/Gigabit Ethernet, Myrinet) на базе шинной архитектуры или коммутатора. Такие суперкомпьютерные системы являются самыми дешевыми, поскольку собираются на базе стандартных комплектующих элементов, процессоров, коммутаторов, дисководов и внешних устройств.

Условное деление кластеров на классы предложено Я. Радаевским и Д. Эдлайном:

- *первый класс* - машина строится целиком из стандартных деталей, которые продают многие продавцы компьютерных компонент (низкие цены, простое обслуживание, аппаратные компоненты доступны из различных источников);

- *второй класс* - система включает эксклюзивные или не широко распространенные детали, т.е. можно достичь очень хорошей производительности, однако при более высокой стоимости.

Кластеры могут существовать в различных конфигурациях. Наиболее употребляемыми типами кластеров являются: системы высокой надежности; системы для высокопроизводительных вычислений; многопоточные системы.

Кластеры для высокопроизводительных вычислений предназначены для параллельных расчетов. Эти кластеры обычно собраны из большого

числа компьютеров. Разработка таких кластеров является сложным процессом, реализовать подобную схему удается далеко не всегда, и она обычно применяется лишь для не слишком больших систем.

Многопоточные системы используются для обеспечения единого интерфейса к различным ресурсам, которые могут со временем произвольно наращиваться (или сокращаться) в размере. Наиболее общий пример этого представляет собой группа Web-серверов.

Архитектура кластерной системы (способ соединения процессоров друг с другом) определяет ее производительность в большей степени, чем тип используемых в ней процессоров. Критическим параметром, влияющим на величину производительности такой системы, является расстояние между процессорами. Так, соединив вместе 10 персональных компьютеров, можно получить систему для проведения высокопроизводительных вычислений. Проблема, однако, будет состоять в нахождении наиболее эффективного способа соединения стандартных средств друг с другом, поскольку при увеличении производительности каждого процессора в 10 раз производительность системы в целом в 10 раз не увеличится.

Наиболее естественным представляется соединение в виде плоской решетки, где внешние концы используются для подсоединения внешних устройств (рис.3.1).



Рис.3.1. Схема соединения процессоров в виде плоской решетки



Теория же показывает [2], что если в системе максимальное расстояние между процессорами больше 4, то такая система не может работать эффективно.

Наиболее эффективной считается архитектура с топологией «толстого дерева» (fat-tree). Архитектура «fat-tree» (hypertree) предложена Лейзерсоном (Charles E. Leiserson) в 1985 г. Процессоры локализованы в листьях дерева, в то время как внутренние узлы дерева скомпонованы во внутреннюю сеть (рис.3.2). Поддеревья могут общаться между собой, не затрагивая более высоких уровней сети.

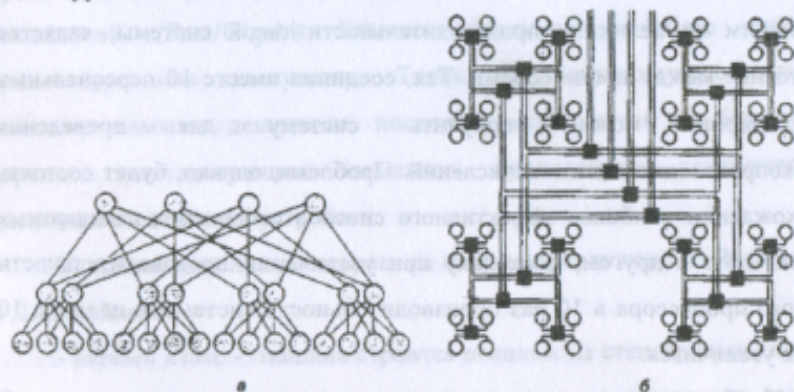


Рис.3.2.Кластерная архитектура «Fat-tree»:

а - вид «сбоку»; б - вид сверху»

В кластерах, как правило, используются операционные системы, стандартные для рабочих станций, чаще всего, свободно распространяемые — Linux, FreeBSD, вместе со специальными средствами поддержки параллельного программирования и балансировки нагрузки. При работе с кластерами так же, как и с MPP системами, используют так называемую Massive Passing Programming Paradigm — парадигму программирования с передачей данных. Дешевизна подобных систем оборачивается большими накладными расходами на взаимодействие параллельных процессов между собой, что сильно сужает потенциальный класс решаемых задач.

### 3.6 Перспективные типы процессоров ЭВМ

#### 3.6.1 Ассоциативные процессоры

Традиционная адресная обработка данных, где последние должны быть представлены в виде ограниченного количества форматов (например, массивы, списки, записи), обуславливает громоздкость операционных систем и систем программирования, а также служит препятствием к созданию вычислительных средств с архитектурой, ориентированной на более эффективное использование параллелизма обработки данных [2].

Ассоциативный способ обработки данных позволяет преодолеть многие ограничения, присущие адресному доступу к памяти, за счет задания некоторого критерия отбора и проведение требуемых преобразований только над теми данными, которые удовлетворяют этому критерию. Критерием отбора может быть совпадение с любым элементом данных, достаточным для выделения искомым данных из всех данных. Поиск данных может происходить по фрагменту, имеющему большую или меньшую корреляцию с заданным элементом данных.

Если реализуется только ассоциативная выборка с последующим поочередным использованием найденных данных, то говорят об ассоциативной памяти или памяти, адресуемой по содержимому. При достаточно полной реализации всех свойств ассоциативной обработки используется термин *ассоциативный процессор*.

#### 3.6.2 Клеточные и ДНК- процессоры

В настоящее время в поисках реальной альтернативы полупроводниковым технологиям создания новых вычислительных систем, ученые обращают все большее внимание на биотехнологии, или *биокомпьютинг*, который представляет собой гибрид информационных и молекулярных технологий. Биокомпьютинг позволяет решать сложные вычислительные задачи, пользуясь методами, принятыми в биохимии и молекулярной биологии, организуя вычисления с помощью живых тканей, клеток, вирусов и биомолекул. Наибольшее распространение получил подход, где в



качестве основного элемента (процессора) используются молекулы дезоксирибонуклеиновой кислоты.

Центральное место в этом подходе занимает так называемый ДНК-процессор. Кроме ДНК в качестве биопроцессора могут быть использованы также белковые молекулы и биологические мембраны ДНК-процессоры, характеризующиеся структурой и набором команд. В данном случае структура процессора — это структура молекулы ДНК, а набор команд — это перечень биохимических операций над молекулами. Принцип устройства компьютерной ДНК-памяти основан на последовательном соединении четырех нуклеотидов (основных кирпичиков ДНК-цепи). Три нуклеотида, соединяясь в любой последовательности, образуют элементарную ячейку памяти — *кодон*. Кодоны затем формируют цепь ДНК. Основная трудность в разработке ДНК-компьютеров связана с проведением избирательных *однокодонных* реакций (взаимодействий) внутри цепи ДНК. Существует экспериментальное оборудование, позволяющее работать с одним из 1020 кодонов или молекул ДНК. Другой проблемой является *самосборка* ДНК, приводящая к потере информации. Ее преодолевают введением в клетку специальных веществ, предотвращающих химическую реакцию самосборки.

Первую модель биокомпьютера (в виде механизма из пластмассы) в 1999 г. создал И. Шапиро из Вейцмановского института естественных наук. Она имитировала работу «молекулярной машины» в живой клетке, собирающей белковые молекулы по информации с ДНК, используя РНК в качестве посредника между ДНК и белком.

В 2001 г. Шапиро удалось реализовать вычислительное устройство на основе ДНК, которое может работать почти без вмешательства человека. Система имитирует машину Тьюринга — одну из фундаментальных абстракций вычислительной техники, которая теоретически может решить любую вычислительную задачу. По своей природе молекулы ДНК работают

аналогичным образом, распадаясь и рекомбинируя в соответствии с информацией, закодированной в цепочках химических соединений. Разработанная установка кодирует входные данные и программы в состоящих из двух цепей молекулах ДНК и смешивает их с двумя ферментами.

В конце февраля 2002 г. появилось сообщение, что фирма Olympus Optical претендует на первенство в создании коммерческой версии ДНК-компьютера, предназначенного для генетического анализа. Машина была создана в сотрудничестве с доцентом Токийского университета А. Тояма. Компьютер, построенный Olympus Optical, имеет молекулярную и электронную составляющие. Первая из них осуществляет химические реакции между молекулами ДНК, обеспечивает поиск и выделение результата вычислений, вторая — обрабатывает информацию и анализирует полученные результаты.

#### 3.6.4. Клеточные компьютеры

Клеточные компьютеры представляют собой самоорганизующиеся колонии различных «умных» микроорганизмов, в геном которых удалось включить некую логическую схему, которая могла бы активизироваться в присутствии определенного вещества. Для этой цели идеально подошли бы бактерии, стакан с которыми и представлял бы собой компьютер. Главным свойством компьютера такого рода является то, что каждая их клетка представляет собой миниатюрную химическую лабораторию. Если биоорганизм запрограммирован, то он просто производит нужные вещества. Достаточно вырастить одну клетку, обладающую заданными качествами, и можно легко и быстро вырастить тысячи клеток с такой же программой.

Основная проблема, с которой сталкиваются создатели клеточных биокомпьютеров, — организация всех клеток в единую работающую систему. На сегодняшний день практические достижения в области клеточных компьютеров напоминают достижения 20-х гг. в области



ламповых и полупроводниковых компьютеров. Сейчас в Лаборатории искусственного интеллекта Массачусетского технологического университета создана клетка, способная хранить на генетическом уровне один бит информации. Также разрабатываются технологии, позволяющие единичной бактерии отыскивать своих соседей, образовывать с ними упорядоченную структуру и осуществлять массив параллельных операций.

К достоинствам, выгодно отличающим биокомпьютеры от компьютеров, основанных на кремниевых технологиях, относятся:

- более простая технология изготовления, не требующая для своей реализации столь жестких условий, как при производстве полупроводников;
- использование не бинарного, а тернарного кода (информация кодируется тройками нуклеотидов), что позволит при меньшем количестве шагов перебрать большее число вариантов при анализе сложных систем;
- потенциально исключительно высокая производительность, которая может составлять до  $10^{14}$  операций в секунду за счет одновременного вступления в реакцию триллионов молекул ДНК;
- возможность хранить данные с плотностью, во много раз превышающей показатели оптических дисков;
- исключительно низкое энергопотребление.

Однако, наряду с очевидными достоинствами, биокомпьютеры имеют и существенные недостатки, такие как: сложность со считыванием результатов, низкая точность вычислений, связанная с возникновением мутаций, прилипанием молекул к стенкам сосудов и т. д.; невозможность длительного хранения результатов вычислений в связи с распадом ДНК в течение времени.

### 3.6.5. Коммуникационные процессоры

Коммуникационные процессоры — это микрочипы, являющие собой

нечто среднее между жесткими специализированными интегральными микросхемами и гибкими процессорами общего назначения. Идея создания процессоров, предназначенных для оптимизации сетевой работы, и при этом достаточно универсальных для программной модификации, родилась в связи с необходимостью устранить различия в подходах к созданию локальных сетей (различные подходы к архитектуре сети, классификации потоков и т. д.). Коммуникационные процессоры программируются, как и привычные ПК-процессоры, но построены с учетом сетевых задач, оптимизированы для сетевой работы, и на их основе производители (как процессоров, так и другого оборудования) создают программное обеспечение для специфических приложений. Коммуникационный процессор имеет собственную память и оснащен высокоскоростными внешними каналами для соединения с другими процессорными узлами. Его присутствие позволяет в значительной мере освободить вычислительный процессор от нагрузки, связанной с передачей сообщений между процессорными узлами. Скоростной коммуникационный процессор с RISC-ядром позволяет управлять обменом данными по нескольким независимым каналам, поддерживать практически все распространенные протоколы обмена, гибко и эффективно распределять и обрабатывать последовательные потоки данных с временным разделением каналов.

### 3.6.6. Нейронные процессоры

Одно из наиболее перспективных направлений разработки принципиально новых архитектур вычислительных систем тесно связано с созданием компьютеров нового поколения на основе принципов обработки информации, заложенных в искусственных нейронных сетях (НС).

Первые практические работы по искусственным нейросетям и нейрокомпьютерам начались еще в 40—50-е гг. Под *нейронной сетью* обычно понимают совокупность элементарных преобразователей информации,



называемых *нейронами*, которые определенным образом соединены друг с другом каналами обмена информации *синоптическими связями*.

Одним из основных достоинств нейровычислителя является то, что его основу составляют относительно простые, чаще всего однотипные элементы, имитирующие работу нейронов мозга. Каждый нейрон характеризуется своим текущим состоянием по аналогии с нервными клетками головного мозга, которые могут быть возбуждены или заторможены. Он обладает группой *синапсов* — однонаправленных входных связей, соединенных с выходами других нейронов, а также имеет *аксон* — выходную связь данного нейрона, с которой сигнал (возбуждения или торможения) поступает на синапсы следующих нейронов. Общий вид нейрона приведен на рис.3.3, а, общий вид нейросети, б.

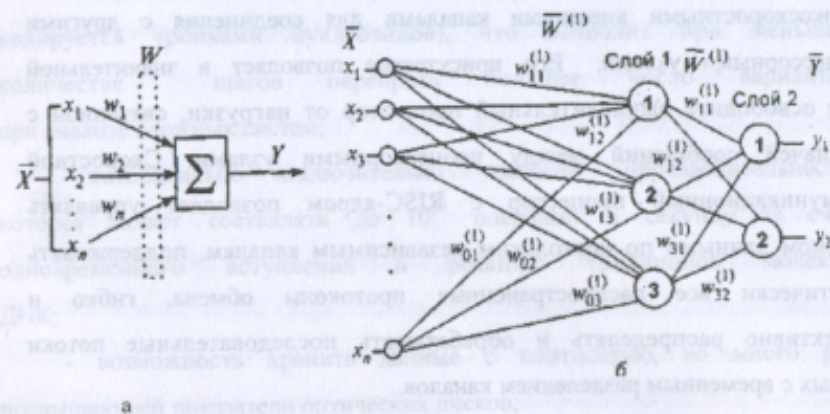


Рис.3.3. Нейрон и нейросеть

Каждый синапс характеризуется *величиной синоптической связи* или ее весом  $w_i$ , который по физическому смыслу эквивалентен электрической проводимости. Текущее состояние нейрона определяется как взвешенная сумма его входных сигналов:

$$s = \sum_{i=1}^n x_i w_i$$

Выход нейрона есть функция его состояния:  $y=f(s)$ , которая называется *активационной*. Состояния нейронов изменяются в процессе

функционирования и составляют кратковременную память нейросети. Каждый нейрон вычисляет взвешенную сумму пришедших к нему по синапсам сигналов и производит над ней нелинейное преобразование.

В отличие от традиционных вычислительных систем нейросетевые вычислители, аналогично нейронным сетям, дают возможность с большей скоростью обрабатывать информационные потоки дискретных и непрерывных сигналов, содержат простые вычислительные элементы и с высокой степенью надежности позволяют решать информационные задачи обработки данных, обеспечивая при этом режим самоперестройки вычислительной среды в зависимости от полученных решений.

Вообще говоря, под термином *нейрокомпьютер* подразумевается довольно широкий класс вычислителей. Это происходит по той причине, что формально нейрокомпьютером можно считать любую аппаратную реализацию нейросетевого алгоритма от простой модели биологического нейрона до системы распознавания символов или движущихся целей. Нейрокомпьютеры не являются компьютерами в общепринятом смысле этого слова. В настоящее время технология еще не достигла того уровня развития, при котором можно было бы говорить о нейрокомпьютере общего назначения.

### 3.6.7. Процессоры с многозначной (нечеткой логикой)

Идея построения процессоров с *нечеткой логикой* (fuzzy logic) основывается на *нечеткой математике*. В отличие от традиционной формальной логики, известной со времен Аристотеля и оперирующей точными и четкими понятиями типа «истина» и «ложь», «да» и «нет», «0» и «1», нечеткая логика имеет дело со значениями, лежащими в некотором (непрерывном или дискретном) диапазоне (рис.3.4). *Функция принадлежности* элементов к заданному множеству также представляет собой не жесткий порог «принадлежит — не принадлежит», а линию, проходящую все значения от нуля до единицы.



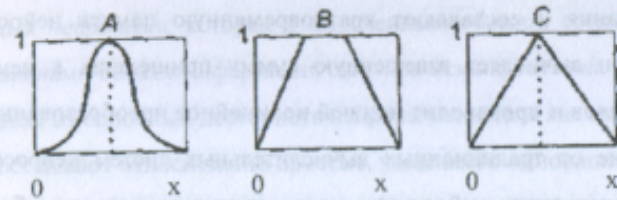


Рис.3.4. Различные типы функций принадлежности

Задачи с помощью нечеткой логики решаются по следующему принципу (рис.3.5):

- 1) численные данные (показания измерительных приборов, результаты анкетирования) *фаззируются* (от англ. *fuzzy*) (переводятся в нечеткий формат);
- 2) обрабатываются по определенным правилам;
- 3) *дефаззируются* и в виде привычной информации подаются на выход.

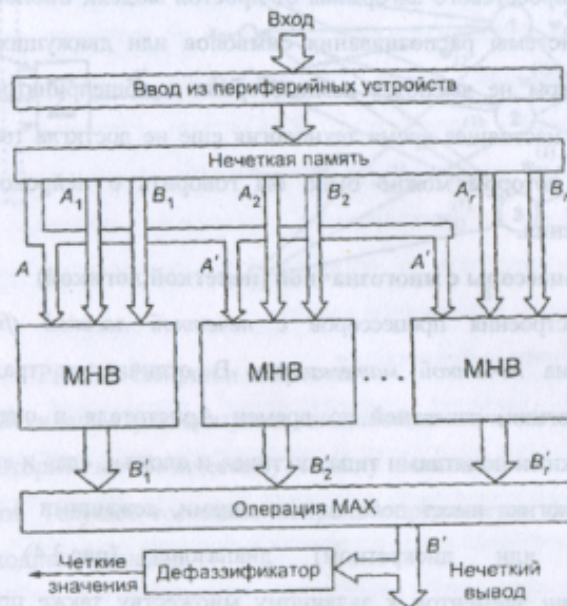


Рис.3.5. Архитектура нечеткого компьютера

В 1986 г. в AT&T Bell Labs создавались процессоры с «прошитой» нечеткой логикой обработки информации. В начале 90-х компания Adaptive

Logic из США выпустила кристалл, сделанный по аналого-цифровой технологии. Он позволит сократить сроки конструирования многих встроенных систем управления реального времени, заменив собой традиционные схемы нечетких микроконтроллеров. Аппаратный процессор нечеткой логики второго поколения принимает аналоговые сигналы, переводит их в нечеткий формат, затем, применяя соответствующие правила, преобразует результаты в формат обычной логики и далее - в аналоговый сигнал. Все это осуществляется без внешних запоминающих устройств, преобразователей и какого бы ни было программного обеспечения нечеткой логики.

Нечеткая логика не решит всех тех задач, которые не решаются на основе логики двоичной, но во многих случаях она удобнее, производительнее и дешевле. Разработанные на ее основе специализированные аппаратные решения (*fuzzy-вычислители*) позволят получить реальные преимущества в быстродействии. Если каскадировать *fuzzy-вычислители*, получается один из вариантов нейропроцессора или нейронной сети. Во многих случаях эти понятия объединяют, называя общим термином «*neuro-fuzzy logic*».

#### Контрольные вопросы

- 1) Что понимается под архитектурой ЭВМ?
- 2) Каким целям служит классификация архитектур?
- 3) Что лежит в основе классификации М. Флинна?
- 4) Перечислите компоненты архитектуры по классификации Скилликорна.
- 5) Что является основным признаком PVP- систем?
- 6) Что понимается под кластером, и в чем преимущества кластеризации?
- 7) Как влияет способ соединения процессоров друг с другом на производительность кластера? Приведите примеры.
- 8) В чем различия традиционного и ассоциативного способа обработки данных?
- 9) Перечислите перспективные типы процессоров.



## Литература

1. Хомоненко А. Д. Основы современных компьютерных технологий: Учебник. – М.: Корона Принт, 2009.
2. Максимов Н.В., Партыка Т.Л., Попов И.И. Архитектура ЭВМ и вычислительных систем: Учебник. – М.: ФОРУМ: ИНФРА-М, 2005.
3. Информатика. Базовый курс/Под ред. С.В. Симоновича. – СПб.: Питер, 2004.
4. Воройский Ф.С. Информатика: Новый систематизированный толковый словарь-справочник (вводный курс по информатике и вычислительной техники в терминах)-2-ое изд. перераб. и доп. – М.: Либерия, 2001.
5. Валда Хиллей. Секреты Windows NT Server 4.0 -Киев, Диалектика 1997.
6. Советов Б.Я. Информационная технология: Учеб. для вузов по спец. Автоматизированные системы обработки информации и управления. – М.: Высшая школа, 1994.
7. Решетников В.Н., Сотников А.Н. Информатика - что это?.. М.: Радио и связь, 1989.
8. Линькова В.П. Информационное и информационно-логическое моделирование в курсе информатики: Монография. – М.: Изд-во ИОСО РАО, 1999.
9. Computing & Multimedia. Словарь. — М.: Внешсигма, 1996.

## ПРИЛОЖЕНИЕ

## Контрольные тесты

- 1) *Возможность достижения поставленной цели с использованием полученной информации* - это
  1. Прагматический аспект информации
  2. Семантический аспект информации
  3. Синтаксический аспект информации
- 2) *Способ представления информации* - это
  1. Прагматический аспект информации
  2. Семантический аспект информации
  3. Синтаксический аспект информации
- 3) *Оценка смысла передаваемой информации* - это
  1. Прагматический аспект информации
  2. Семантический аспект информации
  3. Синтаксический аспект информации
- 4) *Вся информация, подлежащая передаче* - это
 

1. Тезариус	2. Информационный ресурс
3. Сообщение	4. Информация
- 5) *Концентрация имеющихся факторов данных и знаний по направлениям науки и техники* - это
 

1. Тезариус	2. Информационный ресурс
3. Сообщение	4. Информация
- 6) *Словарь, отражающий связь между словами или другими смысловыми элементами языка* - это
 

1. Тезариус	2. Информационный ресурс
3. Сообщение	4. Информация



7) Разъяснение, изложение, осведомленность - это

- |              |                          |
|--------------|--------------------------|
| 1. Тезариус  | 2. Информационный ресурс |
| 3. Сообщение | 4. Информация            |

8) Наука о создании ЭВМ и программном обеспечении, которое можно использовать в процессах, аналогичных с мыслительным процессом человека - это

- |                                 |                      |
|---------------------------------|----------------------|
| 1. Искусственный интеллект      | 2. Информатика       |
| 3. Теория информационных систем | 4. Теория информации |

9) Процессами энергоинформационного обмена в природе занимается наука

- |                  |                |
|------------------|----------------|
| 1. Информатика   | 2. Эоника      |
| 3. Робототехника | 4. Кибернетика |

10) Совокупность взаимосвязанных, хранящихся вместе данных, которые можно обновлять, дополнять, модифицировать другими программами с помощью специальных средств, называется:

- |                 |         |
|-----------------|---------|
| 1. Базой данных | 2. СУБД |
| 3. Информацией  |         |

11) Установите правильную последовательность этапов преобразования информации

- |  |                         |
|--|-------------------------|
| 1. Сбор, подготовка, регистрация, преобразование информации, пригодной для ввода в ЭВМ | 3. Обработка информации |
| 2. Передача и ввод в ЭВМ   |                         |
| 4. Вывод и отображение полученных результатов  |                         |

12) Теория и методология создания и использования технических систем сбора, хранения, обработки и передачи информации, ориентированных на использование средств электронной вычислительной техники и связи -

- |                            |                              |
|----------------------------|------------------------------|
| 1. Техническая информатика | 2. Теоретическая информатика |
| 3. Прикладная информатика  |                              |

13) Теория информации, методы информационного моделирования, теория информационных систем, теория систем искусственного интеллекта, компьютерная лингвистика -

- |                            |                              |
|----------------------------|------------------------------|
| 1. Техническая информатика | 2. Теоретическая информатика |
| 3. Прикладная информатика  |                              |

14) Теория и методология создания и использования информационных технологий, комплексов и систем в различных сферах социальной практики -

- |                            |                              |
|----------------------------|------------------------------|
| 1. Техническая информатика | 2. Теоретическая информатика |
| 3. Прикладная информатика  |                              |

15) Укажите неверное утверждение:

1. Под архивацией понимают такое перекодирование данных, которое позволяет уменьшить по сравнению с исходным объемом памяти необходимый для их хранения объем.
2. Архивация используется для уменьшения размера архива при длительном хранении и ускорении передачи данных по сети.
3. Архивация - это специальная программа, способная преобразовывать файл, уменьшая при этом его размер в байтах.
4. Алгоритм Хаффмана преобразует первоначальную информацию по определенной схеме к виду когда, например, для текстовых данных хранятся не все символы последовательно, так как они записаны в тексте, а запоминается порядок следования в тексте и частота повторений каждого символа.

16) Укажите неверное утверждение: операционная система -

1. Набор специальных программ, обеспечивающих работоспособность компьютерной системы, управление аппаратурой и прикладными программами, интерфейс с пользователем.
2. Совокупность системных программ, записанных в память компьютера в процессе загрузки ОС, которые осуществляют общее



планирование ресурсами и управление всеми действиями компьютера.

3. Комплекс программ, предназначенный для обеспечения определенного уровня эффективности вычислительной системы за счет автоматизированного управления ее работой и предоставляемого пользователям услуг.

4. Программы, записанные в постоянную память компьютера и составляющие базовую систему ввода/вывода.

17) Основные функции операционной системы:

1. Управление аппаратурой, управление процессами ввода/вывода и передачи данных на внешние устройства, поддержка файловой системы, управление работой прикладных программ, администрирование, поддержка сетевого взаимодействия.

2. Управление аппаратурой, управление процессами ввода/вывода и передачи данных на внешние устройства, управление работой прикладных программ, защита информации, поддержка системы паролей в многопользовательских системах, учет использования ресурсов вычислительной системы, обеспечение выполнения всех операций ввода/вывода, соответствующих специфическим особенностям работы каждого из периферийных устройств данного компьютера.

3. Управление процессами ввода/вывода и передачи данных на внешние устройства, поддержка файловой системы, управление работой прикладных программ, администрирование, поддержка сетевого взаимодействия, функциональная избыточность.

18) Принцип распространения ПО, включающий следующие условия:

\* потребитель может использовать программу в любых целях и модифицировать ее, как сочтет нужным;

\* изготовитель не несет никакой ответственности за правильность работы программы и последствия ее применения;

\* изготовитель может (но не обязан) давать консультации по использованию программы или ее настройки для решения задач пользователя

называется:

- |               |             |
|---------------|-------------|
| 1. Shareware  | 3. Freeware |
| 2. Alpha test | 4. Software |

19) Свойства операционной системы:

1. Надежность, защита, предсказуемость, удобства, эффективность, гибкость, расширяемость, ясность.

2. Надежность, процессорное время, защита, оперативная память, и математическое обеспечение.

3. Управление ресурсами, эффективность, защита, предсказуемость, взаимное влияние процессов

20) Укажите номер неправильного ответа.

Под ресурсами микроЭВМ подразумевается следующее:

1. Процессорное время, оперативная память.

2. Оперативная память, математическое обеспечение.

3. Периферийные устройства, процессорное время.

4. Математическое обеспечение, оперативная память.

5. Расширяемость, процессорное время.

21) Укажите номер неверного утверждения.

Цель управления ресурсами заключается:

1. В эффективном использовании ресурсов пользователем.

2. В освобождении пользователя от бремени задач по оперированию ресурсами.



3. В распределении ресурсов и задач по управлению этими ресурсами.

22) Режим, предусматривающий параллельное, т. е. одновременное выполнение более чем одной программы по разным задачам, но использующих результат одной задачи как исходные данные для другой, называется:

- |                     |                    |
|---------------------|--------------------|
| 1. Многозадачный    | 3. Однопрограммный |
| 2. Многопрограммный | 4. Поточковый      |

23) Укажите номер неверного утверждения:

1. Файловая система FAT (File Allocation Table) получила свое наименование по названию метода организации данных - таблицы распределения файлов.

2. При создании пользователем файла с длинным именем Windows NT создает элемент каталога для этого файла, соответствующий соглашению "8+3" по тем правилам, что и для NTFS, плюс один или несколько вторичных элементов каталога.

3. Файловая система Windows NT FAT функционирует аналогично MS-DOS и Windows.

4. FAT — система с точной записью, с быстрым выполнением преобразованных в последовательность операций записи, не обеспечивает функций защиты данных и автоматического восстановления.

24) Укажите номер неверного утверждения.

1. Файловая система HPFS (High Performance File System) используется для совместимости с приложениями OS/2.

2. В HPFS резервируется пространство под два битовых массива объемом 2Кб для каждого дискового интервала в 16Мб.

3. При форматировании тома под HPFS, первые 18 секторов резервируются для блока начальной загрузки, суперблока и запасного блока. Эти структуры используются для загрузки ОС, поддержки файловой системы и восстановления при возможных ошибках.

4. Особенностью HPFS является использование технологии B-tree - древовидной структуры с корнем и несколькими узлами, содержащей данные, организованные некоторым логическим способом.

5. HPFS имеет особенности, способствующие эффективному управлению большими объемами жесткого диска - HPFS предполагает применение 512-байтовых секторов, HPFS - система с отложенной записью, HPFS поддерживает длинные имена файлов (до 255 символов).

25) Укажите номер неверного утверждения:

1. NTFS обеспечивает сочетание эффективности, надежности и совместимости, невозможное в FAT или HPFS.

2. Она разработана для быстрого выполнения стандартных файловых операций (чтения, записи и поиска), а так же улучшенных операций, например, восстановления файловой системы на очень больших жестких дисках.

3. NTFS, включая возможности безопасности, требуемые для файловых серверов и высококачественных персональных компьютеров в корпоративной среде, поддерживает управление доступом к данным и привилегии владельца, что важно для целостности корпоративных данных.

4. Файловая система поддерживает объектно-ориентированные приложения, обрабатывая все файлы как объекты с атрибутами, определяемыми пользователем и системой.

5. Каждый файл на томе NTFS представлен записью в специальном файле - главной файловой таблице (Master File Table — MFT). Местоположения сегментов данных MFT и зеркального файла MFT записаны в логическом центре диска, дубликат - находится в секторе начальной загрузки.

26) Укажите номер неверного утверждения:

1. Физическая реализация процессов обработки означает представление данных в ЭВМ с помощью машинных кодов и выполнение ряда типовых операций над данными.



2. Целью обработки является решение с помощью ЭВМ вычислительных задач оптимизационного, либо расчетного характера, отображающих функциональные задачи управления в системе.

3. Верхний уровень системы реализует основную задачу ИВ сети, т.е. использование вычислительных ресурсов для обработки информации.

4. Фазы передачи информации реализуются в виде многоуровневой системы, в которой в зависимости от задач обмена и обработки информации, а также от типа используемых средств передачи можно выделить три уровня: демодуляция, декодирование, преобразование полученной информации в соответствующую форму.

### 3. Файловая система Windows NT FAT

Файловая система Windows NT FAT (File Allocation Table) — это система, позволяющая хранить файлы и папки на жестком диске.

4. FAT — система с таблицей размещения файлов (File Allocation Table), позволяющая хранить файлы и папки на жестком диске.

Файловая система Windows NT FAT (File Allocation Table) — это система, позволяющая хранить файлы и папки на жестком диске.

24) Укажите номер правильного утверждения.

1. Файловая система HPFS (High Performance File System) используется для хранения данных на жестком диске.

2. Файловая система HPFS (High Performance File System) используется для хранения данных на жестком диске.

3. При форматировании диска под HPFS, версия 1.0, секторы резервируются для блока начальной загрузки, контрольного блока и заголовка блока. Эти структуры используются для загрузки ОС, поддержки файловой системы и восстановления при аварийных ситуациях.