

Глава 2

ИНФОРМАЦИОННЫЕ ПРОЦЕССЫ

§ 7

Хранение информации

Из курса основной школы вам известно:

Человек хранит информацию в собственной памяти, а также в виде записей на различных внешних (по отношению к человеку) носителях: на камне, папирусе, бумаге, магнитных и оптических носителях и пр. Благодаря таким записям, информация передается не только в пространстве (от человека к человеку), но и во времени — из поколения в поколение.

Рассмотрим способы хранения информации более подробно.

Информация может храниться в различных видах: в виде записанных текстов, рисунков, схем, чертежей, фотографий, звукозаписей, кино- или видеозаписей. В каждом случае применяются свои носители.

Носитель — это материальная среда, используемая для записи и хранения информации.



Практически носителем информации может быть любой материальный объект. Информацию можно сохранять на камне, дереве, стекле, ткани, песке, теле человека и т. д. Здесь мы не станем обсуждать различные исторические и экзотические варианты носителей. Ограничимся современными средствами хранения информации, имеющими массовое применение.

Использование бумажных носителей информации

Носителем, имеющим наиболее массовое употребление, до сих пор остается бумага. Изобретенная во II веке н. э. в Китае бумага служит людям уже 19 столетий.

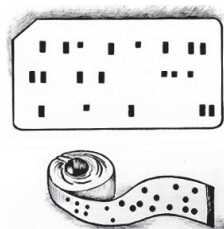
Для сопоставления объемов информации на разных носителях будем пользоваться единицей — байтом, считая, что один

знак текста «весит» 1 байт. Нетрудно подсчитать информационный объем книги, содержащей 300 страниц с размером текста на странице примерно 2000 символов. Текст такой книги имеет объем примерно 600 000 байтов, или 586 Кб. Средняя школьная библиотека, фонд которой составляют 5000 томов, имеет информационный объем приблизительно $2861 \text{ Мб} = 2,8 \text{ Гб}$.



Что касается долговечности хранения документов, книг и прочей бумажной продукции, то она очень сильно зависит от качества бумаги, красителей, используемых при записи текста, условий хранения. Интересно, что до середины XIX века (с этого времени для производства бумаги начали использовать древесину) бумага делалась из хлопка и текстильных отходов — тряпья. Чернилами служили натуральные красители. Качество рукописных документов того времени было довольно высоким, и они могли храниться тысячи лет. С переходом на древесную основу, с распространением машинописи и средств копирования, с началом использования синтетических красителей срок хранения печатных документов снизился до 200–300 лет.

На первых компьютерах бумажные носители использовались для цифрового представления вводимых данных. Это были перфокарты: картонные карточки с отверстиями, хранящие двоичный код вводимой информации. На некоторых типах ЭВМ для тех же целей применялась перфорированная бумажная лента.



Использование магнитных носителей информации

В XIX веке была изобретена магнитная запись. Первоначально она использовалась только для сохранения звука. Самым первым носителем магнитной записи была стальная проволока диаметром до 1 мм. В начале XX столетия для этих целей использовалась также стальная катаная лента. Тогда же (в 1906 г.) был выдан

и первый патент на магнитный диск. Качественные характеристики всех этих носителей были весьма низкими. Достаточно сказать, что для производства 14-часовой магнитной записи устных докладов на Международном конгрессе в Копенгагене в 1908 г. потребовалось 2500 км, или около 100 кг проволоки.

В 20-х годах XX века появляется магнитная лента сначала на бумажной, а позднее — на синтетической (лавсановой) основе, на поверхность которой наносится тонкий слой ферромагнитного порошка. Во второй половине XX века на магнитную ленту научились записывать изображение, появляются видеокамеры, видеомэгнитофоны.

На ЭВМ первого и второго поколений магнитная лента использовалась как единственный вид сменного носителя для устройств внешней памяти. Любая компьютерная информация на любом носителе хранится в двоичном (цифровом) виде. Поэтому независимо от вида информации — текст это, или изображение, или звук — ее объем можно измерить в битах и байтах. На одну катушку с магнитной лентой, использовавшейся в лентопротяжных устройствах первых ЭВМ, помещалось приблизительно 500 Кб информации.

С начала 1960-х годов в употребление входят компьютерные магнитные диски: алюминиевые или пластмассовые диски, покрытые тонким магнитным порошковым слоем толщиной в несколько микрон. Информация на диске располагается по круговым концентрическим дорожкам, на которые она записывается и считывается в процессе вращения диска с помощью магнитных головок.

На первых ПК использовались гибкие магнитные диски (флоппи-диски) — сменные носители информации с небольшим объемом памяти — до 2 Мб. Начиная с 1980-х годов, в ПК начали использоваться встроенные в системный блок **накопители**



на жестких магнитных дисках, или НЖМД (англ. *HDD* — Hard Disk Drive). Их еще называют винчестерскими дисками.

Винчестерский диск представляет собой пакет магнитных дисков, надетых на общую ось, которая при работе компьютера находится в постоянном вращении. С каждой магнитной поверхностью пакета дисков контактирует своя магнитная головка.

Информационная емкость современных винчестерских дисков измеряется в терабайтах.

Оптические диски и флеш-память

Применение **оптического**, или **лазерного**, способа записи информации начинается в 1980-х годах. Его появление связано с изобретением квантового генератора — лазера, источника очень тонкого (толщина порядка микрона) луча высокой энергии. Луч способен выжигать на поверхности плавкого материала двоичный код данных с очень высокой плотностью. Считывание происходит в результате отражения от такой «перфорированной» поверхности лазерного луча с меньшей энергией («холодного» луча). Первоначально на ПК вошли в употребление оптические **компакт-диски** — **CD**, информационная емкость которых составляет от 190 Мб до 700 Мб.

Во второй половине 1990-х годов появились цифровые универсальные видеодиски DVD (Digital Versatile Disk) с большой емкостью, измеряемой в гигабайтах (до 17 Гб). Увеличение их емкости по сравнению с CD связано с использованием лазерного луча меньшего диаметра, а также двухслойной и двусторонней записи. Вспомните пример со школьной библиотекой. Весь ее книжный фонд можно разместить на одном DVD.



В настоящее время оптические диски (CD и DVD) являются наиболее надежными материальными носителями информации, записанной цифровым способом. Эти типы носителей бывают как однократно записываемыми — пригодными только для чтения, так и перезаписываемыми — пригодными для чтения и записи.

В последнее время появилось множество мобильных цифровых устройств: цифровые фото- и видеокамеры, МРЗ-плееры, карманные компьютеры, мобильные телефоны, устройства для чтения электронных книг, GPS-навигаторы и др. Все эти устройства нуждаются в переносных носителях информации. Но поскольку все мобильные устройства довольно миниатюрные, к носителям информации для них предъявляются особые требования. Они должны быть компактными, обладать низким энергопотреблением при работе, быть энергонезависимыми при хранении, иметь большую емкость, высокие скорости записи и чтения, долгий срок службы. Всем этим требованиям удовлетворяют **флеш-карты** памяти. Информационный объем флеш-карты может составлять несколько гигабайтов.

В качестве внешнего носителя для компьютера широкое распространение получили так называемые флеш-брелоки (их называют в просторечии «флешки»), выпуск которых начался в 2001 году. Большой объем информации, компактность, высокая скорость чтения/записи, удобство в использовании — основные достоинства этих устройств. Флеш-брелок подключается к USB-порту компьютера и позволяет скачивать данные со скоростью около 10 Мб в секунду.



В последние годы активно ведутся работы по созданию еще более компактных носителей информации с использованием нанотехнологий, работающих на уровне атомов и молекул вещества. В результате один компакт-диск, изготовленный по нанотехнологии, сможет заменить тысячи оптических дисков. По предположениям экспертов, приблизительно через 20 лет плотность хранения информации возрастет до такой степени, что на носителе объемом примерно с кубический сантиметр можно будет записать каждую секунду человеческой жизни.



Система основных понятий

Хранение информации							
Носители информации							
Нецифровые	Цифровые (компьютерные)						
Исторические: камень, дерево, папирус, пергамент, шелк и др.	Магнитные			Оптические		Флеш-носители	
	Ленты	Диски	Карты	CD	DVD	Флеш-карты	Флеш-брелоки
	Факторы качества носителей						
	Вместимость — плотность хранения данных, объем данных			Надежность хранения — максимальное время сохранности данных, зависимость от условий хранения			
	Наибольшей вместимостью и надежностью на сегодня обладают оптические носители CD и DVD						
	Перспективные виды носителей: носители на базе нанотехнологий						
Современные: бумага							



Вопросы и задания

1. Какая, с вашей точки зрения, сохраняемая информация имеет наибольшее значение для всего человечества, для отдельного человека?
2. Назовите известные вам крупные хранилища информации.
3. Можно ли человека назвать носителем информации?
4. Где и когда появилась бумага?
5. Когда была изобретена магнитная запись? Какими магнитными носителями вы пользуетесь или пользовались?
6. Какое техническое изобретение позволило создать оптические носители информации? Назовите типы оптических носителей.
7. Назовите сравнительные преимущества и недостатки магнитных и оптических носителей.
8. Что означает свойство носителя «только для чтения»?
9. Какими устройствами, в которых используются флеш-карты, вы пользуетесь? Какой у них информационный объем?
10. Какие перспективы, с точки зрения хранения информации, открывают нанотехнологии?

§ 8

Передача информации

Из курса основной школы вам известно:

- Распространение информации происходит в процессе ее передачи.
- Процесс передачи информации протекает от источника к приемнику по информационным каналам связи.

В этом параграфе более подробно будут рассмотрены технические системы передачи информации.

В § 2 уже говорилось о том, что первой в истории технической системой передачи информации стал телеграф. В 1876 году американец Александр Белл изобрел телефон. На основании открытия немецким физиком Генрихом Герцем электромагнитных волн (1886 год), А. С. Попов в России в 1895 году и почти одновременно с ним в 1896 году Г. Маркони в Италии изобрели радио. Телевидение и Интернет появились в XX веке.



Модель передачи информации К. Шеннона

Все перечисленные способы информационной связи основаны на передаче на расстояние физического (электрического или электромагнитного) сигнала и подчиняются некоторым общим законам. Исследованием этих законов занимается **теория связи**, возникшая в 1920-х годах. Математический аппарат теории связи — математическую теорию связи разработал американский ученый Клод Шеннон.

Клодом Шенноном была предложена модель процесса передачи информации по техническим каналам связи, представленная схемой на рис. 2.1.

Работу такой схемы можно пояснить на знакомом всем процессе разговора по телефону. Источником информации является говорящий человек. Кодировующим устройством — микрофон телефонной трубки, с помощью которого звуковые волны (речь) преобразуются в электрические сигналы. Каналом связи служит телефонная сеть (провода, коммутаторы телефонных узлов, через которые проходит сигнал). Декодировующим устройством является телефонная трубка (наушник) слушающего человека — приемни-

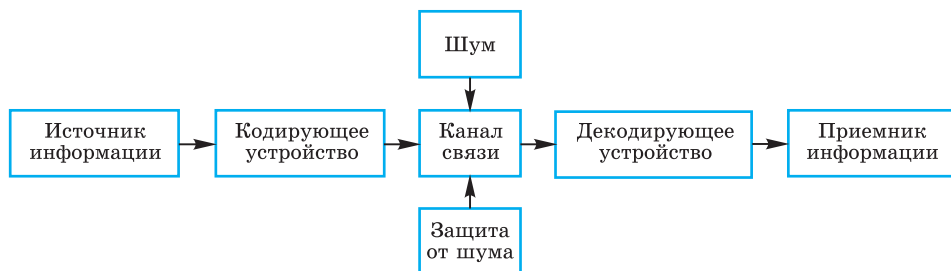


Рис. 2.1. Модель передачи информации по техническим каналам связи

ка информации. Здесь пришедший электрический сигнал превращается в звук.

В § 2 уже говорилось о кодировании на примере передачи информации через письменный документ. Кодирование там было определено как процесс представления информации в виде, удобном для ее хранения и/или передачи.

Применительно к процессу передачи информации по технической системе связи *под кодированием понимается любое преобразование информации, идущей от источника, в форму, пригодную для ее передачи по каналу связи.*

Современные компьютерные системы передачи информации — компьютерные сети, работают по тому же принципу. Есть процесс кодирования, преобразующий двоичный компьютерный код в физический сигнал того типа, который передается по каналу связи. Декодирование заключается в обратном преобразовании передаваемого сигнала в компьютерный код. Например, при использовании телефонных линий в компьютерных сетях функции кодирования/декодирования выполняет прибор, который называется *модемом*.

Пропускная способность канала и скорость передачи информации

Разработчикам технических систем передачи информации приходится решать две взаимосвязанные задачи: как обеспечить наибольшую скорость передачи информации и как уменьшить потери информации при передаче. Клод Шеннон был первым ученым, взявшимся за решение этих задач и создавшим новую для того времени науку — **теорию информации**.



Шеннон определил способ измерения количества информации, передаваемой по каналам связи. Им было введено понятие **пропускной способности канала как максимально возможной скорости передачи информации**. Эта скорость измеряется в битах в секунду (а также килобитах в секунду, мегабитах в секунду).

Пропускная способность канала связи зависит от его технической реализации. Например, в компьютерных сетях используются следующие средства связи:

- телефонные линии;
- электрическая кабельная связь;
- оптоволоконная кабельная связь;
- радиосвязь.

Пропускная способность телефонных линий — десятки и сотни Кбит/с; пропускная способность оптоволоконных линий и линий радиосвязи измеряется десятками и сотнями Мбит/с.

Скорость передачи информации связана не только с пропускной способностью канала связи. Представьте себе, что текст на русском языке, содержащий 1000 знаков, передается с использованием двоичного кодирования. В первом случае используется телеграфная 5-разрядная кодировка. Во втором случае — компьютерная 8-разрядная кодировка. Тогда длина кода сообщения в первом случае составит 5000 битов, во втором случае — 8000 битов. При передаче по одному и тому же каналу второе сообщение будет передаваться дольше в 1,6 раза ($8000/5000$). Отсюда, казалось бы, следует вывод: длину кода сообщения нужно делать минимально возможной.

Однако существует другая проблема, которая на рис. 2.1 отмечена словом «шум».

Шум, защита от шума

Термином «шум» называют разного рода помехи, искажающие передаваемый сигнал и приводящие к потере информации. Подобные помехи, прежде всего, возникают по техническим причинам, таким как плохое качество линий связи, незащищенность друг от друга различных потоков информации, передаваемых по одним и тем же каналам. Существуют и другие источники помех, имеющие физическое происхождение.

Иногда, например, беседуя по телефону, мы слышим шум, треск, мешающие понять собеседника, или на наш разговор накладывается разговор других людей.

Наличие шума приводит к потере передаваемой информации. В таких случаях необходима защита от шума. Для этого в первую очередь применяются технические способы защиты каналов связи от воздействия шумов. Такие способы бывают самыми разными, иногда простыми, иногда очень сложными. Например: использование экранированного кабеля вместо «голого» провода; применение разного рода фильтров, отделяющих полезный сигнал от шума и пр.

Шеннон разработал специальную **теорию кодирования**, дающую методы борьбы с шумом. Одна из важных идей этой теории состоит в том, что передаваемый по линии связи код должен быть *избыточным*. За счет этого потеря какой-то части информации при передаче может быть компенсирована. Например, если при разговоре по телефону вас плохо слышно, то, повторяя каждое слово дважды, вы имеете больше шансов на то, что собеседник поймет вас правильно.

В системах передачи информации используется так называемое **помехоустойчивое кодирование**, вносящее определенную избыточность.

Однако нельзя делать избыточность слишком большой. Это приведет к задержкам и удорожанию связи. Теория кодирования как раз и позволяет получить такой код, который будет оптимальным: избыточность передаваемой информации будет минимально возможной, а достоверность принятой информации — максимальной.

Большой вклад в научную теорию связи внес известный советский ученый Владимир Александрович Котельников. В 1940–1950-х годах им получены фундаментальные научные результаты по проблеме помехоустойчивости систем передачи информации.



**Владимир
Александрович
Котельников**
(1908–2005),
Россия

В современных системах цифровой связи для борьбы с потерей информации при передаче часто применяется следующий прием. Всё сообщение разбивается на порции — блоки. Для каждого блока *вычисляется контрольная сумма* (сумма двоичных цифр), которая передается вместе с данным блоком. В месте приема заново вычисляется контрольная сумма принятого блока и, если она не совпадает с первоначальной суммой, передача данного

блока повторяется. Так происходит до тех пор, пока исходная и конечная контрольные суммы не совпадут.

Система основных понятий

Передача информации в технических системах связи			
Модель К. Шеннона			
Процедура кодирования	Процесс передачи по каналу связи		Процедура декодирования
	Пропускная способность канала	Воздействие шумов на канал связи	
Защита информации от потерь при воздействии шума			
Кодирование с оптимально-избыточным кодом	Частичная потеря избыточной информации при передаче		Полное восстановление исходного сообщения

Вопросы и задания

1. Для чего нужна процедура кодирования передаваемой информации?
2. Что такое декодирование? Каким должен быть его результат?
3. Каким техническим средством связи вы чаще всего пользуетесь? Замечали ли вы при этом факты потери информации?
4. Назовите устройства кодирования и декодирования при использовании радиосвязи.
5. Что такое шум по отношению к системам передачи данных?
6. Какие существуют способы борьбы с шумом?
7. Пропускная способность канала связи 100 Мбит/с. Уровень шума пренебрежимо мал (например, оптоволоконная линия). Определите, за какое время по каналу будет передан текст, информационный объем которого составляет 100 Кб.
8. Пропускная способность канала связи 10 Мбит/с. Канал подвержен воздействию шума, поэтому избыточность кода передачи составляет 20%. Определите, за сколько времени по каналу будет передан текст, информационный объем которого составляет 100 Кб.

§ 9

Обработка информации и алгоритмы

Из курса основной школы вам известно:

Обработка информации, наряду с хранением и передачей, относится к основным видам информационных процессов.

Варианты обработки информации

Обработка информации производится каким-то субъектом или объектом (например, человеком или компьютером) в соответствии с определенными правилами. Будем его называть **исполнителем** обработки информации. Информация, которая подвергается обработке, представляется в виде **исходных данных**. На рисунке 2.2 в обобщенном виде представлен процесс обработки информации.



Рис. 2.2. Модель обработки информации

Можно привести множество примеров, иллюстрирующих схему на рис. 2.2.

Первый пример: ученик (исполнитель), решая задачу по математике, производит обработку информации. Исходные данные содержатся в условии задачи. Математические правила, описанные в учебнике, определяют последовательность вычислений. Результат — это полученный ответ.



Второй пример: перевод текста с одного языка на другой — это пример обработки информации, при которой не меняется ее содержание, но изменяется форма представления — другой язык. Перевод осуществляет переводчик по определенным правилам, в определенной последовательности.



Третий пример: работник библиотеки составляет картотеку книжного фонда. На каждую книгу заполняется карточка, на которой указываются все параметры книги: автор, название, год издания, объем и пр. Из карточек формируется каталог библиотеки, где эти карточки располагаются в строгом порядке. Например, в алфавитном каталоге карточки располагаются в алфавитном порядке фамилий авторов.

Четвертый пример: в телефонной книге вы ищете телефон нужной вам организации, например плавательного бассейна; или в том же библиотечном каталоге разыскиваете сведения о нужной вам книге. В обоих случаях исходными данными является информационный массив — телефонный справочник или каталог библиотеки, а также критерии поиска — название организации или фамилия автора и название книги.



Приведенные примеры иллюстрируют четыре различных вида обработки информации:

- 1) получение новой информации, новых сведений;
- 2) изменение формы представления информации;
- 3) систематизация, структурирование данных;
- 4) поиск информации.

Все эти виды обработки может выполнять как человек, так и компьютер. В чем состоит принципиальное различие между процессами обработки, выполняемыми человеком и машиной?

Если исполнителем обработки информации является человек, то правила обработки, по которым он действует, не всегда формальны и однозначны. Человек часто действует творчески, неформально. Даже однотипные математические задачи он может

решать разными способами. Работа журналиста, ученого, переводчика и других специалистов — это творческая работа с информацией, которая выполняется ими не по формальным правилам.

Об алгоритмах

Для обозначения формализованных правил, определяющих последовательность шагов обработки информации, в информатике используется понятие алгоритма.



Аль-Хорезми
(780–850 гг. н. э.)

Из курса информатики основной школы вы знаете, что слово «алгоритм» произошло от имени выдающегося математика средневекового Востока Мухаммеда аль-Хорезми, описавшего еще в IX веке правила выполнения вычислений с многозначными десятичными числами. Правила сложения, вычитания, умножения столбиком, деления «уголком», которым вас учили в младших классах, — это алгоритмы аль-Хорезми.

С понятием алгоритма в математике ассоциируется известный способ вычисления наибольшего общего делителя (НОД) двух натуральных чисел, который называют алгоритмом Евклида. В словесной форме его можно описать так:

1. Если числа не равны, то большее из них заменить на разность большего и меньшего из чисел.
2. Если два числа равны, то за НОД принять любое из них, иначе перейти к выполнению пункта 1.

Первоклассник, который не знает, что такое НОД, но умеет сравнивать целые числа и выполнять вычитание, сможет исполнить алгоритм. Действовать при этом он будет формально.

Такой формализованный алгоритм легко запрограммировать для современного компьютера. Мечта создать машину — автоматическое устройство, которое сможет без вмешательства человека производить расчеты, появилась очень давно. Для ее реализации требовались не только технические возможности, но и глубокое понимание сущности алгоритмов обработки информации и разработка формализованного способа представления таких алгоритмов.

Алгоритмические машины и свойства алгоритмов

В 30-х годах XX века возникает новая наука — **теория алгоритмов**. Вопрос, на который ищет ответ эта наука: для всякой ли задачи обработки информации может быть построен алгоритм решения? Но чтобы ответить на этот вопрос, надо сначала договориться об исполнителе, на которого должен быть ориентирован алгоритм.

Английский ученый Алан Тьюринг предложил модель такого исполнителя, получившую название «машина Тьюринга». По замыслу Тьюринга, его «машина» является универсальным исполнителем обработки любых символьных последовательностей в любом алфавите. Практически одновременно с Тьюрингом (1936–1937 гг.) другую модель алгоритмической машины описал Эмиль Поста. Машина Поста работает с двоичным алфавитом и несколько проще в своем «устройстве». Можно сказать, что машина Поста является частным случаем машины Тьюринга. Однако именно работа с двоичным алфавитом представляет наибольший интерес, поскольку, как вы знаете, современный компьютер тоже работает с двоичным алфавитом. Подробнее с машиной Поста вы познакомитесь в следующем параграфе.

На основании моделей Тьюринга, Поста и некоторых других ученые пришли к выводу о существовании алгоритмически неразрешимых задач.

Язык программирования алгоритмических машин представляет собой описание конечного числа простых команд, которые могут быть реализованы в автоматическом устройстве.

Совокупность всех команд языка исполнителя называется системой команд исполнителя алгоритмов — СКИ.

Алгоритм управления работой алгоритмической машины представляет собой конечную последовательность команд, посредством выполнения которой машина решает задачу обработки информации.

Алгоритм управления такой машиной должен обладать следующими свойствами:

- дискретностью (каждый шаг алгоритма выполняется отдельно от других);
- понятностью (в алгоритме используются только команды из СКИ);



Алан Тьюринг
(1912–1954),
Англия



- точностью (каждая команда определяет однозначное действие исполнителя);
- конечностью (за конечное число шагов алгоритма получается искомый результат).

Отметим разницу между понятиями «команда алгоритма» и «шаг алгоритма». Команда — это отдельная инструкция в описании алгоритма, а шаг алгоритма — это отдельное действие, которое исполнитель выполняет по команде. В циклических алгоритмах число шагов при выполнении алгоритма может быть больше, чем число команд в алгоритме, за счет повторного выполнения одних и тех же команд.



Система основных понятий

Обработка информации			
Виды обработки информации			
Получение новой информации (новых данных)	Изменение формы представления информации	Структурирование данных	Поиск данных
Исполнитель обработки			
Человек		Автомат (машина)	
Алгоритм обработки — формализованные правила, определяющие последовательность шагов обработки информации			
Алгоритмическая машина — автоматический исполнитель обработки знаковых последовательностей			
Модели алгоритмических машин в теории алгоритмов			
Машина Тьюринга		Машина Поста	
Свойства алгоритма			
Дискретность: каждый шаг алгоритма выполняется отдельно от других	Понятность: в алгоритме используются только команды из СКИ	Точность: каждая команда определяет однозначное действие исполнителя	Конечность: за конечное число шагов алгоритма получается искомый результат

Вопросы и задания

1. Приведите примеры процессов обработки информации, которые чаще всего вам приходится выполнять во время учебы. Для каждого примера определите исходные данные, результаты и правила обработки. К каким видам обработки относятся ваши примеры?
2. Если вы решаете задачу по математике или физике и при этом используете калькулятор, то какова ваша функция в этом процессе и какова функция калькулятора?
3. Используя алгоритм Евклида, найдите НОД для чисел 114 и 66. Сколько шагов алгоритма при этом вам пришлось выполнить?
4. Какие проблемы решает теория алгоритмов?
5. Почему калькулятор нельзя назвать алгоритмической машиной, а компьютер можно?
6. Придумайте минимально необходимую систему команд для кассового аппарата, который подсчитывает стоимость покупок и сумму сдачи покупателю. Опишите алгоритм управления работой такого автомата.

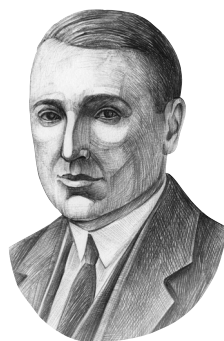
§ 10

Автоматическая обработка информации

В качестве примера автомата, выполняющего обработку информации, рассмотрим машину Э. Поста. Алгоритм, по которому работает машина Поста, будем называть программой.

Договоримся о терминологии: под словом «*программа*» мы всегда будем понимать алгоритм, записанный по строгим правилам языка команд исполнителя — *на языке программирования* для данного исполнителя.

Опишем архитектуру машины Поста (рис. 2.3). Имеется бесконечная информационная лента, разделенная на позиции — клетки. В каждой клетке может либо стоять метка (некоторый знак), либо отсутствовать (пусто).



Эмиль Пост
(1897–1954),
США



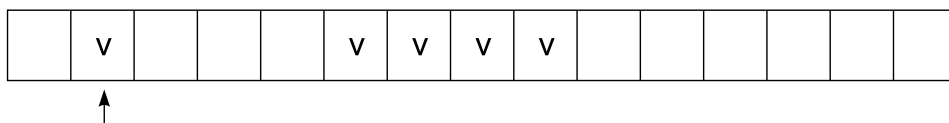


Рис. 2.3. Модель машины Поста

Вдоль ленты движется каретка — считывающее устройство. На рисунке 2.3 она обозначена стрелкой. Каретка может передвигаться шагами: один шаг — смещение на одну клетку вправо или влево. Клетку, под которой установлена каретка, будем называть текущей.

Каретка является еще и процессором машины. С ее помощью машина может:

- распознать, пустая клетка или помеченная знаком;
- стереть знак в текущей клетке;
- записать знак в пустую текущую клетку.

Если произвести замену меток на единицы, а пустых клеток — на нули, то информацию на ленте можно будет рассматривать как аналог двоичного кода телеграфного сообщения или данных в памяти компьютера. Существенное отличие каретки-процессора машины Поста от процессора компьютера состоит в том, что в компьютере возможен доступ процессора к ячейкам памяти в произвольном порядке, а в машине Поста — только последовательно.

Назначение машины Поста — производить преобразования на информационной ленте. Исходное состояние ленты можно рассматривать как исходные данные задачи, конечное состояние ленты — как результат решения задачи. Кроме того, в исходные данные входит информация о начальном положении каретки.

Теперь рассмотрим систему команд машины Поста (табл. 2.1). Запись всякой команды начинается с ее порядкового номера в программе — n . Затем следует код операции и после него — номер следующей выполняемой команды программы — m .

Рассмотрим пример программы решения задачи на машине Поста. Исходное состояние показано на рис. 2.3. Машина должна стереть знак в текущей клетке и присоединить его слева к группе знаков, расположенных справа от каретки. Программа приведена в табл. 2.2.

Таблица 2.1

Система команд машины Поста

Команда	Действие
$n \leftarrow m$	Сдвиг каретки на шаг влево и переход к выполнению команды с номером m
$n \rightarrow m$	Сдвиг каретки на шаг вправо и переход к выполнению команды с номером m
$n \vee m$	Запись метки в текущую пустую клетку и переход к выполнению команды с номером m
$n \updownarrow m$	Стирание метки в текущей клетке и переход к выполнению команды с номером m
$n !$	Остановка выполнения программы
$n ? m, k$	Переход в зависимости от содержимого текущей клетки: если текущая клетка пустая, то следующей будет выполняться команда с номером m , если непустая — команда с номером k

Таблица 2.2

Программа для машины Поста

Команда	Действие
$1 \updownarrow 2$	Стирание метки; переход к следующей команде
$2 \rightarrow 3$	Сдвиг вправо на один шаг
$3 ? 2, 4$	Если клетка пустая, то переход к команде 2, иначе — к команде 4
$4 \leftarrow 5$	Сдвиг влево на шаг (команда выполнится, когда каретка выйдет на первый знак группы)
$5 \vee 6$	Запись метки в пустую клетку
$6 !$	Остановка машины

В процессе выполнения приведенной программы многократно повторяется выполнение команд с номерами 2 и 3. Такая ситуация называется *циклом*. Напомним, что цикл относится к числу основных алгоритмических структур вместе со *следованием* и *ветвлением*.

А теперь научим машину Поста играть в интеллектуальную игру, которая называется «Игра Баше». Опишем правила игры.

Играют двое. Перед ними 21 (или 16, или 11 и т. д.) фишка. Игроки берут фишки по очереди. За один ход можно взять от 1 до 4 фишек. Проигрывает тот, кто забирает последнюю фишку.

Имеется выигрышная тактика для игрока, берущего фишки вторым. Она заключается в том, чтобы брать такое количество фишек, которое дополняет число фишек, взятых соперником на предыдущем ходе, до пяти.

Роль фишек на информационной ленте машины Поста будут выполнять метки (знаки). Машина играет с человеком. Человеку предоставляется возможность стирать метки (брать фишки) первым. Машина будет вступать в игру второй. Исходная обстановка: на ленте массив из 21 клетки содержит метки. Каретка установлена на крайней слева клетке этого массива. Стирать метки можно только подряд. Выигрышным результатом должна быть одна оставшаяся метка перед очередным ходом человека.

Еще раз напомним принцип выигрышной тактики: стирать столько меток, чтобы в сумме с метками, стертыми противником за предыдущий ход, их было пять.

Программа управления машиной Поста в игре Баше против человека приведена в табл. 2.3.

Таблица 2.3

Программа игры Баше

Команда	Действие
1 ? 2,1	Машина ждет появления пустой клетки над кареткой. После очередного хода человека машина делает свой ход. Если человек видит всего одну метку на ленте, он прекращает игру, признав свое поражение
2 → 3	Эта серия команд выведет каретку на пятую (десятую, пятнадцатую и т. д.) позицию. Какой бы ход ни сделал соперник, в ней обязательно будет стоять метка
3 → 4	
4 → 5	
5 → 6	
6 ↑ 7	Стирается метка в текущей клетке
7 ← 8	Шаг влево
8 ? 9,6	Если клетка не пустая, то возврат к команде 6
9 → 10	Каретка перемещается к первой помеченной клетке. После этого машина возвращается к команде 1 и ждет хода человека (или признания им своего поражения)
10 ? 9,1	

Действуя по данной программе и начиная стирать метки второй после человека, машина всегда будет выигрывать, если правильно задано начальное число меток, которое должно быть равно $5n + 1$, где n — любое натуральное число. В противном случае машина может проиграть.

Подведем итог. Автоматическая обработка информации возможна, если:

- 1) информация представлена в формализованном виде — в конечном алфавите некоторой знаковой системы;
- 2) реализован исполнитель, обладающий конечной системой команд, достаточной для построения алгоритмов решения определенного класса задач обработки информации;
- 3) реализовано программное управление работой исполнителя.

Машина Поста — пример автоматического исполнителя обработки информации с ограниченными возможностями. Компьютер удовлетворяет всем вышеперечисленным свойствам. Он является универсальным автоматическим исполнителем обработки информации.

Система основных понятий



Автоматическая обработка информации		
Свойства алгоритмической машины		
Дискретное, символьное представление данных (информации) в памяти машины	Достаточная конечная система команд для решения класса задач	Автоматическое исполнение процессором программы, изменяющей содержимое памяти
Алгоритмическая машина Поста		
Двоичное представление информации на бесконечной ленте памяти	— команды выставления и удаления метки; — команды смещения каретки влево и вправо; — безусловный и условный переходы; — остановка машины	Исполнение начинается с первой команды и заканчивается командой остановки. Процессор последовательно выполняет команды программы



Вопросы и задания

1. На информационной ленте машины Поста расположен массив из N меток. Каретка находится под крайней левой меткой. Какое состояние установится на ленте после выполнения следующей программы?
 $1 \rightarrow 2$
 $2 \updownarrow 3$
 $3 \rightarrow 4$
 $4 ? 5, 2$
 $5 \leftarrow 6$
 $6 \vee 7$
 $7 !$
2. На информационной ленте на некотором расстоянии справа от каретки, стоящей под пустой клеткой, находится непрерывный массив меток. Требуется присоединить к правому концу массива одну метку.
3. На ленте расположен массив из $2n - 1$ меток. Составить программу отыскания средней метки и стирания ее.
4. На ленте расположен массив из $2n$ меток. Составить программу, по которой машина раздвинет на расстояние в одну клетку две половины данного массива.

§ 11

Информационные процессы в компьютере

Из курса основной школы вам известно:

- Компьютер (ЭВМ) — автоматическое, программно-управляемое устройство для работы с информацией.
- В состав компьютера входят устройства памяти (хранение данных и программ), процессор (обработка информации), устройства ввода/вывода (прием/передача информации).
- В 1946 году Джоном фон Нейманом были сформулированы основные принципы устройства ЭВМ, которые называют фон-неймановской архитектурой.
- Современный компьютер представляет собой единство аппаратуры (hardware) и программного обеспечения (software).

Серийное производство электронных вычислительных машин (ЭВМ) начинается в разных странах в 1950-х годах. Историю развития ЭВМ принято делить на поколения. Переход от одного поколения к другому связан со сменой элементной базы, на которой создавались машины, с изменением архитектуры ЭВМ,

с развитием основных технических характеристик (скорости вычислений, объема памяти и др.), с изменением областей применения и способов эксплуатации машин.

Под **архитектурой ЭВМ** понимаются наиболее общие принципы построения компьютера, реализующие программное управление его работой и взаимодействие основных функциональных узлов.

В основе архитектуры ЭВМ разных поколений лежат принципы Джона фон Неймана. Однако в процессе развития происходят некоторые отклонения от фон-неймановской архитектуры.

Однопроцессорная архитектура ЭВМ

Элементной базой ЭВМ **первого поколения** (1950-е годы) были электронные лампы, а ЭВМ **второго поколения** (1960-е годы) создавались на базе полупроводниковых элементов. Однако их архитектура была схожей. Она в наибольшей степени соответствовала принципам фон Неймана. В этих машинах один процессор управлял работой всех устройств: внутренней и внешней памяти, устройств ввода и вывода, как показано на рис. 2.4.

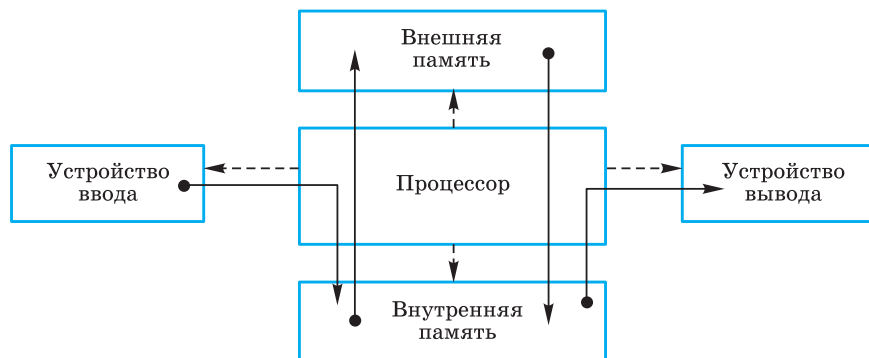


Рис. 2.4. Структура однопроцессорной ЭВМ. Сплошные стрелки — передача данных, пунктирные стрелки — управляющее воздействие

Согласно принципам фон Неймана, исполняемая программа хранится во внутренней памяти — в оперативном запоминающем устройстве (ОЗУ). Там же находятся данные, с которыми работает программа. Каждая команда программы и каждая величина (элемент данных) занимают определенные ячейки памяти, как показано на рис. 2.5.

	Внутренняя память	
	Номер ячейки	Содержимое ячейки
Программа	1	
	2	
	...	
	...	
	N	Команда STOP
Данные	$N + 1$	Величина 1
	$N + 2$	Величина 2
	...	...

Рис. 2.5. Размещение в ОЗУ программы и данных

Процессор начинает выполнение программы с первой команды и заканчивает на команде остановки, назовем ее STOP. При выполнении очередной команды процессор извлекает из памяти обрабатываемые величины и заносит их в специальные ячейки внутренней памяти процессора — регистры. Затем выполняется команда, например складываются два числа, после чего полученный результат записывается в определенную ячейку памяти. Процессор переходит к выполнению следующей команды. Исполнение программы закончится, когда процессор обратится к команде STOP.

Среди команд программы существуют команды обработки данных и команды обращения к внешним устройствам. Команды обработки данных выполняет сам процессор с помощью входящего в него арифметико-логического устройства — АЛУ, и этот процесс происходит сравнительно быстро. А команды управления внешними устройствами выполняются самими этими устройствами: устройствами ввода/вывода, внешней памятью. Время выполнения этих команд во много раз больше, чем время выполнения команд обработки данных. При однопроцессорной архитектуре ЭВМ, показанной на рис. 2.4, процессор, отдав команду внешнему устройству, ожидает завершения ее выполнения. При большом числе обращений к внешним устройствам может оказаться, что большую часть времени выполнения программы процессор «простаивает» и, следовательно, его КПД оказывается низким. Быстродействие ЭВМ с такой архитектурой находилось в пределах 10–20 тысяч операций в секунду (оп./с).

Использование периферийных процессоров

Следующим шагом в развитии архитектуры ЭВМ стал отказ от однопроцессорного устройства. Уже на последних моделях машин второго поколения, помимо *центрального процессора (ЦП)*, выполнявшего обработку данных, присутствовали *периферийные процессоры*, которые назывались каналами ввода/вывода (рис. 2.6). Их задача состояла в автономном управлении устройствами ввода/вывода и внешней памяти, что освобождало от этой работы центральный процессор. В результате КПД центрального процессора существенно возрос. Быстродействие некоторых моделей машин с такой архитектурой составляло от 1 до 3 млн оп./с.

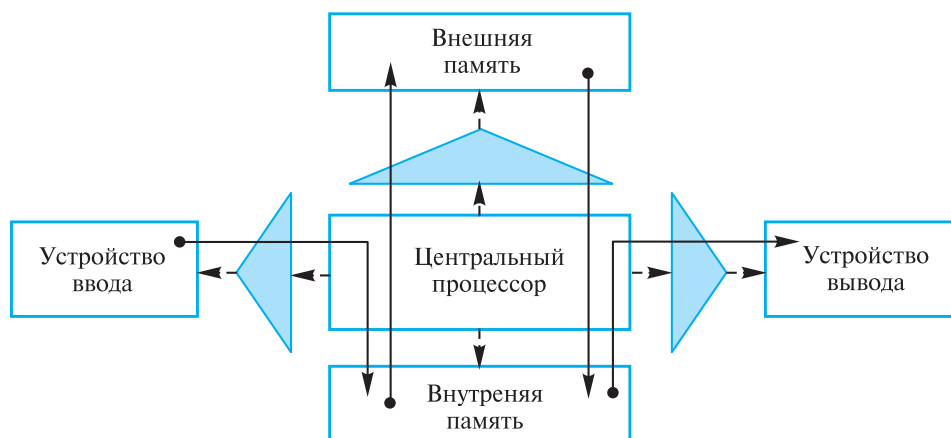


Рис. 2.6. Структура ЭВМ с одним центральным процессором и периферийными процессорами управления внешними устройствами (треугольники)

На всех моделях ЭВМ **третьего поколения**, которые создавались на базе интегральных схем (1970–80-е годы), использовалась архитектура с одним центральным процессором и периферийными процессорами внешних устройств. Такая многопроцессорная архитектура позволяла реализовать *мультипрограммный режим работы*: пока одна программа занята вводом/выводом данных, которым управляет периферийный процессор, другая программа занимает центральный процессор, выполняя вычисления. Благодаря совершенствованию элементной базы и других аппаратных средств на некоторых моделях ЭВМ третьего поколения достигалось быстродействие до 10 млн оп./с.

Для разделения ресурсов ЭВМ между несколькими выполняемыми программами потребовалось создание специального программного обеспечения: **операционной системы (ОС)**. К разделяемым ресурсам, прежде всего, относятся время работы центрального процессора и оперативная память. Задача ОС состоит в том, чтобы разные программы, выполняемые одновременно на ЭВМ, «не мешали» друг другу и чтобы КПД центрального процессора был максимальным, иначе говоря, чтобы ЦП не «простаивал». ОС берет на себя также заботу об очередности использования несколькими программами общих внешних устройств: внешней памяти, устройств ввода/вывода.

Архитектура персонального компьютера

Персональный компьютер (ПК) — самый распространенный в наше время тип компьютера. Появление ПК связано с созданием микропроцессоров, которое началось в 1970-х годах. До недавнего времени в устройстве ПК существовал один центральный процессор и множество периферийных процессоров, управляющих внешними устройствами, которые называются контроллерами. Архитектура такого ПК изображена на рис. 2.7.

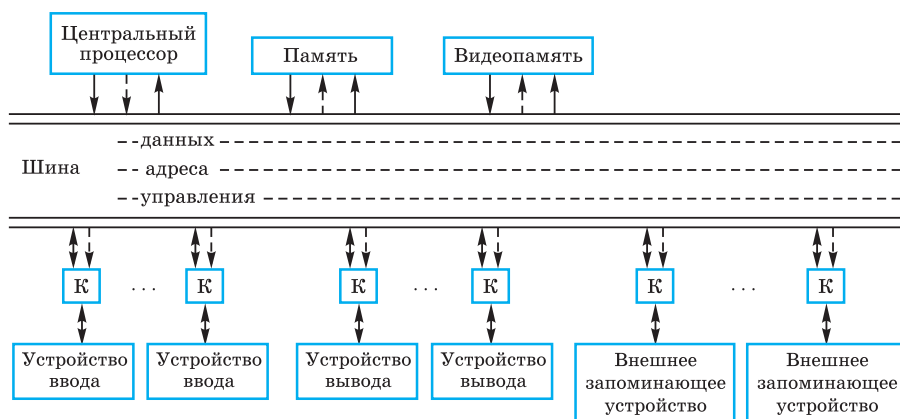


Рис. 2.7. Архитектура персонального компьютера (сплошные стрелки — направление потоков информации, пунктирные — направление управляющих сигналов, К — контроллер)

Для связи между отдельными функциональными узлами ПК используется общая информационная магистраль, которая называется **системной шиной**.

Системная шина состоит из трех частей:

- шина данных (для передачи данных);
- шина адреса (для передачи адресов устройств, которым передаются данные);
- шина управления (для передачи управляющих сигналов, синхронизирующих работу разных устройств).

Важное достоинство такой архитектуры — возможность подключения к компьютеру новых устройств или замена старых устройств на более современные. Это называется **принципом открытой архитектуры**. Для каждого типа и модели устройства используется свой контроллер, а в составе операционной системы имеется управляющая программа, которая называется драйвером устройства.

Открытая архитектура персонального компьютера — это архитектура, предусматривающая модульное построение компьютера с возможностью добавления и замены отдельных устройств.

Важное событие в совершенствовании архитектуры ПК произошло в 2005 году: был создан первый двухъядерный микропроцессор. Каждое ядро способно выполнять функции центрального процессора. Эта особенность архитектуры позволяет производить на ПК параллельную обработку данных, что существенно увеличивает его производительность. Выпускаемые в настоящее время микропроцессоры содержат до 8 ядер.

Архитектура ненејмановских вычислительных систем

Несмотря на стремительно нарастающую производительность ЭВМ, которая каждые 4–5 лет по важнейшим показателям практически удваивается, всегда есть классы задач, для которых никакой производительности не хватает. Укажем некоторые из них.

1. Математические расчеты, лежащие в основе реализации математических моделей многих процессов. Гигантские вычислительные ресурсы, которые можно реализовать очень быстро (как иногда говорят, в *масштабе реального времени*), необходимы для более надежного и долгосрочного прогноза погоды, для решения аэрокосмических задач, в том числе и оборонных, для решения многих инженерных задач и т. д.
2. Поиск информации в гигантских базах данных, в информационном пространстве Интернета.

3. Моделирование интеллекта — при всех фантастических показателях, объем оперативной памяти современных компьютеров составляет лишь малую долю объема памяти человека.

Быстродействие компьютера с одним центральным процессором имеет физическое ограничение: повышение тактовой частоты процессора ведет к повышению тепловыделения, которое не может быть неограниченным. Перспективный путь повышения производительности компьютера лежит на пути отказа от единственности главных устройств компьютера: либо процессора, либо оперативной памяти, либо шины, либо всего этого вместе. Это путь еще большего отступления от архитектуры фон Неймана.

Чтобы стало понятнее, зачем компьютеру несколько процессоров, обсудим алгоритм решения простейшей математической задачи. Есть массив из 100 чисел: $a_1, a_2, \dots, a_{100}$. Требуется найти их сумму.

Нет ничего проще! И на компьютере, и без него мы, скорее всего, поступим так: сложим первые два числа, как-то обозначим их сумму (например, S), затем прибавим к ней третье, и будем делать это еще 98 раз. Это пример последовательного вычислительного процесса. Его блок-схема приведена на рис. 2.8.

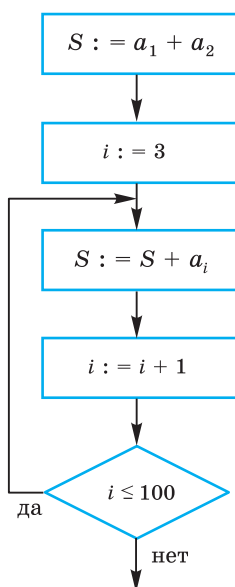


Рис. 2.8. Алгоритм решения задачи последовательного сложения массива чисел

Поскольку у человека нет второй головы, иначе эту задачу в одиночку не решить. Но представим, что мы решаем ее не в одиночку, а всем классом (25 человек). Тогда возникает возможность совсем иной последовательности действий.

1. Объединим числа в пары — по два на каждого (итого распределили 50 чисел); например, ученик № 1 берет себе a_1 и a_2 , ученик № 2 — a_3 и a_4 , и т. д.
2. Даем команду «складывай!» — и каждый складывает свои числа.
3. Даем команду «записывай!» — и каждый записывает мелом на классной доске свой результат.
4. Поскольку у нас осталось еще 50 необработанных чисел ($a_{51}, \dots, a_{100}$), повторяем пункты 1–3. После этого имеем на доске 50 чисел $b_1 = a_1 + a_2, \dots, b_{50} = a_{99} + a_{100}$ — результаты парных сложений.
5. Объединим в пары числа b_i и повторим выполнение пунктов 2–4.

Продолжаем этот процесс (2–5) до тех пор, пока не останется одно число — искомая сумма.

Первое впечатление, что всё это очень сложно, гораздо сложнее, чем алгоритм на рис. 2.8. Если бы мы захотели записать этот алгоритм в виде блок-схемы, то нам пришлось бы кроме описания порядка и объектов действий сделать то, что мы никогда при записи алгоритмов не делали, — предусмотреть синхронизацию параллельных процессов по времени. Например, выполнение команд 2 и 3 должно завершиться всеми участниками вычислений до того, как они будут продолжены (до перехода к п. 4), иначе даже при решении этой простой задачи наступит хаос.

Но сложность не есть объективная причина отвергнуть такой путь, особенно если речь идет о возможности значительного ускорения компьютерных вычислений. То, что мы предложили выше, называется на языке программистов *распараллеливанием вычислений* и вполне поддается формальному описанию. Эффект ускорения вычислений очевиден: пункт 2 в приведенном выше алгоритме ускоряет соответствующий этап работы в 25 раз!

Следующий вопрос: что надо изменить в устройстве компьютера, чтобы он смог так работать? Для реализации подобной схемы вычислений компьютеру потребуется 25 процессоров, объединенных в одну архитектуру и способных работать параллельно. Такие

многопроцессорные вычислительные комплексы — реальность сегодняшней вычислительной техники.

Вернемся, однако, к описанной выше последовательности действий — в ней еще есть источники проблем. Представим себе, что в схеме на рис. 2.7 мы дорисовали еще 24 центральных процессора, соединенных с шиной. При реализации в таком компьютере команды 3 произойдет *одновременное* обращение 25 процессоров к системной шине для пересылки результатов сложения в оперативную память. Но поскольку шина одна, числа по ней могут пересылаться только по одному! Значит, для выполнения команды 3 придется организовать очередь на передачу чисел в память. Тут же возникает вопрос: не сведет ли к нулю эта очередь все преимущества от параллельности выполнения операций на шаге 2? А если преимущества останутся, то насколько они велики? Окупятся ли расходы на 24 дополнительных процессора?

В возникшей ситуации естественен следующий шаг «изобретательской мысли»: ввод в архитектуру нескольких системных шин. А если еще подумать над возможными проблемами, то и нескольких устройств оперативной памяти.

Как видите, всё это очень непросто! Обсуждаемые изменения в устройстве компьютера приводят к «ненеймановским» архитектурам. Изобретателям таких систем приходится искать компромисс между возрастающей сложностью (и, как следствие, — стоимостью) и ускорением их работы.

Варианты реализации ненеймановских вычислительных систем

В самом общем смысле под **параллельными вычислениями** понимаются процессы обработки данных, в которых одновременно могут выполняться нескольких машинных операций. Параллельные вычисления реализуются как за счет новой архитектуры вычислительной техники, так и за счет новых технологий программирования. Такие технологии называются **параллельным программированием**.

Распределенные вычисления — способ реализации параллельных вычислений путем использования множества компьютеров, объединенных в сеть. Такие вычислительные системы еще называют **мультикомпьютерными**.

Распределенные вычисления часто реализуются с помощью **компьютерных кластеров** — нескольких компьютеров, связанных в локальную сеть и объединенных специальным программ-

ным обеспечением, реализующим параллельный вычислительный процесс. Распределенные вычисления могут производиться и с помощью **многомашинных вычислительных комплексов**, образуемых объединением нескольких отдельных компьютеров через глобальные сети.

Мультипроцессорные системы образуют единый компьютер, который относится к классу **суперкомпьютеров**. Достижение параллелизма в них происходит благодаря возможности независимой работы отдельных устройств и их дублирования: несколько процессоров, блоков оперативной памяти, шин и т. д. Мультипроцессорная система может использовать разные способы доступа к общей для всей системы памяти. Если все процессоры имеют равный (*однородный*) доступ к единой памяти, то соответствующая вычислительная система называется *векторным суперкомпьютером*.

Один из самых мощных в мире суперкомпьютеров под названием «Ломоносов» (рис. 2.9) произведен в России и работает в Московском государственном университете. Его быстродействие составляет более ста триллионов операций в секунду.



Рис. 2.9. Суперкомпьютер «Ломоносов» (МГУ им. М. В. Ломоносова)



Система основных понятий

Информационные процессы в компьютере				
Эволюция поколений ЭВМ				
	1 1950-е годы	2 1960-е годы	3 1970-е годы	4 (ПК, суперЭВМ) Начиная с 1970-х
Элемент- ная база	Электрон- ные лампы	Транзисторы	Интеграль- ные схемы (ИС) и боль- шие интег- ральные схе- мы (БИС)	БИС, СБИС (сверхболь- шие ИС), микропро- цессоры
Макси- мальное быстро- действие (оп./с)	10–20 тыс.	100 тыс. — 3 млн	10 млн	10^9 – 10^{12}
Архитек- тура	Фон-нейма- новская однопроцес- сорная	Фон-нейманов- ская однопро- цессорная. Появление периферийных процессоров	Центральный процессор + каналы вво- да/вывода. Шинная ар- хитектура	Конвейерно- векторные, матричные, многопро- цессорные, мульти- компьютер- ные системы
Ненеймановские вычислительные системы Ведущий принцип: отказ от последовательного выполнения операций				
Техническое решение		Программное решение		
Мультипроцессорные системы — супер- компьютеры; мульти- компьютерные системы (кластеры)		Параллельное программирование: выделение в программе нескольких одновременно вы- полняемых действий (распараллеливание)		

Вопросы и задания

1. Расскажите о смене элементной базы компьютеров, происходившей при переходе от одного поколения к другому. Как при этом менялись основные характеристики ЭВМ?
2. В чем состоял отход от архитектуры фон Неймана на ЭВМ второго и третьего поколений?
3. Что позволило реализовать мультипрограммный режим работы на ЭВМ третьего поколения?
4. Чем принципиально отличается архитектура ПК от классической архитектуры компьютеров первых поколений?
5. Какие функции выполняют контроллеры внешних устройств на ПК?
6. В чем состоит принцип открытости архитектуры ПК?
7. Какие функции выполняли первые операционные системы?
8. Для каких классов задач нужны сверхпроизводительные вычислительные системы?
9. Что такое параллельные вычисления?
10. Для примера со сложением чисел 25 учениками попробуйте проанализировать следующие ситуации: в классе всего 1 кусочек мела; в классе 5 кусочков мела; в классе 25 кусочков мела. Оцените, как от этого зависит время решения задачи (учтите еще ширину доски и время перемещения учеников по классу). Попробуйте построить модель такого процесса. Переведите эту ситуацию на язык компьютерной терминологии для многопроцессорных систем.
11. Чем отличаются мультикомпьютерные системы от мультипроцессорных? По какому принципу работают суперкомпьютеры?

ЭОР к главе 2 на сайте ФЦИОР (<http://fcior.edu.ru>)

- Архитектура компьютера
- Архитектура машин пятого поколения
- Внутренняя память компьютера. Внешняя память компьютера. Типы накопителей информации
- Магистраль. Передача данных внутри компьютера
- От абака до ноутбука. Поколения компьютерной техники
- Понятие алгоритма
- Принцип открытой архитектуры
- Принципы и системы передачи информации
- Принципы и системы передачи информации. Вычисление объема информации при передаче. Практическая работа

