

Алгоритмы.

Программный принцип работы ПК

1. Алгоритмы и способы их описания
2. Свойства алгоритмов
3. Способы описания алгоритмов
4. Компьютер как исполнитель команд. Программный принцип работы ПК

1. Алгоритмы и способы их описания

Алгоритм – предписание, однозначно задающее процесс преобразования исходной информации в виде последовательности элементарных дискретных шагов, приводящих за конечное число их применений к результату.

Алгоритмами, например, являются правила сложения, умножения, решения алгебраических уравнений, умножения матриц и т.п. Слово алгоритм происходит от *algoritmi*, являющегося латинской транслитерацией арабского имени хорезмийского математика IX века аль-Хорезми. Благодаря латинскому переводу трактата аль-Хорезми европейцы в XII веке познакомились с позиционной системой счисления, и в средневековой Европе алгоритмом называлась десятичная позиционная система счисления и правила счета в ней.

Алгоритм – это точная инструкция, а инструкции встречаются практически во всех областях человеческой деятельности. Возможны алгоритмы проведения физического эксперимента, сборки шкафа или телевизора, обработки детали. *Однако не всякая инструкция есть алгоритм.*

Инструкция становится алгоритмом только тогда, когда она удовлетворяет определенным требованиям. Эти требования частично сформулированы в определении, хотя упомянутые в определении понятия *однозначности и элементарности* сами нуждаются в уточнении.

Алгоритм однозначен, если при применении к одним и тем же данным он даст один и тот же результат. Но как по описанию алгоритма определить, однозначен он или нет.

Применительно к ЭВМ алгоритм определяет вычислительный процесс, начинающийся с обработки некоторой совокупности возможных исходных данных и направленный на получение определенных этими исходными данными результатов. Термин *вычислительный процесс* распространяется и на обработку других видов информации, например, символьной, графической или звуковой.

2. Свойства алгоритмов

Если вычислительный процесс заканчивается получением результатов, то говорят, что соответствующий алгоритм применим к рассматриваемой совокупности исходных данных. В противном случае говорят, что алгоритм неприменим к совокупности исходных данных. Любой применимый алгоритм обладает следующими основными свойствами:

- дискретностью;
- определенностью;
- результативностью;
- массовостью.

Дискретность – последовательное выполнение простых или ранее определённых (подпрограммы) шагов. Преобразование исходных данных в результат осуществляется дискретно во времени.

Определенность состоит в совпадении получаемых результатов независимо от пользователя и применяемых технических средств (однозначность толкования инструкций).

Результативность означает возможность получения результата после выполнения конечного количества операций.

Массовость заключается в возможности применения алгоритма к целому классу однотипных задач, различающихся конкретными значениями исходных данных (разработка в общем виде).

2. Свойства алгоритмов

Для задания алгоритма необходимо описать следующие его элементы:

- набор объектов, составляющих совокупность возможных исходных данных, промежуточных и конечных результатов;
- правило начала;
- правило непосредственной переработки информации (описание последовательности действий);
- правило окончания;
- правило извлечения результатов.

Алгоритм всегда рассчитан на конкретного исполнителя. В нашем случае таким исполнителем является ЭВМ. Для обеспечения возможности реализации на ЭВМ алгоритм должен быть описан на языке, понятном компьютеру, то есть на языке программирования.

Понятия алгоритма и программы разграничены не очень чётко. Обычно программой называют окончательный вариант алгоритма решения задачи, ориентированный на конкретного пользователя.

Таким образом, можно дать следующее определение программы для ЭВМ:

Программа – это описание алгоритма и данных на некотором языке программирования, предназначенное для последующего автоматического выполнения.

3. Способы описания алгоритмов

К основным способам описания алгоритмов можно отнести следующие:

- словесно-формульный (на естественном языке);
- структурный или блок-схемный;
- с использованием специальных алгоритмических языков;
- с помощью граф-схем (граф - совокупность точек и линий, в которой каждая линия соединяет две точки. Точки называются вершинами, линии - рёбрами);
- с помощью сетей Петри.

Перед составлением программ чаще всего используются словесно-формульный и блок-схемный способы. Иногда перед составлением программ на низкоуровневых языках программирования типа языка Ассемблера алгоритм программы записывают, пользуясь конструкциями некоторого высокоуровневого языка программирования. Удобно использовать программное описание алгоритмов функционирования сложных программных систем. Так, для описания принципов функционирования ОС использовался Алголоподобный высокоуровневый язык программирования.

Словесно-формульный способ

При словесно-формульном способе алгоритм записывается в виде текста с формулами по пунктам, определяющим последовательность действий.

Пусть, например, необходимо найти значение следующего выражения:

$$y=2a-(x+6).$$

Словесно-формульным способом алгоритм решения этой задачи может быть записан в следующем виде:

1. Ввести значения a и x .
2. Сложить x и 6 .
3. Умножить a на 2 .
4. Вычесть из $2a$ сумму $(x+6)$.
5. Вывести y как результат вычисления выражения.

Блок-схемы

При блок-схемном описании алгоритм изображается геометрическими фигурами (блоками), связанными по управлению линиями (направлениями потока) со стрелками. В блоках записывается последовательность действий.

Данный способ по сравнению с другими способами записи алгоритма имеет ряд преимуществ. Он наиболее нагляден: каждая операция вычислительного процесса изображается отдельной геометрической фигурой. Кроме того, графическое изображение алгоритма наглядно показывает разветвления путей решения задачи в зависимости от различных условий, повторение отдельных этапов вычислительного процесса и другие детали.

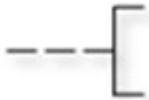
Основные элементы схем алгоритма

Наименование	Обозначение	Функция
Блок начало-конец (пуск-остановка)		Элемент отображает выход во внешнюю среду и вход из внешней среды (наиболее частое применение – начало и конец программы). Внутри фигуры записывается соответствующее действие.
Блок действия		Выполнение одной или нескольких операций, обработка данных любого вида (изменение значения данных, формы представления, расположения). Внутри фигуры записывают непосредственно сами операции, например, операцию <u>присваивания</u> : $a = 10 * b + c$.
Логический блок (блок условия)		Отображает решение или функцию переключательного типа с одним входом и двумя или более альтернативными выходами, из которых только один может быть выбран после вычисления условий, определенных внутри этого элемента. Вход в элемент обозначается линией, входящей обычно в верхнюю вершину элемента. Если выходов два или три, то обычно каждый выход обозначается линией, выходящей из оставшихся вершин (боковых и нижней). Если выходов больше трех, то их следует показывать одной линией, выходящей из вершины (чаще нижней) элемента, которая затем разветвляется. Соответствующие результаты вычислений могут записываться рядом с линиями, отображающими эти пути. Примеры решения: в общем случае – сравнение (три выхода: $>$, $<$, $=$); в <u>программировании</u> – условные операторы <u>if</u> (два выхода: <u>true</u> , <u>false</u>) и <u>case</u> (множество выходов).

Основные элементы схем алгоритма

Предопределённый процесс		Символ отображает выполнение процесса, состоящего из одной или нескольких операций, который определен в другом месте программы (в подпрограмме, модуле). Внутри символа записывается название процесса и передаваемые в него данные. Например, в программировании – вызов процедуры или функции.
Данные (ввод-вывод)		Преобразование данных в форму, пригодную для обработки (ввод) или отображения результатов обработки (вывод). Данный символ не определяет носителя данных (для указания типа носителя данных используются специфические символы).
Граница цикла		Символ состоит из двух частей – соответственно, начало и конец цикла – операции, выполняемые внутри цикла, размещаются между ними. Условия цикла и приращения записываются внутри символа начала или конца цикла – в зависимости от типа организации цикла. Часто для изображения на блок-схеме цикла вместо данного символа используют символ условия, указывая в нём решение, а одну из линий выхода замыкают выше в блок-схеме (перед операциями цикла).

Основные элементы схем алгоритма

Соединитель		<p>Символ отображает вход в часть схемы и выход из другой части этой схемы. Используется для обрыва линии и продолжения её в другом месте (для <u>избежания излишних пересечений</u> или слишком длинных линий, а также, если схема состоит из нескольких страниц). Соответствующие соединительные символы должны иметь одинаковое (при том уникальное) обозначение.</p>
Комментарий		<p>Используется для более подробного описания шага, процесса или группы процессов. Описание помещается со стороны квадратной скобки и охватывается ей по всей высоте. Пунктирная линия идет к описываемому элементу, либо группе элементов (при этом группа выделяется замкнутой пунктирной линией). Также символ комментария следует использовать в тех случаях, когда объём текста, помещаемого внутри некоего символа (например, символ процесса, символ данных и др.), превышает размер самого этого символа.</p>

4. Компьютер как исполнитель команд.

Программный принцип работы ПК

Принципы обработки информации компьютером остаются неизменными на протяжении долгих лет, особенно, если учесть скоротечность модернизации информационной и компьютерной отраслей. Для того, чтоб обработать введенную информацию в компьютер, необходимо, чтобы в машине существовали нужные определенные алгоритмы работы и обработки. Если их не будет, то информация преобразовываться не будет. Компьютер должен быть снабжен такими алгоритмами и должен уметь их применять к вводимой информации с тем, чтобы «правильно» преобразовывать ее в выходные данные. Компьютер для того и создан, чтоб пользователи имели доступ к быстрой обработке данных и ее преобразованию. Все устройство компьютера обусловлено требованием обработки информации в кратчайшие сроки, наиболее быстрым способом. Под действие данной обработки понимают любые функции, направленные на преобразование информации из одного состояния в другое. Соответственно, компьютер имеет специальное устройство, называемое процессором, которое предназначено исключительно для чрезвычайно быстрой обработки данных, со скоростями, достигающими до миллиардов операций в секунду. Процессор берет все необходимые данные из оперативной памяти. Это устройство направлено на временное хранение, как входящей, так и выходящей информации. Там же в оперативной памяти находится и место для хранения промежуточных данных, формируемых в процессе обработки информации. Таким образом, процессор как получает данные из оперативной памяти, так и записывает обработанные данные в оперативную память.