25. Алгоритм и его свойства. Способы записи алгоритмов

Автор: Александр
26.08.2014 14:18

Понятие алгоритма - одно из самых фундаментальных понятий информатики. Алгоритмизация наряду с моделированием выступает в качестве общего метода информатики. К реализации определенных алгоритмов сводятся процессы управления в различных системах, что делает понятие алгоритма близким к кибернетике.

Алгоритмы являются объектом систематического исследования пограничной между математикой и информатикой научной дисциплины, примыкающей к математической логике - теории алгоритмов. Особенность положения, однако, состоит в том, что при решении практических задач, предполагающих разработку алгоритмов для реализации на ЭВМ, и тем более при использовании на практике информационных технологий, можно, как правило, не опираться на высокую формализацию данного понятия, и поэтому представляется целесообразным познакомиться с алгоритмами и алгоритмизацией на основе содержательного толкования сущности понятия алгоритма и рассмотрения основных его свойств. При таком подходе алгоритмизация более выступает в основном как набор определенных практических приемов, особых специфических навыков рационального мышления в рамках заданных языковых средств. Можно провести аналогию между этим обстоятельством и рассмотренным выше подходом к измерению информации: тонкие математические построения при "кибернетическом" подходе не очень нужны при использовании гораздо более простого "объемного" подхода при практической работе с компьютером.

Само слово "алгоритм" происходит от algorithmi - латинской формы написания имени великого математика IX века аль-Хорезми, который сформулировал правила выполнения арифметических действий. Первоначально под алгоритмами и понимали только правила выполнения четырех арифметических действий над многозначными числами.

Алгоритм - это определенным образом организованная последовательность действий, за конечное число шагов приводящая к решению задачи.

Свойства алгоритмов:

Алгоритм должен быть составлен таким образом, чтобы исполнитель, в расчете на которого он создается, мог однозначно и точно следовать командам алгоритма и эффективно получать определенный результат. Это накладывает на записи алгоритмов целый ряд обязательных требований, суть которых вытекает, вообще говоря, уже из приведенного выше неформального толкования понятия алгоритма. Сформируем эти требования в виде перечня свойств, которым должны удовлетворять алгоритмы, адресуемые заданному исполнителю.

1. Одно из первоначальных требований, которое предъявляется к алгоритму, состоит в том, что описываемый процесс должен быть разбит на последовательность отдельных шагов. Возникающая в результате такого разбиения запись представляет собой упорядоченную совокупность четко отделенных друг от друга предписаний (директив, команд, операторов), образующих прерывную (или, как говорят, дискретную) структуру алгоритма. Только выполнив требования одного предписания, можно приступить к выполнению следующего. Дискретная структура алгоритмической записи может, например, подчеркиваться сквозной нумерацией отдельных команд алгоритма, хотя это требование не является обязательным. Рассмотренное свойство алгоритмов называют дискретность.

2. Используемые на практике алгоритмы составляются с ориентацией на определенного исполнителя. Чтобы составить для него алгоритм, нужно знать, какие команды этот исполнитель может понять и исполнить, а какие не может. Известно, что у каждого исполнителя имеется своя система команд. Очевидно, что, составляя запись алгоритма для определенного исполнителя, можно использовать лишь те команды, которые имеются в его СКИ. Это свойство алгоритмов будем называть понятностью.

3. Будучи понятным, алгоритм не должен все же содержать предписаний, смысл которых может восприниматься неоднозначно. Это означает, что одна и та же команда, будучи понятна разным исполнителям, после исполнения каждым из них должна давать одинаковый результат.

В данном случае речь фактически идет о том, что запись алгоритма должна быть настолько четкой, полной и продуманной в деталях, чтобы у исполнителя никогда не могло возникнуть потребности в принятии каких-либо самостоятельных решений, не предусмотренных составителем алгоритма. Говоря иначе, алгоритм не должен оставлять места для произвола исполнителя. Кроме того, в алгоритмах недопустимы также ситуации, когда после выполнения очередной команды алгоритма исполнителю неясно, какая из команд алгоритма должна выполняться на следующем шаге. Отмеченное свойства алгоритмов называют определенностью или детерминированностью.

4. Обязательное требование к алгоритмам - результативность. Смысл этого требования состоит в том, что при точном исполнении всех предписаний алгоритма процесс должен прекратиться за конечное число шагов и при этом должен получиться определенный ответ на вопрос задачи (либо вывод о том, что решения не существует).

5. Наиболее распространены алгоритмы, обеспечивающие решение не одной исключительной задачи, а некоторого класса задач данного типа. Это свойство алгоритма называют массовостью. В простейшем случае массовость обеспечивает возможность использования различных значений исходных данных.

Способы записи алгоритма

Основными изобразительными средствами алгоритмов являются следующие способы их записи:

- словесный;

- формульно-словесный;

- блок-схемный;

- псевдокод;

- структурные диаграммы;

- языки программирования.

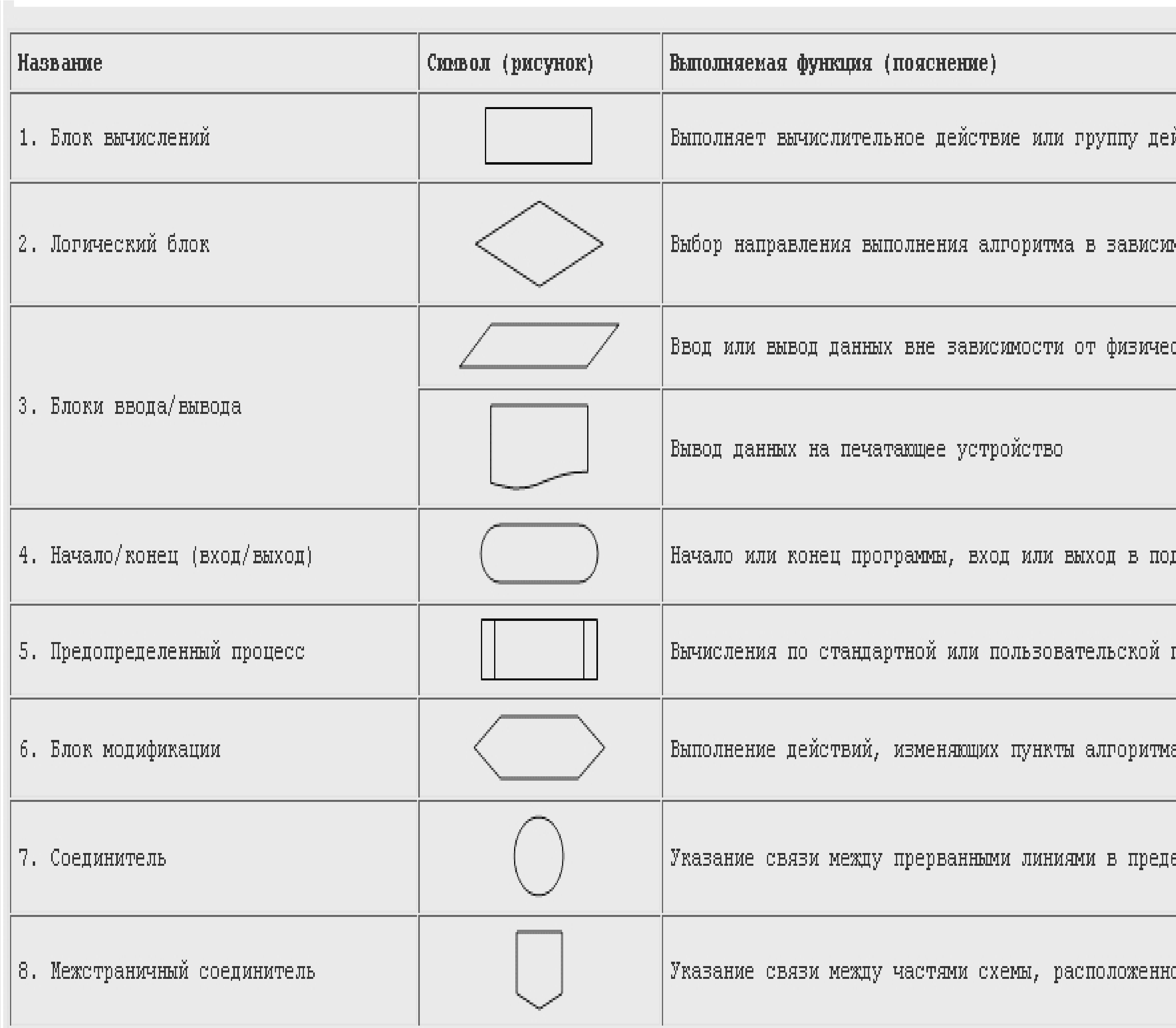
Словесный способ – это содержание этапов вычислений задается на естественном языке в произвольной форме с требуемой детализацией.

Формульно-словесный способ - это задание инструкций с использованием математических символов и выражений в сочетании со словесными пояснениями.

Блок-схемный способ - это графическое изображение логической структуры алгоритма, в котором каждый этап процесса переработки данных представляется в виде геометрических фигур (блоков), имеющих определенную конфигурацию в зависимости от характера выполняемых операций.

В приведенной ниже табл.16 представлены основные блоки, используемые при построении алгоритма в блок-схемной форме.

Таблица 16. Графические изображения блок-схем



Название	Символ (рисунок)	Выполняемая функция (пояснение)
1. Блок вычислений		Выполняет вычислительное действие или группу дей
2. Логический блок		Выбор направления выполнения алгоритма в зависи
3. Блоки ввода/вывода		Ввод или вывод данных вне зависимости от физичес
		Вывод данных на печатающее устройство
4. Начало/конец (вход/выход)		Начало или конец программы, вход или выход в по
5. Предопределенный процесс		Вычисления по стандартной или пользовательской
6. Блок модификации		Выполнение действий, изменяющих пункты алгоритма
7. Соединитель		Указание связи между прерванными линиями в преде
8. Межстраничный соединитель		Указание связи между частями схемы, расположенно

[illegible]