**Понятие алгоритма. Исполнитель алгоритма. Система команд исполнителя (на примере учебного исполнителя). Свойства алгоритма. Способы записи алгоритмов; блок-схемы.**

Само слово "алгоритм" происходит от "algorithmi" - латинской формы написания имени выдающегося математика IX века аль-Хорезми, который сформмулировал правила выполнения арифметических операций.

Что такое алгоритм?

**Алгоритм – это понятное и точное предписание исполнителю выполнить конечную последовательность команд, приводящую от исходных данных к искомому результату.**

Алгоритм выполняется за определенное количество шагов. Шаг – отдельное законченное действие.

Рассмотрим пример алгоритма для нахождения середины отрезка при помощи циркуля и линейки.  
  
*Алгоритм деления отрезка АВ пополам:*  
1) поставить ножку циркуля в точку А;  
2) установить раствор циркуля равным длине отрезка АВ;  
3) провести окружность;  
4) поставить ножку циркуля в точку В;  
5) провести окружность;  
6) через точки пересечения окружностей провести прямую;  
7) отметить точку пересечения этой прямой с отрезком АВ.

Каждое указание алгоритма предписывает исполнителю выполнить одно конкретное законченное действие. Исполнитель не может перейти к выполнению следующей операции, не закончив полностью выполнения предыдущей. Предписания алгоритма надо выполнять последовательно одно за другим, в соответствии с указанным порядком их записи. Выполнение всех предписаний гарантирует правильное решение задачи. Данный алгоритм будет понятен исполнителю, умеющему работать с циркулем и знающему, что такое поставить ножку циркуля, провести окружность и т. д.  
  
Анализ примеров различных алгоритмов показывает, что запись алгоритма распадается на отдельные указания исполнителю выполнить некоторое законченное действие. Каждое такое указание называется командой. Команды алгоритма выполняются одна за другой. После каждого шага исполнения алгоритма точно известно, какая команда должна выполняться следующей.  
  
Поочередное выполнение команд алгоритма за конечное число шагов приводит к решению задачи, к достижению цели. Разделение выполнения решения задачи на отдельные операции (выполняемые исполнителем по определенным командам) — важное свойство алгоритмов, называемое дискретностью.

Исполнитель алгоритмов

Исполнители часто встречаются в сказках. В одной из них Иван-Царевич говорит Избушке-На-Курьих-Ножках: “Избушка, избушка! Встань к лесу задом, ко мне передом!”. При этом команда должна быть задана **очень точно,** чтобы исполнитель ее понял. В сказке “Али-Баба и сорок разбойников” волшебная дверь открывалась по команде “Сезам, откройся!”. Жадный Касым, тайно проникший в пещеру, забыл эту фразу и не смог выйти из пещеры.

И Избушка-На-Курьих-Ножках, и волшебная дверь имеют много общего: они умеют понимать и выполнять некоторые точно заданные команды, то есть являются исполнителями .

• ***Исполнитель алгоритма*** — это некоторая абстрактная или реальная (техническая, биологическая или биотехническая) система, способная выполнить действия, предписываемые алгоритмом.

• ***Система команд исполнителя (СКИ)*** – это все команды, которые исполнитель умеет выполнять.

• ***Среда исполнителя*** – обстановка, в которой функционирует исполнитель.

Исполнителями могут быть

* *машины:* станки, роботы, компьютеры;
* *растения:* подсолнечник (разворачивается на солнце), кувшинки (закрываются на ночь);
* *животные:* дрессированная собака (санитар, розыскная, охотничья), кошка,
* *люди:* ученик, рабочий, учитель, ...

**Свойства алгоритма**

1. ***Понятность*** для исполнителя — т.е. исполнитель алгоритма должен знать, как его выполнять.
2. **Дискpетность** (прерывность, раздельность) — т.е. алгоpитм должен пpедставлять пpоцесс pешения задачи как последовательное выполнение пpостых (или pанее опpеделенных) шагов (этапов).
3. **Опpеделенность** — т.е. каждое пpавило алгоpитма должно быть четким, однозначным и не оставлять места для пpоизвола. Благодаpя этому свойству выполнение алгоpитма носит механический хаpактеp и не тpебует никаких дополнительных указаний или сведений о pешаемой задаче.
4. **Pезультативность** (или конечность). Это свойство состоит в том, что алгоpитм должен пpиводить к pешению задачи за конечное число шагов.
5. **Массовость** . Это означает, что алгоpитм pешения задачи pазpабатывается в общем виде, т.е. он должен быть пpименим для некотоpого класса задач, pазличающихся лишь исходными данными. Пpи этом исходные данные могут выбиpаться из некотоpой области, котоpая называется областью пpименимости алгоpитма.

**В какой форме записываются алгоритмы?**

На практике наиболее распространены следующие формы представления алгоритмов:

* словесная (записи на естественном языке);
* графическая (изображения из графических символов);
* псевдокоды (полуформализованные описания алгоритмов на условном алгоритмическом языке, включающие в себя как элементы языка программирования, так и фразы естественного языка, общепринятые математические обозначения и др.);
* программная (тексты на языках программирования).

Графический способ записи алгоритмов  
(блок-схема)

Графический способ представления алгоритмов является более компактным и наглядным по сравнению со словесным.

При графическом представлении алгоритм изображается в виде последовательности связанных между собой функциональных блоков, каждый из которых соответствует выполнению одного или нескольких действий.

Такое графическое представление называется схемой алгоритма или **блок-схемой** .

В блок-схеме каждому типу действий (вводу исходных данных, вычислению значений выражений, проверке условий, управлению повторением действий, окончанию обработки и т.п.) соответствует геометрическая фигура, представленная в виде *блочного символа*. Блочные символы соединяются *линиями переходов*, определяющими очередность выполнения действий.

**Основные алгоритмические блоки**

|  |  |  |
| --- | --- | --- |
| Овал |  | Начало или конец алгоритма |
| Параллелограмм |  | Данные, которые вводятся и/или выводятся |
| Прямоугольник |  | Выполнение действия алгоритма |
| Ромб | Да Нет | Проверка условия |

Пример: составить блок-схему для следующего алгоритма: если будет хорошая погода, то я пойду гулять. Если нет, то я позвоню другу. Если он дома, то мы поиграем по сети в компьютерную игру. Если нет, то буду читать книгу.

Погода хорошая

Да Нет

Друг дома

Позвонить другу

Пойти гулять

Да Нет

Играть в компьютер

Читать книгу