

Алгоритмы

Алгоритм — это описание процесса решения той или иной задачи.

При работе на компьютере важно знать и понимать, что такое алгоритм и для чего он нужен. Прежде чем поручить компьютеру выполнение определенной работы, следует составить план действий — алгоритм. В нем необходимо предусмотреть порядок ввода и преобразования исходных данных, а также очередность и форму вывода результата.

Появление алгоритмов связывают с зарождением математики. Более 1000 лет назад (в 825 году) ученый из города Хорезма Абдулла (или Абу Джафар) Мухаммед бен Муса аль-Хорезми создал книгу по математике, в которой описал способы выполнения арифметических действий над многозначными числами («Трактат Аль-Хорезми об арифметическом искусстве индусов»). Эти способы и сейчас изучают в школе. Само слово «алгоритм» возникло в Европе после перевода на латынь книги этого среднеазиатского математика, в которой его имя писалось как «Алгоритми». «Так говорил Алгоритми», — начинали европейские ученые, ссылаясь на правила, предложенные Мухаммедом аль-Хорезми.

Область математики, известная как теория алгоритмов, посвящена исследованию свойств, способов записи, видов и сферы применения различных алгоритмов, созданию новых алгоритмов. Научное определение понятия алгоритма дал А.Черч в 1930 году. Позже и другие математики вносили свои уточнения в это определение. В школьном курсе информатики вы будете пользоваться следующими определениями:

Алгоритм — описание последовательности действий, строгое исполнение которых приводит к решению поставленной задачи за конечное число шагов.

Алгоритмизация — процесс разработки алгоритма (плана действий) для решения задачи.

Свойства алгоритмов

Дискретность (от лат. discretus — разделенный, прерывистый). Это свойство указывает, что любой алгоритм должен состоять из конкретных действий, следующих в определенном порядке. В приведенных выше алгоритмах общим является необходимость строгого соблюдения последовательности выполнения действий. Например, вы хорошо знаете, как открывать дверь ключом. Однако, чтобы научить этому малыша, придется четко разъяснить и сами действия, и порядок их выполнения:

1. Достать ключ из кармана.
2. Вставить ключ в замочную скважину.
3. Повернуть ключ два раза против часовой стрелки.
4. Вынуть ключ.

Попробуем переставить в примере второе и третье действия. Вы, конечно, сможете выполнить и этот алгоритм, но дверь вряд ли откроется.

Детерминированность (от лат. determinate — определенность, точность). Это свойство указывает, что любое действие алгоритма должно быть строго и недвусмысленно определено в каждом случае. Например, если к остановке подходят автобусы разных маршрутов, то в алгоритме должен быть указан конкретный номер маршрута — 5. Кроме того, необходимо указать точное количество остановок, которое надо проехать, — скажем, три.

Конечность. Это свойство определяет, что каждое действие в отдельности и алгоритм в целом должны иметь возможность завершения. В приведенных примерах каждое описанное действие реально и может быть выполнено. Поэтому и алгоритм имеет предел, то есть конечен.

Массовость. Это свойство показывает, что один и тот же алгоритм можно использовать с разными исходными данными. Ниже описан алгоритм приготовления любого бутерброда.

1. Отрезать ломтик хлеба.
2. Намазать его маслом.
3. Отрезать кусок любого другого пищевого продукта (колбасы, сыра, мяса).
4. Наложить отрезанный кусок на ломоть хлеба.

Результативность. Это свойство требует, чтобы в алгоритме не было ошибок.

Способы записи алгоритма:





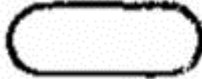

- 1) словесный;
- 2) графический (в виде схем).
- 3) Псевдокод (компактный язык описания алгоритмов, использующий ключевые слова языков программирования, но опускающий несущественные подробности и специфический синтаксис.)

Словесный способ записи это пошаговая запись действий в определенном порядке, при этом действия стоят в повелительном наклонении (см. примеры выше).

Графический способ записи.

Для записи алгоритма в виде схем используются специальные обозначения (блоки), которые определены в Государственном отраслевом стандарте ГОСТ 19.70190 (ISO 580785). В скобках указан соответствующий документ, разработанный Международной организацией по стандартизации (International Standard Organization).

Основные блоки, использующиеся в графическом способе записи

Название блока	Назначение блока	Изображение блока
Данные	Блок отображает данные алгоритма. Обозначает ввод и вывод.	
Процесс	Блок отображает функцию обработки данных (выполнение определенной операции или группы операций, приводящее к изменению значения)	
Решение	Блок отображает функцию переключательного типа, имеющую один вход и два выхода, один из которых может быть активизирован (выбран) после вычисления условий, определенных внутри этого блока.	
Линия	Блок отображает потоки данных или потоки управления	
Терминатор	Блок отображает выход во внешнюю среду и вход из внешней среды (начало или конец схемы алгоритма).	
Комментарий	Блок используют для добавления комментариев	

Правила выполнения соединений

Потоки данных или потоки управления в схемах показываются линиями. Направление потока слева направо и сверху вниз считается стандартным.

Если поток имеет направление, отличное от стандартного, стрелки должны указывать это направление. (В случаях, когда необходимо внести большую ясность в схему (например, при соединениях), на линиях используются стрелки.)

В схемах следует избегать пересечения линий.

Линии должны быть направлены к центру символа. Они должны подходить к символу либо сверху, либо слева, а исходить либо снизу, либо справа.

Виды алгоритмов.

1. Линейный алгоритм
2. Разветвленный алгоритм
3. Циклический алгоритм

Линейный (последовательный) алгоритм – описание действий, которые выполняются однократно в заданном порядке.

Линейный алгоритм применяется при вычислении арифметического выражения, если в нем используются только действия сложения и вычитания.

Примеры линейных алгоритмов – алгоритм отпирания дверей, заваривания чая и др.

Запишем алгоритм вычисления выражения

$$(51+148) \times 58 - 122 \times 64$$

Используем словесный способ записи:

1. Определить порядок выполнения действий

$$(51+148) \times 58 - 122 \times 64$$

1 2 4 3

2. Выполнить действие 1 (сложить числа в скобках **51+148**)
3. Выполнить действие 2 (умножить результат действия 1 **x 58**)
4. Выполнить действие 3 (умножить **122 x 58**)
5. Выполнить действие 4 (вычесть результат действия 3 из результата действия 2)

Домашнее задание:

Составить линейный алгоритм.



Условие в алгоритмах. Порядок составления разветвленного алгоритма

В повседневной жизни линейные алгоритмы встречаются редко. В жизни часто приходится принимать решения в зависимости от сложившейся обстановки. Обычно появляются какие-нибудь условия, которые изменяют порядок действий, т.е. изменяют алгоритм. Например, во время урока заканчивается паста в авторучке; вы подошли к светофору, а сигнал светофора переключился на красный свет; если на улице идет дождь, то мы берем зонт и надеваем плащ.

Условие – это выражение, которое находится между словами **если** и **то** принимает значение **истина** или **ложь**.

Логика принятия решения можно описать так:

ЕСЛИ <условие> ТО <действия1>
ИНАЧЕ <действия2>

Примеры:

ЕСЛИ уроки выучены, ТО иди гулять, ИНАЧЕ учи уроки дальше.

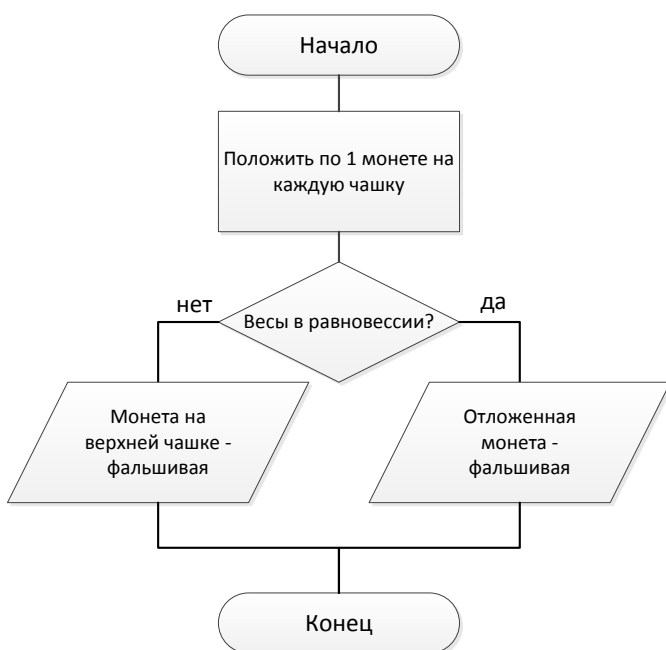
ЕСЛИ уроки выучены, ТО иди гулять, ИНАЧЕ учи уроки дальше.

В некоторых случаях <действия2> могут отсутствовать.

Таким образом, появляется новый вид алгоритма – разветвлённый.

Разветвлённый алгоритм – это алгоритм, в котором в зависимости от условия выполняется либо одна, либо другая последовательность действий.

Задача. Из трех монет одинакового достоинства одна фальшивая (более лёгкая). Как её найти с помощью одного взвешивания на чашечных весах без гирь? Составим алгоритм.



Домашнее задание:
Составить схему алгоритма перехода через улицу, выполняя действия в соответствии с сигналами светофора.

Цикл. Составление циклических алгоритмов

В окружающем мире многие процессы основываются на многократном повторении последовательности действий (например, смена времен года). На практике также часто встречаются задачи, в которых действие или группа действий повторяются некоторое количество раз. Такие процессы называют циклическими.

Циклический алгоритм – описание действий, которые повторяются указанное число раз или пока не выполнится заданное условие.

Рассмотрим пример: Робот должен покрасить забор.

Для робота составлен алгоритм:

1. Покрасить доску
2. Переместиться к следующей доске
3. Перейти к п.1

Когда робот сможет закончить работу? Какие изменения нужно внести в алгоритм?

Необходимо добавить правило (условие) окончания работы.

Алгоритм будет следующим образом:

1. Покрасить доску
2. Есть ещё доска?

Если Да, то переместиться к следующей: затем перейти к п.1

Если Нет, то завершить работу.



Домашнее задание:

Составьте алгоритм окраски забора