

# Алгоритмы в нашей жизни

Подготовила: ученица 9 класса Пустовитова Е.А.

# Что такое алгоритм

- Термин «алгоритм» произошёл от имени великого математика Мухаммеда аль-Хорезми (по-латыни algorithmus). Мухаммед аль-Хорезми ещё в IX веке разработал правила выполнения четырёх действий арифметики.
- Алгоритм — набор инструкций, описывающих порядок действий исполнителя для достижения некоторого результата. Алгоритмы имеют свойства, форму представления, а так же структуру.
- Разработчиком алгоритмов является человек. Исполняют алгоритмы люди и всевозможные технические устройства.
- Исполнитель – это некоторый объект (человек, животное, техническое устройство), способный выполнять определенный набор команд. Команды, которые может выполнить конкретный исполнитель, образуют систему команд исполнителя (СКИ).
- Исполнители делятся на формальные и неформальные.
- В роли неформального исполнителя чаще всего выступает человек. Неформальный исполнитель сам отвечает за свои действия.
- В роли формального исполнителя чаще всего выступает техническое устройство.

- Формальный исполнитель одну и ту же команду всегда выполняет одинаково. Для каждого формального исполнителя можно указать:
  - круг решаемых задач;
  - среду;
  - систему команд;
  - систему отказов;
  - режимы работы.
- Формы записи алгоритмов – словесная и графическая.
- Алгоритмы, исполнителем которых является человек, удобно записывать в словесной форме, в табличной форме, в виде блок-схем.
- Для обозначения шагов в блок-схеме используются фигуры (овал, параллелограмм, ромб, прямоугольник и другие).
- Алгоритм, записанный на языке, понятном исполнителю, называется программой.
- Алгоритмы делятся на 3 типа:
  - линейные;
  - ветвление;
  - циклические.
- Линейным называется алгоритм, в котором команды выполняются в порядке их записи.
- Ветвлением называется алгоритм, при котором в зависимости от выполнения некоторого условия совершается одна или другая последовательность команд.
- Циклическим называется алгоритм, в котором повторяется выполнение одной и той же последовательности команд.

# Свойства алгоритма:

- Дискретность — алгоритм должен представлять процесс решения задачи как последовательное выполнение некоторых простых шагов.
- Детерминированность. В каждый момент времени следующий шаг работы однозначно определяется состоянием системы. Таким образом, алгоритм выдаёт один и тот же результат для одних и тех же исходных данных.
- Понятность — алгоритм должен включать только те команды, которые доступны исполнителю и входят в его систему команд.
- Массовость. Алгоритм должен быть применим к разным наборам исходных данных.
- Результативность — завершение алгоритма определёнными результатами

# Алгоритмы в нашей жизни

- Любой человек ежедневно встречается с множеством задач: от самых простых и хорошо известных до очень сложных. Для многих задач существуют определенные правила (инструкции, предписания), объясняющие исполнителю, как решать данную задачу. Эти правила человек может изучить заранее или сформулировать сам в процессе решения задачи. Чем точнее и понятнее будут описаны правила решения задач, тем быстрее человек овладеет ими и будет эффективнее их применять.
- Решение многих задач человек может передавать техническим устройствам - автоматам, роботам, компьютерам. Применение таких технических устройств предъявляет очень строгие требования к точности описания правил и последовательности выполнения действий. Поэтому разрабатываются специальные языки для четкого и строгого описания различных правил. Это одна из задач информатики.

- Каждый из нас ежедневно использует различные алгоритмы: инструкции, правила, рецепты и т. п. Обычно мы это делаем не задумываясь. Например, открывая дверь ключом, никто не размышляет над тем, в какой последовательности выполнять действия. Однако, чтобы кого –нибудь (скажем, младшего брата) научить открывать дверь, придется четко указать и сами действия, и порядок их выполнения. Например, так:
  - Достать ключ.
  - Вставить ключ в замочную скважину.
  - Повернуть ключ 2 раза против часовой стрелки.
  - Вынуть ключ.

- А теперь представьте себе, что вас пригласили в гости. Наверняка вы попросите подробно и точно объяснить, как добраться. Вот как может выглядеть объяснение:
  
- Выйти из дома.
- Повернуть направо.
- Пройти 2 квартала до автобусной остановки..
- Сесть в автобус № 25, идущий к центру города.
- Проехать 3 остановки.
- Выйти из автобуса.

- Посмотрим на эти алгоритмы. На первый взгляд, между ними нет ничего общего. Одно дело – открывать дверь, другое – ехать в гости. Однако если приглядеться внимательно, можно заметить существенное сходство между ними. Прежде всего, это строгий порядок выполнения действий. Давайте переставим в первом алгоритме второе и третье действия:
- Достать ключ.
- Повернуть ключ 2 раза против часовой стрелки.
- Вставить ключ в замочную скважину.
- Вынуть ключ.



- Вы, конечно, сможете выполнить и этот алгоритм. Но дверь вряд ли откроется. А что произойдет, если поменять местами четвертое и пятое действия во втором алгоритме? Он станет невыполнимым! Итак, мы убедились, что для алгоритма важен не только набор действий, но и то, как они организованы, т. е. в каком порядке выполняются.
- Мы можем теперь сказать, что алгоритмы - это строго определенная последовательность действий. Существует очень много определений понятия алгоритм. И надо подчеркнуть, что в информатике это понятие является основным. Таким же, какими являются понятия точки, прямой и плоскости в геометрии, пространства и времени в физике, вещества в химии. Поэтому мы не сможем дать полное определение алгоритма, а будем уточнять смысл этого понятия на примерах.

□ Алгоритмы принято записывать с помощью служебных слов, т.е. имеется алгоритмический язык, алфавит:

□ алг (алгоритм) название

□     арг (аргументы)

□     рез (результаты)

□     нач (начало)

□             команды

□     кон (конец)

□ Легко и просто было бы жить (даже неинтересно), если бы удалось раз и навсегда расписать, какие поступки и в какой последовательности совершать. На самом деле нам приходится принимать решения в зависимости от создавшейся ситуации. Если идет дождь, то мы надеваем плащ. Если жарко, то идем купаться. Иногда встречаются и более сложные положения, когда надо сделать выбор. В таких случаях говорят, что алгоритм содержит составную команду или ветвление. Команда ветвления записывается следующим образом:

□ если условие

□       то серия 1

□       иначе серия 2

□ все

- В своей практической деятельности мы постоянно встречаемся с задачами, для решения которых требуется многократно повторять одни и те же действия. Именно для этого применяется составная команда повторения (цикл). Команда повторения записывается так:
  - пока условие
  - нц
  - серия
  - кц
- На уроках информатики мы составляли очень много алгоритмов из жизни, учебных предметов, сказок и т.д. Но нас заинтересовало то, а можно ли алгоритмы каким - то образом классифицировать, т. е. составить модель классификации алгоритмов.

# Алгоритмы в художественных произведениях

- Сказка «Гуси-лебеди»
- если съешь ржаного пирожка
- то спрячу
- иначе не спрячу
- все
- «Горячий камень», А.П. Гайдар:
- если кто снесет этот камень на гору и там разобьет на части
- то тот вернет свою молодость и начнет жить сначала
- все
- Если внимательно прочитать любую сказку, то можно сделать вывод, что все сказки строятся по тому, или иному типу алгоритма. Чаще всего алгоритмы можно применить в эпизодах сказок. В результате в одной сказке мы встретим различные типы алгоритмов.
- Например, в сказке «Теремок» пока приходят сказочные персонажи – циклический алгоритм. Затем пришел медведь, и теремок развалился, - разветвляющийся алгоритм.

- В сказке «Гуси – лебеди» эпизоды с яблоней, печкой, рекой очень похожи друг на друга и реализованы по разветвляющемуся алгоритму: выполнит девочка просьбу или нет. Эпизоды сказки повторяются. Можно смело утверждать, что это в целом циклический алгоритм. Финал сказки вновь ветвление. Если все просьбы выполнены – благополучный исход, если не выполнены – печальный.
  
- Мы анализировали следующие сказки:
  - Колобок.
  - Курочка Ряба.
  - Конек Горбунок.
  - Сказка о рыбаке и рыбке.
  - Иван Меньшой – разум большой.