

Предмет: информатика

Класс: 9

Тема: «**Понятие алгоритма. Виды алгоритмов и их свойства**»

Цель: Интересно и понятно объяснить новую тему.

Задачи:

- ознакомить учащихся с темой: «Понятие алгоритма. Виды алгоритмов и их свойства»;
- развитие у детей памяти, мышления, логического мышления;
- учащиеся **должны знать** понятие алгоритма, свойства алгоритмов;
- учащиеся **должны уметь** приводить примеры алгоритмов.

Ход урока:

1. Изучить новую тему и записать, выучить основные понятия.

Знакомство с понятием алгоритма начнем с рассмотрения примера. Предположим, вы хотите вырезать из бумаги модель автомобиля. Результат во многом будет зависеть от вашего умения и опыта. Однако достичь поставленной цели окажется гораздо легче, если вы предварительно наметите план действий, например, следующий:

1. Изучить образ автомобиля по имеющейся модели.
2. Начертить двери, кузов машины на бумаге.
3. Вырезать эскизы.
4. Попробовать скрепить эскизы, откорректировать ошибки.
5. Склеить части модели.

Следуя подготовленному плану, любой человек, даже не обладающий художественными способностями, но имеющий терпение, обязательно получит хороший результат. Подобный план с подробным описанием действий, необходимых для получения ожидаемого результата, получил название алгоритма.

Понятие алгоритма

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 спо собы выполнения арифметических действий над многозначными числами. Само слово «алгоритм» возникло в Европе после перевода на латынь книги этого среднеазиатского математика, в которой его имя писалось как «Алгоритми».

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

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

Другие примеры алгоритмов

Любой прибор, купленный в магазине, снабжается инструкцией по его использованию.

Каждый шофер должен знать правила дорожного движения.

Массовый выпуск автомобилей стал возможен только тогда, когда был придуман порядок сборки машины на конвейере.

Свойства алгоритмов

Мы на каждом шагу встречаем алгоритмы. Некоторые из них мы выполняем машинально, даже не задумываясь об этом. Выполняя некоторые действия мы даже не подозреваем, что выполняем определенный алгоритм. Например, вы хорошо знаете, как открывать дверь ключом. Однако, чтобы научить этому малыша, придется четко разъяснить и сами действия, и порядок их выполнения. Запишите **алгоритм выполнения открывания двери**.

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

Запишите другой алгоритм. *Вас пригласили в гости и подробно объяснили, как добраться:*

1. Выйти из дома.
2. Повернуть направо.
3. Пройти два квартала до остановки.
4. Сесть в автобус № 5, идущий к центру города.
5. Проехать три остановки.
6. Выйти из автобуса.
7. Найти по указанному адресу дом и квартиру.

Эти примеры не что иное, как алгоритм. Несмотря на значительное различие в сути самих действий этих примеров, можно найти в них много общего. Эти *общие характеристики называют свойствами алгоритма*. Рассмотрим их.

Дискретность (от лат. *discretus* — *разделенный, прерывистый*) – это разбиение алгоритма на ряд отдельных законченных действий (*шагов*). В приведенных выше алгоритмах общим является необходимость строгого соблюдения последовательности выполнения действий. Попробуем переставить в первом примере второе и третье действия. Вы, конечно, сможете выполнить и этот алгоритм, но дверь вряд ли откроется. А если поменять местами, предположим, пятое и второе действия во втором примере, алгоритм станет невыполнимым.

Детерминированность (от лат. *determinate* — *определенность, точность*) - 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 - 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 - один и тот же алгоритм можно использовать с разными исходными данными.

Например: 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 - в алгоритме не было ошибок.

Пример: рассмотрим алгоритм нахождения большего из двух заданных чисел А и В:

1. Из числа А вычесть число В.
2. Если получилось отрицательное значение, то сообщить, что число В больше.
3. Если получилось положительное значение, то сообщить, что число А больше.

При всей простоте и очевидности алгоритма, не каждый сразу поймет его ошибочность.

Ведь если оба числа равны, то не получится ни какого сообщения. Значит, надо обязательно предусмотреть это вариант, например:

1. Из числа А вычесть число В.

2. Если получилось отрицательное значение, то сообщить, что число В больше.
3. Если получилось положительное значение, то сообщить, что число А больше.
4. Если получился ноль, то сообщить, что числа равны.

Виды алгоритмов

Существует 4 вида алгоритмов: линейный, циклический, разветвляющийся, вспомогательный.

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

Линейными являются алгоритмы отпираания дверей, заваривания чая, приготовления одного бутерброда. Линейный алгоритм применяется при вычислении арифметического выражения, если в нем используются только действия сложения и вычитания.

Циклический алгоритм — описание действий, которые должны повторяться указанное число раз или пока не выполнено заданное условие. Перечень повторяющихся действий называется телом цикла.

Многие процессы в окружающем мире основаны на многократном повторении одной и той же последовательности действий. Каждый год наступают весна, лето, осень и зима. Жизнь растений в течение года проходит одни и те же циклы. Подсчитывая число полных поворотов минутной или часовой стрелки, человек измеряет время.

Условие — выражение, находящееся между словом «если» и словом «то» и принимающее значение «истина» или «ложь».

Разветвляющийся алгоритм — алгоритм, в котором в зависимости от условия выполняется либо одна, либо другая последовательность действий.

Примеры разветвляющихся алгоритмов: если пошел дождь, то надо открыть зонт; если болит горло, то прогулку следует отменить; если билет в кино стоит не больше десяти рублей, то купить билет и занять свое место в зале, иначе (если стоимость билета больше 10 руб.) вернуться домой.

В общем случае схема разветвляющегося алгоритма будет выглядеть так: «если условие, то..., иначе...». Такое представление алгоритма получило название полной формы.

Неполная форма, в которой действия пропускаются: «если условие, то...».

Вспомогательный алгоритм — алгоритм, который можно использовать в других алгоритмах, указав только его имя.

Например: вы в детстве учились суммировать единицы, затем десятки, чтобы суммировать двузначные числа, содержащие единицы вы не учились новому методу суммирования, а воспользовались старыми методами.

Домашняя работа

Придумать свои примеры на каждый вид алгоритма.

Литература

- Семакин «Информатика» для 9 класса
- Материалы сайта: <http://www.alleng.ru>
- А.Г. Гейн, А.И. Сенокосов, Н.А. Юнерман «Информатика»