

## План-конспект урока по информатике

**Дата:** 01.12.2023

**Класс:** 7

**Учитель:** Кунашко Анна Андреевна

**Тема:** Алгоритмы и исполнители

**Цели урока:**

*Образовательные:* формирование у учащихся представления об алгоритме, исполнителе алгоритма, система команд исполнителя, ознакомить со способами описания алгоритмов и основными типами алгоритмов, вспомнить исполнитель Чертежник.

*Развивающие:* содействовать развитию алгоритмического мышления; создание ситуации для формирования коммуникативных умений, рефлексивных способностей, самостоятельности учащихся.

*Воспитательные:* воспитание трудолюбия, ответственности за результаты своего труда; воспитание культуры делового общения при совместной работе в группе, нацеленность на результативность обучения.

**Тип урока:** урок усвоения и закрепление новых знаний.

**Оборудование:** класс компьютерной техники, мультимедийный проект.

**Литература:** Котов, В. М. Информатика: учеб. пособие для 7 кл. учреждений общ. сред. образования с белорус. и рус. обучения / В. М. Котов, А. И. Лапо, Е. Н. Войтехович. – Минск: Нар. асвета, 2017.

### **Структура урока:**

1. Организационный момент (1 мин).
2. Этап актуализации знаний (4 мин).
3. Целемотивационный этап (2 мин).
4. Этап изучения новой темы (15 мин).
5. Физкультминутка (2 мин).
6. Проверка понимания и закрепления изученного материала (15 мин).
7. Информация о домашнем задании (1 мин).
8. Подведение итогов урока (2 мин).
9. Рефлексия (3 мин).

### **Ход урока:**

#### **1. Организационный момент.**

Здравствуйте, дети!

Сегодня на уроке мы начнем изучать новую тему. Запишите сегодняшнюю тему в тетрадь: «Алгоритмы и исполнители».

#### **2. Этап актуализации знаний.**

Начнем наш урок с проверки понимания пройденной темы.

1. Что называют запросом в поисковой системе?
2. Как исключить некоторые записи из области поиска?
3. Какие операторы можно использовать в поисковых запросах?

#### **3. Целемотивационный этап.**

Сегодня мы изучаем тему «Алгоритмы и исполнители». Мы должны за сегодня выполнить некоторые цели: знать: — что такое алгоритм, исполнитель алгоритма, какие есть записи алгоритмов; уметь: — составлять алгоритмические конструкции следовательно, решать задачи с исполнителем Чертежник.

#### **4. Этап изучения новой темы.**

## 8.1. Понятие алгоритма

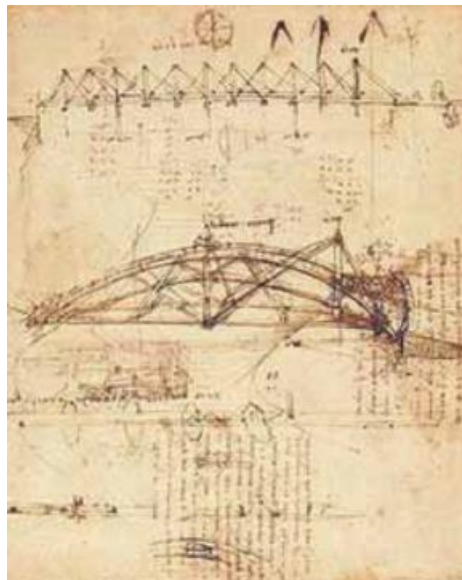
Вспомним некоторые понятия, изученные в 6-м классе.

**Алгоритм** — понятная и конечная последовательность точных действий (команд), формальное выполнение которых позволяет получить решение поставленной задачи.

**Исполнитель алгоритма** — человек, группа людей или техническое устройство, которые понимают команды алгоритма и умеют правильно их выполнять.

**Система команд исполнителя** — команды, которые понимает и может выполнить исполнитель.

Алгоритмы построения чертежей человек разрабатывает с глубокой древности. Появление чертежей связано с практической деятельностью человека — возведением укреплений и городских построек. Первые сведения о чертежах, напоминающих современные, связаны с именем Леонардо да Винчи (1452—1519) — итальянского ученого и художника, который в технических рисунках и эскизах раскрывал свои идеи в области техники и строительства.



В настоящее время чертежи широко применяются в различных отраслях строительства, сельского хозяйства, промышленности и т. д. Сегодня для построения чертежей используются специальные программы, позволяющие автоматизировать процесс черчения. Вот логотипы подобных программ:



Любой исполнитель имеет ограниченную систему команд. Все они разделяются на группы:

1. Команды, которые непосредственно выполняет исполнитель.
2. Команды, меняющие порядок выполнения других команд исполнителя.

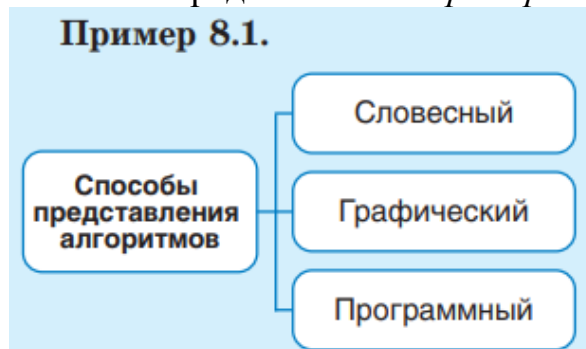
Компьютер является универсальным исполнителем.

Запись алгоритма в виде последовательности команд, которую может выполнить компьютер, называют **программой**.

Существуют следующие способы представления алгоритмов:

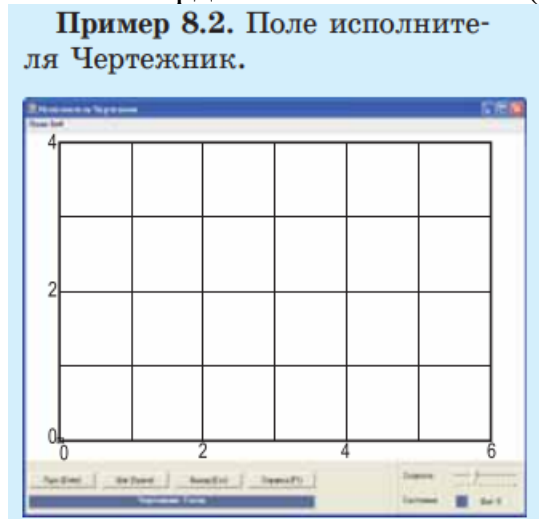
- **словесный** (описание алгоритма средствами естественного языка с точной и конкретной формулировкой фраз);

- **графический (блок-схема)** (графическое изображение команд алгоритма с использованием геометрических фигур, или блоков, и стрелок, соединяющих эти блоки и указывающих на порядок выполнения команд);
  - **программный** (запись алгоритма в виде программы).
- (Схематически данные способы представлены в *примере 8.1.*)



## 8.2. Исполнитель Чертежник

В 6-м классе вы познакомились с исполнителем Чертежник, предназначенным для построения рисунков и чертежей на координатной плоскости (*пример 8.2.*).



Чертежник имеет перо, которым он может рисовать отрезки на плоскости. В исходном положении перо поднято и находится над точкой  $(0, 0)$  — началом координат. После завершения рисования перо также должно быть поднято.

**Система команд Чертежника:**

Команда	Действие
ToPoint ( $x, y$ )	Переместить перо в точку $(x, y)$
PenUp	Поднять перо
PenDown	Опустить перо
Field ( $n, m$ )	Создать поле размером $n \times m$
OnVector ( $a, b$ )	Сместить перо на $a$ единиц по горизонтали и $b$ единиц по вертикали

*Пример 8.3.* Составим алгоритм решения задачи.

Прямоугольный участок, длина которого в 2 раза больше ширины, огородили забором длиной 120 м. Определите длину и ширину участка. Напишите программу, выполнив которую исполнитель Чертежник построит чертеж забора этого участка. Масштаб: 1 клетка равна 10 м.

Словесное описание алгоритма:

1. Длина участка в 2 раза больше ширины, поэтому в сумме длина и ширина составят 3 одинаковые части. Забор огораживает участок по периметру, равному удвоенной сумме длины и ширины, т. е. периметр равен 6 одинаковым частям.
2. Ширина:  $120 : 6 = 20$  м. 3. Длина в 2 раза больше ширины:  $20 * 2 = 40$  м.

Пример 8.3. Запись алгоритма по действиям:

- 1)  $1 + 2 = 3$  (части);
- 2)  $3 \cdot 2 = 6$  (частей);
- 3)  $120 : 6 = 20$  (м);
- 4)  $20 \cdot 2 = 40$  (м).

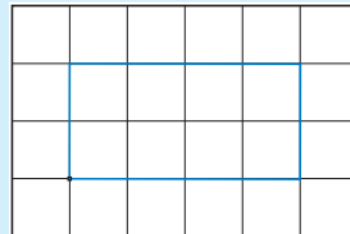
Блок-схема алгоритма:



Программа для исполнителя:

```
uses Drawman;
begin
  Field(6, 4);
  ToPoint(1, 1);
  PenDown;
  OnVector(4, 0);
  OnVector(0, 2);
  OnVector(-4, 0);
  OnVector(0, -2);
  PenUp;
end.
```

Результат работы программы:



### 8.3. Алгоритмическая конструкция следование

Существует большое количество алгоритмов, в которых все команды выполняются последовательно одна за другой в том порядке, в котором они записаны. В подобных алгоритмах отсутствуют команды, меняющие порядок выполнения других команд. Такие программы вы составляли в прошлом году для исполнителя Чертежник.

**Алгоритмическая конструкция следование** — последовательность команд алгоритма, которые выполняются в том порядке, в котором они записаны.

Алгоритмическая конструкция *следование* отображает естественный, последовательный порядок выполнения действий в алгоритме.

Следование использовалось в *примере 8.3*, в котором описывались алгоритмы вычисления длины и ширины участка и построения прямоугольника исполнителем Чертежник.

Алгоритмическая конструкция следование представлена в *примерах 8.4 и 8.5*.

Пример 8.4. Алгоритм изготовления бутерброда:

1. Отрезать ломтик батона.
2. Положить на батон лист салата.
3. Отрезать кусочек ветчины.
4. Положить ветчину на лист салата.
5. Отрезать кусочек помидора.
6. Положить помидор на ветчину.



Пример 8.5. Алгоритм выполнения лабораторной работы по биологии «Строение инфузории туфельки»:

1. Рассмотреть внешний вид и внутреннее строение инфузории туфельки.
2. Зарисовать инфузорию туфельку и обозначить названия ее органов.
3. Подвести итог работе.



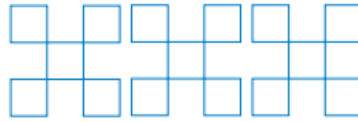
### 8.4. Вспомогательные алгоритмы

Часто в одной программе нужно рисовать одно и то же изображение несколько раз. Получение этого изображения удобно оформить в виде вспомогательного алгоритма, который можно использовать нужное число раз.

**Вспомогательный алгоритм** — алгоритм, целиком используемый в составе другого алгоритма.

Вспомогательный алгоритм решает некоторую подзадачу основной задачи. Вызов вспомогательного алгоритма в программе заменяет несколько команд одной.

*Пример 8.6.* Напишем программу, выполнив которую Чертежник нарисует изображение:



Данный рисунок состоит из одинаковых фигур. Для рисования одной из них можно оформить вспомогательный алгоритм *figura*.

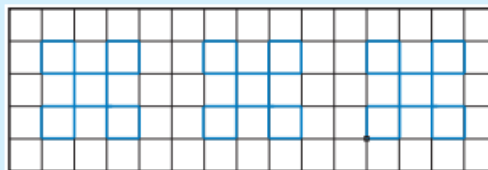
Описание основного алгоритма:  
*перемещение в начальную точку;*  
*рисование фигуры;*  
*перемещение ко второй фигуре;*  
*рисование фигуры;*  
*перемещение к третьей фигуре;*  
*рисование фигуры.*

**Пример 8.6.** Программа для исполнителя Чертежник будет следующей:

```
uses Drawman;  
procedure figura;  
begin  
  PenDown;  
  OnVector(1, 0);  
  OnVector(0, 3);  
  OnVector(-1, 0);  
  OnVector(0, -1);  
  OnVector(3, 0);  
  OnVector(0, 1);  
  OnVector(-1, 0);  
  OnVector(0, -3);  
  OnVector(1, 0);  
  OnVector(0, 1);  
  OnVector(-3, 0);  
  OnVector(0, -1);  
  PenUp;  
end;
```

```
begin  
  Field(15, 5);  
  ToPoint(1, 1);  
  Figura;  
  ToPoint(6, 1);  
  Figura;  
  ToPoint(11, 1);  
  Figura;  
end.
```

Результат выполнения программы:



При решении задач над проектом могут работать несколько человек. Каждый из членов коллектива делает часть своей работы и оформляет ее как отдельный вспомогательный алгоритм.

Построение алгоритмов часто выполняют **методом пошаговой детализации**. При этом сложная задача разбивается на ряд более простых. Для каждой подзадачи составляется свой вспомогательный алгоритм. Подзадачи могут разбиваться на еще более простые подзадачи.

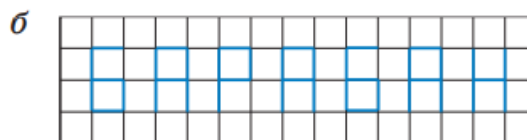
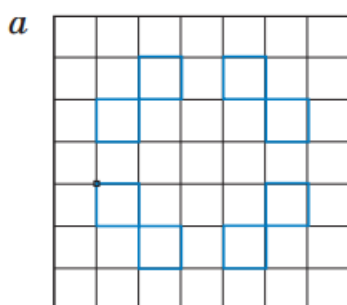
**5. Физкультминутка.**

**6. Проверка понимания и закрепления изученного материала.**

- 1 Какой рисунок получится после выполнения Чертежником следующей программы? Изобразите рисунок и проверьте правильность своих действий, выполнив программу на компьютере.

```
uses Drawman;  
begin  
  Field(8, 8);  
  ToPoint(2, 1);  
  PenDown;  
  OnVector(4, 0);  
  OnVector(0, 1);  
  OnVector(1, 0);  
  OnVector(0, 4);  
  OnVector(-1, 0);  
  OnVector(0, 1);  
  OnVector(-4, 0);  
  OnVector(0, -1);  
  OnVector(-1, 0);  
  OnVector(0, -4);  
  OnVector(1, 0);  
  OnVector(0, -1);  
  PenUp;  
end.
```

- 2 Напишите для исполнителя Чертежник программы получения следующих изображений:



- 3 Придумайте свои рисунки и составьте программы для их рисования с помощью исполнителя Чертежник.

## 7. Подведение итогов.

Наш урок подходит к концу, давайте поговорим о затруднениях в заданиях. Кому какое задание показалось сложнее?

*(оцениваю работу учащихся на уроке выборочно)*

## 8. Информация о домашнем задании.

§ 8

## 9. Этап рефлексии.

Предлагаю закончить фразы:

«Сегодня я понял, что...»

«Урок научил...»

«Я бы изменил в уроке ...»

## Конспект ученика

Алгоритмы и исполнители

01.12.2023

**Алгоритм** — понятная и конечная последовательность точных действий (команд), формальное выполнение которых позволяет получить решение поставленной задачи.

**Исполнитель алгоритма** — человек, группа людей или техническое устройство, которые понимают команды алгоритма и умеют правильно их выполнять.

**Система команд исполнителя** — команды, которые понимает и может выполнить исполнитель.

Запись алгоритма в виде последовательности команд, которую может выполнить компьютер, называют **программой**.

Существуют следующие способы представления алгоритмов:

- **словесный** (описание алгоритма средствами естественного языка с точной и конкретной формулировкой фраз);
- **графический (блок-схема)** (графическое изображение команд алгоритма с использованием геометрических фигур, или блоков, и стрелок, соединяющих эти блоки и указывающих на порядок выполнения команд);
- **программный** (запись алгоритма в виде программы).

**Алгоритмическая конструкция следование** — последовательность команд алгоритма, которые выполняются в том порядке, в котором они записаны.

**Вспомогательный алгоритм** — алгоритм, целиком используемый в составе другого алгоритма.

## Оформление классной доски

Анна Андреевна	Алгоритмы и исполнители    01.12.2023  Алгоритм – Исполнитель алгоритма – Программа – Алгоритмическая конструкция следование – Вспомогательный алгоритм –	§ 8
-------------------	---	-----