



# ПРОГРАММИРОВАНИЕ В ШКОЛЕ

# PYTHON

## **Урок 6: Условные операторы.**

### Программирование ветвящихся алгоритмов

**Цель урока** – общее представление об условных операторах, их использование на простейших примерах.

**Основные операторы:** IF , ELIF, ELSE

**Основные понятия:**

- Условный оператор
- Ветвящийся алгоритм

**Ветвящийся алгоритм** – алгоритм в котором то или иное действие выполняется в зависимости от истинности или ложности некоторого условия.

**Условие**, которое влияет на выбор следующего действия **должно принимать** значение **истина** (true) или **ложь** (false)

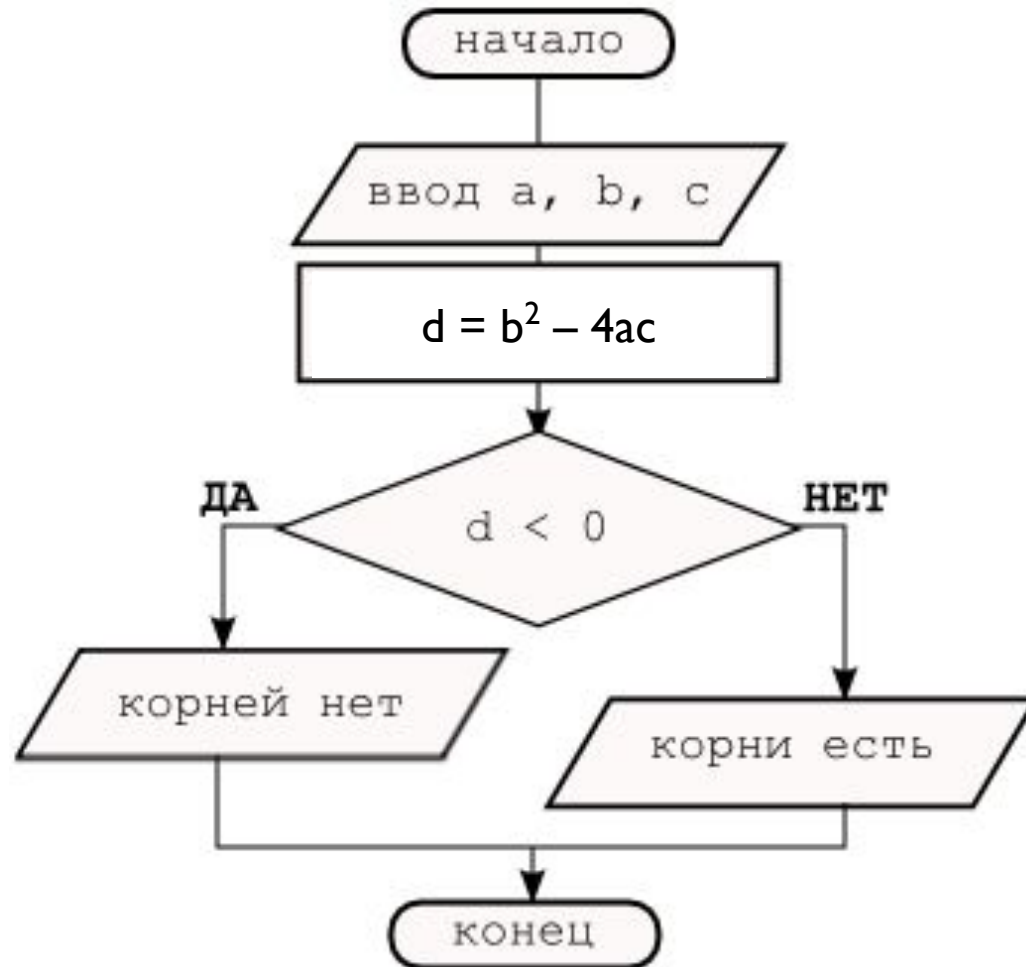


Рисунок 1: пример ветвящегося алгоритма

**Условный оператор IF** – оператор (команда) языка программирования Python, позволяющий реализовать выполнение того или иного действия в зависимости от выполнения условия.

**IF** *выражение:*

*действия для истинности выражения*

**ELSE:**

*действия для ложности выражения*

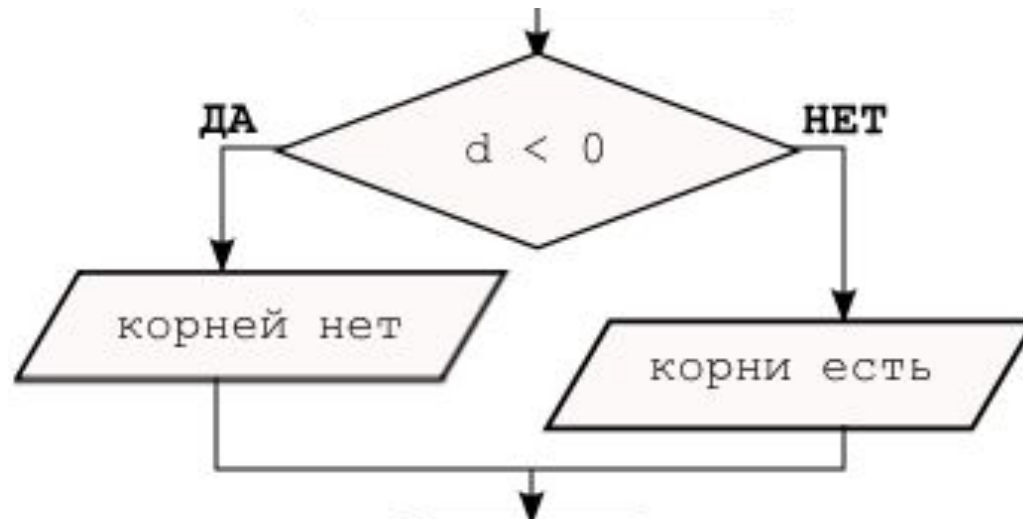


Рисунок 2: часть алгоритма с проверкой условия

```
IF d<0:  
    print "Корней нет"  
ELSE:  
    print "Корни есть"
```

Фрагмент 1: использование условного оператора

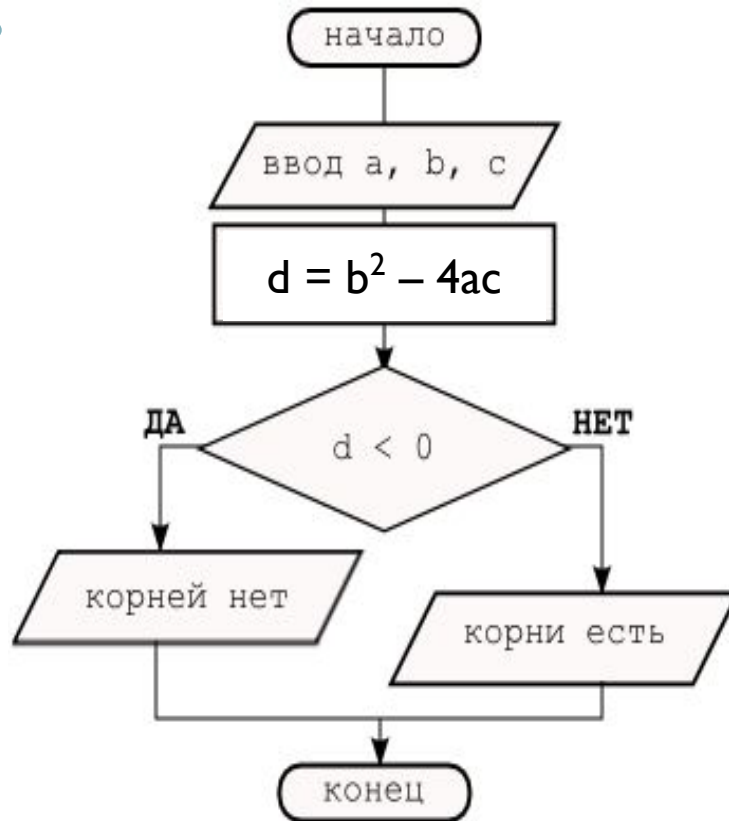


Рисунок 3: пример ветвящегося алгоритма

```

a = int(input())
b = int(input())
c = int(input())
d = b**2 + 4*a*c
  
```

```

if d < 0:
    print('Корней нет')
else:
    print('Корни есть')
  
```

Фрагмент 2: пример решения ветвящегося алгоритма

## Логические операторы проверки условия

**AND** – логическое **И** - проверяется истинность или ложь одновременно всех сравниваемых условий

*X=1 and Y=1*

**OR** – логическое **ИЛИ** – проверяется истинность или ложность одного из сравниваемых условий

*X=1 or Y=1*



## Логические операторы проверки условия

**==** - проверка на равенство

*IF X == 1:*

**!=** - проверка на Неравенство

*IF X != 1:*

## Логические операторы проверки условия

**>** - проверка на большее значение

*IF X > I:*

**>=** - проверка большее или равное значение

*IF X >= I:*

**<** - проверка на меньшее значение

*IF X < I:*

**<=** - проверка на меньшее или равное значение

*IF X <= I:*

## Задание:

Напишите программу,  
которая ожидает ввода  
числа с клавиатуры и  
определяет,  
положительное  
это число или  
отрицательное.

```
a = int(input())  
  
if a < 0:  
    print('Отрицательное')  
else:  
  
    print('Положительное')
```

В вашем решении необходимо предусмотреть **несколько вариантов:**

1. Введенное число меньше нуля
2. Введенное число больше нуля
3. Введенное число **равно** нулю

Условный оператор **IF** имеет возможность проверить сразу несколько решений:

**IF**  $a < 0$ :

```
print "Число отрицательное"
```

**ELIF**  $a = 0$ :

```
print "Ноль"
```

**ELSE**:

```
print "Число положительное"
```

## Задание:

Напишите программу, которая ожидает ввода двух чисел с клавиатуры и определяет, какое число больше, а какое меньше.

Не забудьте учесть вариант, когда числа будут равны друг другу.

```
a = int(input())
b = int(input())

if a < b:
    print('А меньше В')
elif a=b:
    print('А равно В')
else:
    print('А больше В')
```

## Для самостоятельного решения:

1. Ввести с клавиатуры три числа. Возвести в квадрат те из них, значения которых неотрицательны, и в четвертую степень — отрицательные.
2. Даны два угла треугольника (в градусах). Определить, существует ли такой треугольник. Если да, то будет ли он прямоугольным.