

МБОУ СОШ №18 имени Э.Д. Потапова

# Функции в языке программирования QBasic

## 9 класс

Арифметические, строковые и  
логические выражения

Учитель Зацепина Е.М.

Мичуринск- наукоград

# Повторение

1. Какие вы знаете типы выражений?
2. Какие операции производятся над арифметическими выражениями?
3. Какая операция называется конкатенацией?
4. Какие значения могут принимать логические выражения?

# Введение в язык программирования QBasic

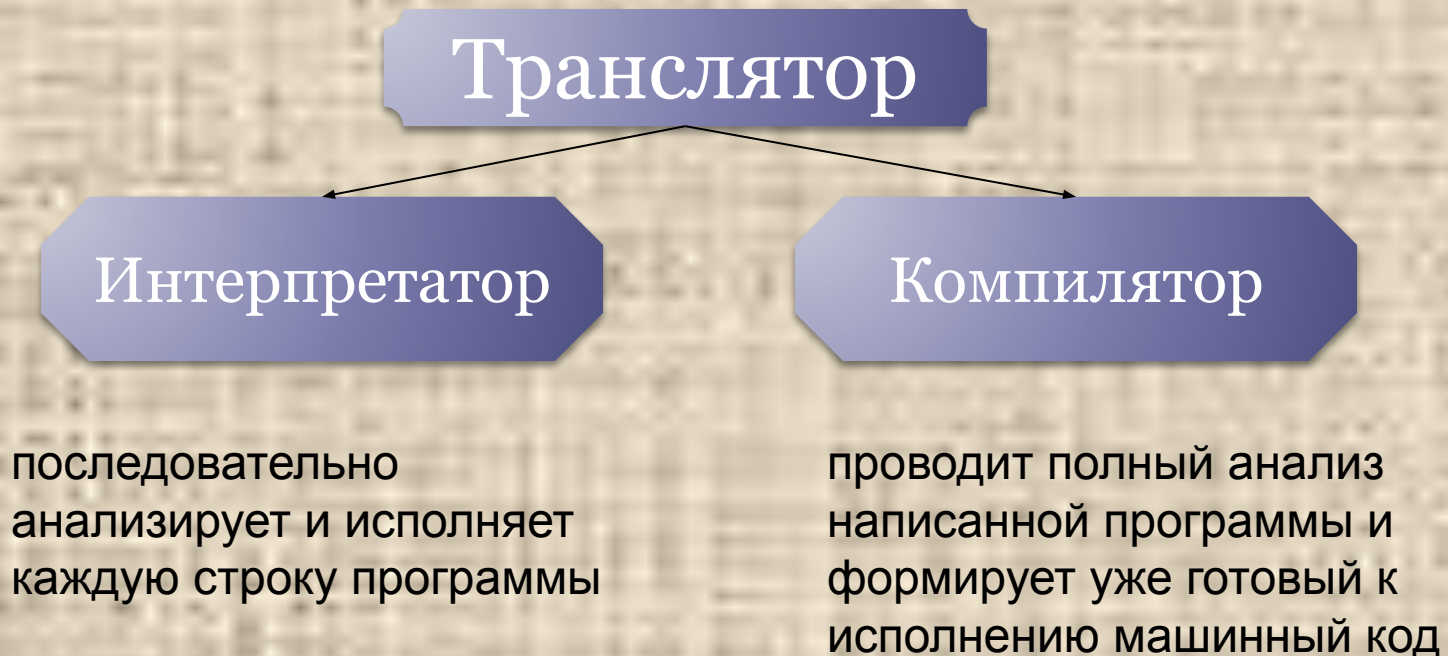
Введение в программирование

# Этапы создания программы

Для представления алгоритма в виде, понятном компьютеру, служат **языки программирования**. Сначала разрабатывается **алгоритм действий**, а потом он записывается на одном из таких языков. В итоге получается **текст программы** - полное, законченное и детальное описание алгоритма на языке программирования. Затем этот текст программы специальными служебными приложениями, которые называются *трансляторами*, либо переводится в **машинный код** (язык нулей и единиц), либо исполняется.

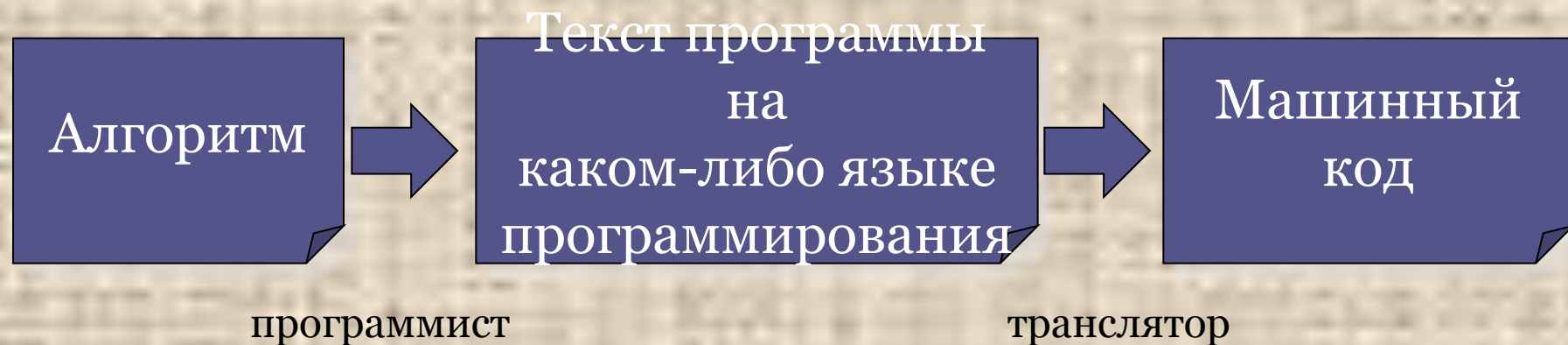
# Транслятор

Транслятор — средство для преобразования текстов из одного языка, понятного человеку, в другой язык, понятный компьютеру.





# Этапы создания программы



Очистить экран  
Ввести a  
Вычислить площадь  
Вывести результат

```
CLS  
INPUT a  
s=a*a  
PRINT s  
END
```

```
001011101010  
010101100101  
010111010110  
101101111010
```

начало

Очистить  
экран

# Языки программирования

Языки программирования - **искусственные** языки. От **естественных** они отличаются ограниченным числом "слов", значение которых понятно транслятору, и очень строгими правилами записи команд (операторов).

Однако следует иметь в виду, что в языках программирования не допускаются даже незначительные (с точки зрения человека) нарушения в правилах написания программ. Ведь даже маленькая неточность может вызвать невыполнимость программы или трудно уловимую ошибку.

# Из истории языков программирования

На заре компьютерной эры, в **50-е** годы XX века, программы писались на машинном языке и представляли собой очень длинные последовательности нулей и единиц. Составление и отладка таких программ было чрезвычайно трудоемким делом.



# Из истории языков программирования

В **60—70-е** годы для облегчения труда программистов начали создаваться языки программирования высокого уровня, формальные языки, кодирующие алгоритмы в привычном для человека виде (в виде предложений). Такие языки программирования строились на основе использования определенного алфавита и строгих правил построения предложений (синтаксиса).

# Из истории языков программирования

Наиболее широко распространенным типом языков программирования высокого уровня являются **процедурные языки**. В таких языках широко используются управляющие конструкции (операторы), которые позволяют закодировать различные алгоритмические структуры (линейную, ветвление, цикл).

Одним из первых процедурных языков программирования был известный всем **Бейсик (Basic)**, созданный в 1964 году. В течение последующего времени Бейсик развивался, появлялись его различные версии (MSX-Basic, Бейсик-Агат, **QBasic** и др.). Другим широко распространенным языком программирования алгоритмического типа является **Pascal**.

Basic (Бейсик) создавался в 60-х годах в качестве учебного языка и очень прост в изучении. По популярности занимает первое место в мире.

# Функции в языке программирования QBasic

*Функции могут быть различных типов:*

- 1) Математические (значениями как аргументов, так и функций являются числа);
- 2) Строковые (строками являются либо аргументы, либо возвращаемые функциями значения);
- 3) Ввода и вывода;
- 4) Даты и времени и т.д.

# Арифметические операции на языке Basic

Операция	Обозначение	Пример	Результат
Сложение	+	2+5	7
Вычитание	-	10-8	2
Умножение	*	3*4	12
Деление	/	15/3	5
		15/4	3.75
Целочисленное деление	\	15\4	3
Возведение в степень	^	2^3	8
Остаток от деления	MOD	13 MOD 5	3



# Математические функции на языке Basic

Корень	SQR(X)
Модуль числа	ABS(X)
Синус	SIN(X)
Косинус	COS(X)
Тангенс	TAN(X)
Целая часть числа	INT(X)
Натуральный логарифм	LOG(X)



# Операторы

В языках программирования высокого уровня программа - это перечень действий.

Исполнение этих действий определено в самом языке программирования, и их принято называть **операторами**.

# Некоторые операторы языка QBasic

**REM** – оператор комментария.

Все что следует после этого оператора до конца строки игнорируется компилятором и предназначено исключительно для человека. Т.е. здесь можно писать что угодно.

*пример:*

**REM Это комментарий**

*можно и так:*

**Это тоже комментарий**

# Некоторые операторы языка QBasic

**CLS** - очистить экран.

Вся информация, которая была на экране, стирается.

*пример:*

**CLS**

# Некоторые операторы языка QBasic

**PRINT** (вывод, печать) – оператор вывода.  
Выводит информацию на экран.

*пример:*

```
PRINT "Привет! Меня зовут Саша."
```

```
PRINT "Привет! ", " Меня зовут Саша."
```

```
PRINT "Площадь квадрата = ",s
```



# Некоторые операторы языка QBasic

**INPUT** (ввод) – оператор ввода.

Используется для передачи в программу каких-либо значений.

*пример:*

**INPUT a**

**INPUT "Введите число a: ", a**



# Некоторые операторы языка QBasic

**END** – оператор конца программы.

*пример:* **END**

**IF** - если, **THEN** - тогда, **ELSE** - иначе

# Некоторые операторы языка QBasic

**DIM** – оператор описания типа переменной.

*Пример:*

**DIM a, b, chislo1 AS INTEGER**

*Integer – целые числа от -32768 до 32768.*

Для задания значения переменной служит **оператор присваивания**. Он записывается так:

**LET** переменная = значение

*или просто:* переменная = значение

*Пример:*

**LET a = 3**

**chislo1 = 15**

# Практическая часть: *Запись математических выражений*

$$I = \frac{U}{R}$$

$$I=U/R$$

$$T = 2\pi \sqrt{\frac{l}{g}}$$

$$T=2*3.14*SQR(L/G)$$

$$s = v_0 t + \frac{at^2}{2}$$

$$s=v0*t+(a*t^2)/2$$

# Программа для вычисления перемещения

```
REM ВЫЧИСЛЕНИЕ ПЕРЕМЕЩЕНИЯ
CLS
INPUT "ВВЕДИТЕ НАЧ. СКОРОСТЬ ", vo
INPUT "ВВЕДИТЕ УСКОРЕНИЕ ", a
INPUT "ВВЕДИТЕ ВРЕМЯ ДВИЖЕНИЯ ", t

$$s = vo * t + (a * t^2) / 2$$

PRINT "ПЕРЕМЕЩЕНИЕ РАВНО: ", s
END
```

# Программа для вычисления перемещения

```
REM ВЫЧИСЛЕНИЕ ПЕРЕМЕЩЕНИЯ  
CLS
```

```
DIM vo, a, t, s AS SINGLE
```

```
INPUT "ВВЕДИТЕ НАЧ. СКОРОСТЬ ", vo
```

```
INPUT "ВВЕДИТЕ УСКОРЕНИЕ ", a
```

```
INPUT "ВВЕДИТЕ ВРЕМЯ ДВИЖЕНИЯ ", t
```

```
s=vo*t+(a*t^2)/2
```

```
PRINT "ПЕРЕМЕЩЕНИЕ РАВНО: ",s
```

```
END
```



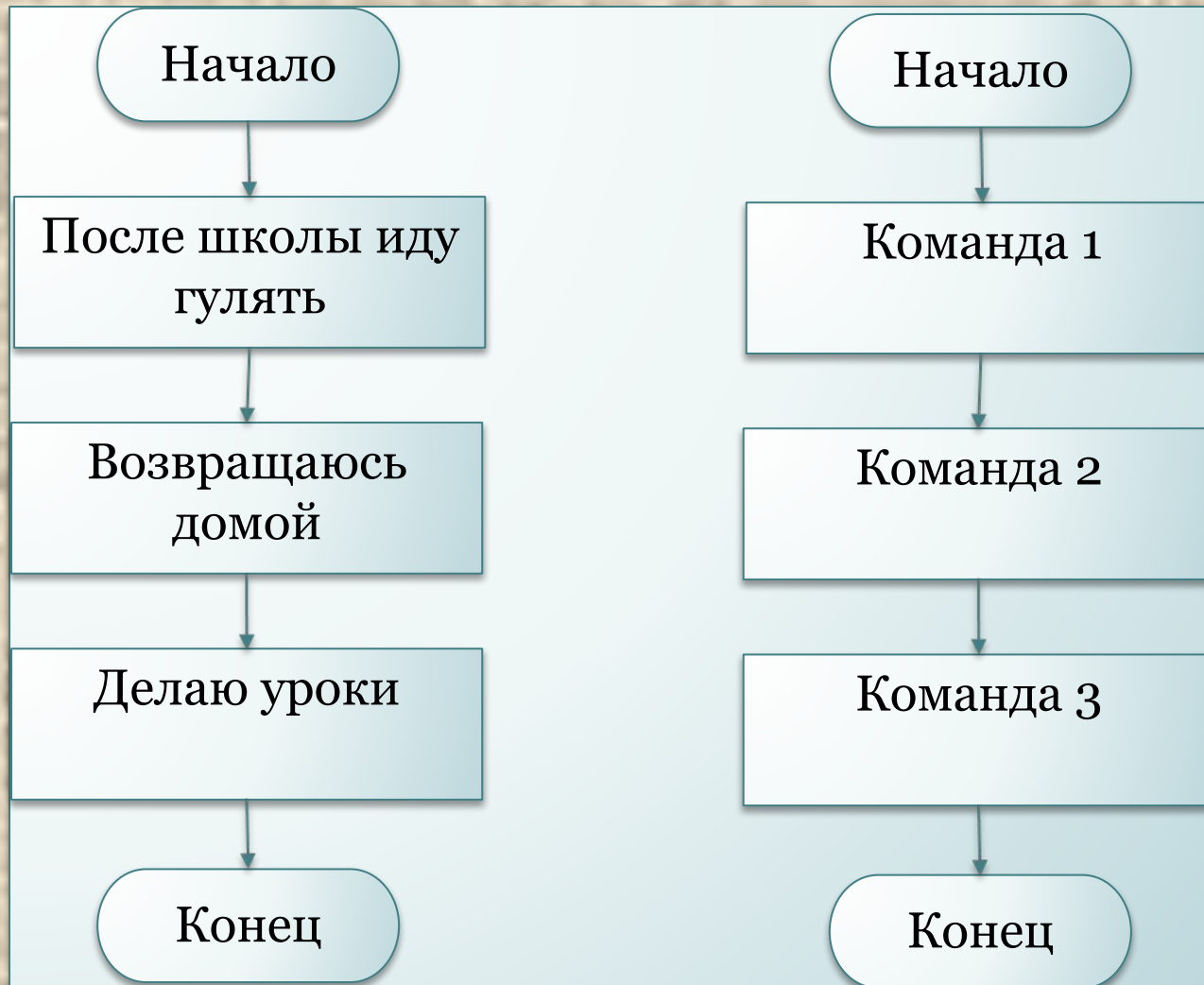
# Линейная структура программы

**Введение в программирование**

# Линейная структура программы

Программа имеет линейную структуру, если все операторы (команды) выполняются последовательно друг за другом.

# Линейная структура программы



CLS

PRINT "Привет!"

# Площадь прямоугольника

```
CLS
```

```
a=3
```

```
b=4
```

```
s=a*b
```

```
PRINT "Площадь прямоугольника равна ",s
```

```
END
```



# Площадь прямоугольника

**CLS**

**INPUT "Введите длину: ", a**

**INPUT "Введите ширину: ", b**

**s=a\*b**

**PRINT "Площадь прямоугольника равна ",s**

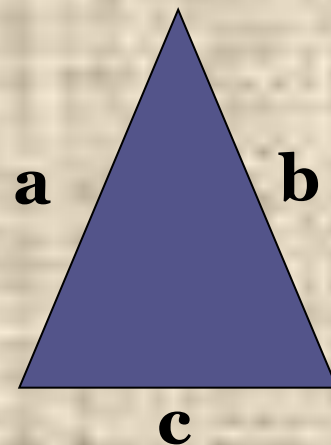
**END**

# Задача:

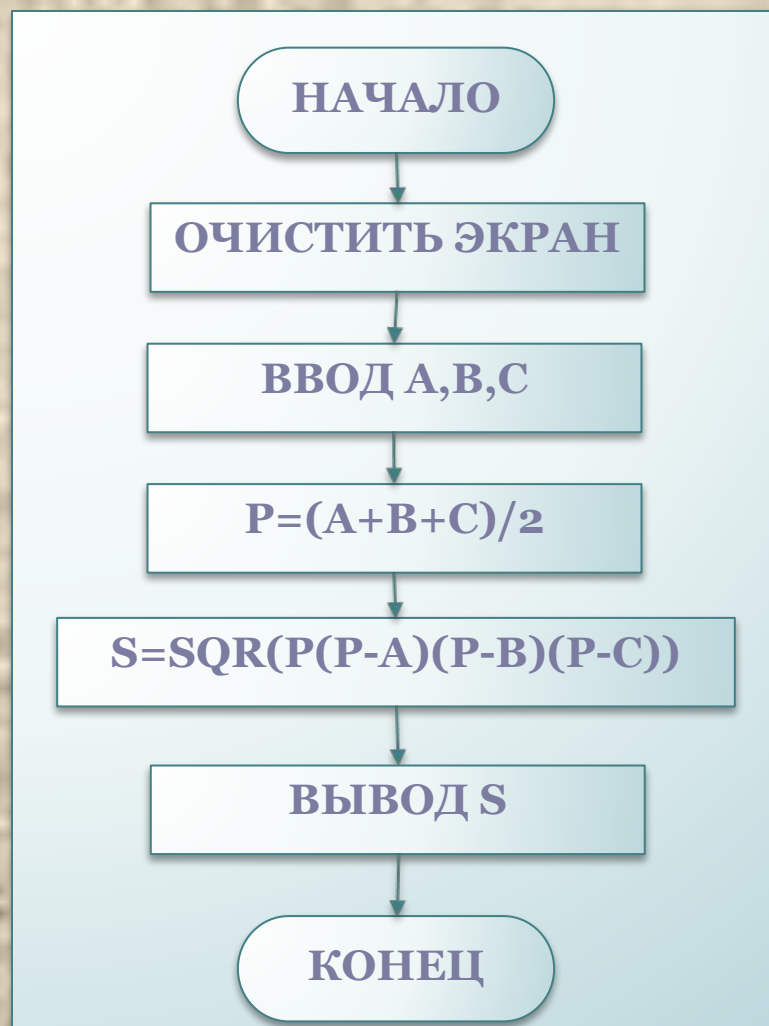
Рассмотрим для примера программу вычисления площади треугольника по 3 сторонам.

$$s = \sqrt{p(p-a)(p-b)(p-c)}$$

$$p = \frac{1}{2}(a+b+c)$$



# Составим алгоритм и запишем его в виде блок-схемы:



Программа может иметь вид: *вычисление площади треугольника по 3 сторонам.*

**REM Вычисление .....по 3**

**.....**

**C....**

**INPUT "Введите .....: ", a**

**....."Введите .....: ", b**

**....."Введите .....**

**$p = (a + b + c) / 2$**

**$s = \text{SQR}(p(p - a)(p - b)(p - c))$**

**PRINT " .....: ", s**

**END**



**Задание:** написать программу  
вычисления выражения

$$y = \sqrt{x} + \sqrt{\frac{x + 2}{5}}$$

$$y = \text{SQR}(x) + \text{SQR}((x + 2) / 5)$$

# Домашнее задание

1. Уметь составлять блок-схемы и программы, имеющие линейную структуру, записывать математические выражения на QBasic.
2. Составить линейную блок-схему и написать программу вычисления длины окружности и площади круга.

# Разветвляющаяся структура программы

**Введение в программирование**

# Вопросы:

- Что такое транслятор?
- Какие языки программирования вы знаете?
- Что такое оператор?
- Какие операторы языка программирования Basic вы знаете?



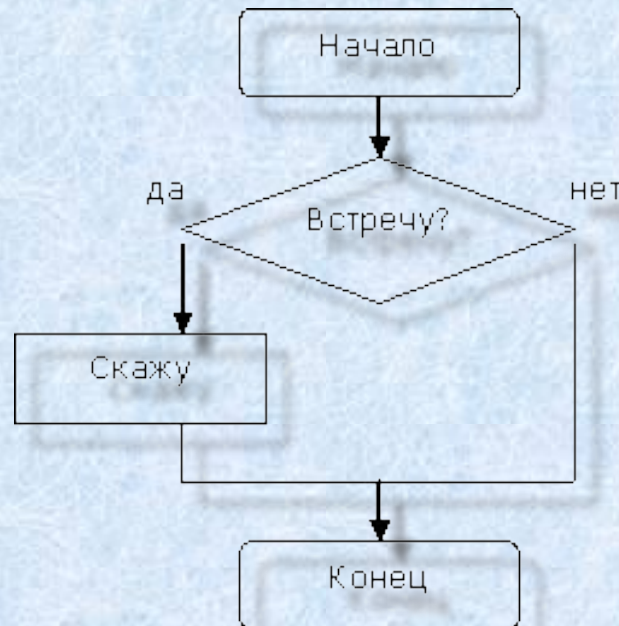
# Разветвляющий алгоритм -

Алгоритм, в котором в зависимости от условия выполняется либо одна, либо другая последовательность действий:

- ☐ Если в комнате темно, тогда надо включить свет.
- ☐ Если пойдет дождь, тогда надо взять зонтик, иначе, зонтик не брать.

# Разветвляющийся алгоритм

Во многих случаях требуется, чтобы при одних условиях выполнялась одна последовательность действий, а при других – другая.



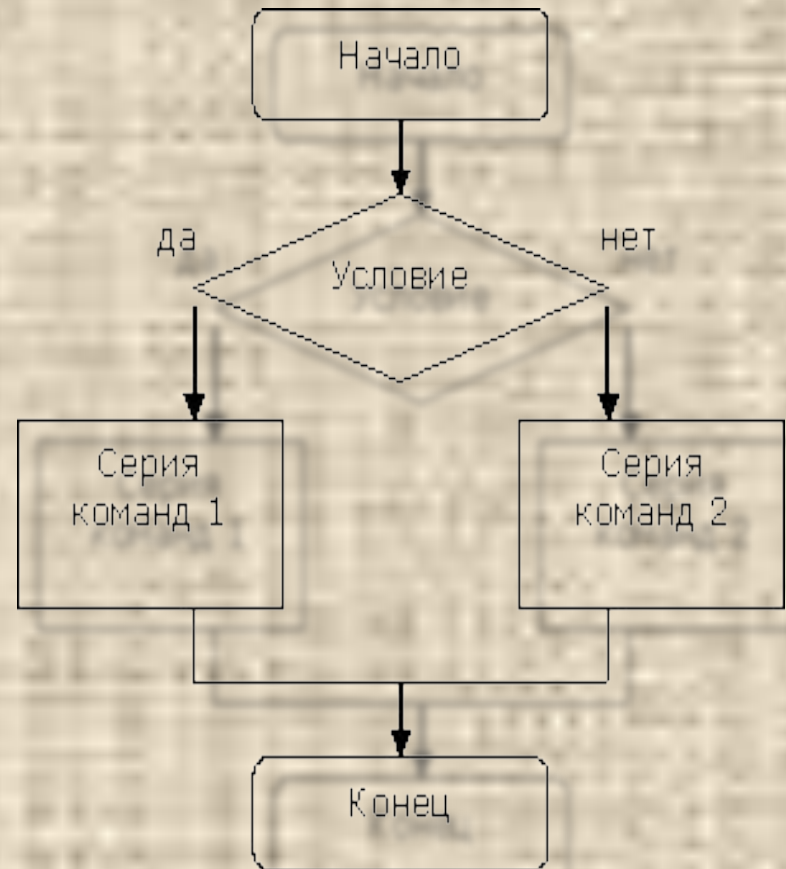
# Рассмотрим запись условного оператора на языке Basic

Простая форма оператора  
выглядит следующим  
образом:

**IF <УСЛОВИЕ> THEN**  
**<ОПЕРАТОР>**

или

**IF УСЛОВИЕ>**  
**<ОПЕРАТОР 1>**  
**<ОПЕРАТОР 2>**  
**... <ОПЕРАТОР N> END IF**



# Первичные конструкции на языке QBasic

*Условия* - еще один тип логических выражений.

В них используются следующие операторы сравнения:

Конструкции	Обозначение	Пример
Больше	>	5 > 3
Меньше	<	5 < 3
Равно	=	3 = 3
Больше или равно Меньше или равно	>=	x >= 0
	<=	x <= 0
Не равно	<>	y <> 0



# Первичные конструкции на языке QBasic

Справа и слева от знака сравнения должны стоять величины, относящиеся к одному типу. В результате сравнения получается логическая величина, имеющее значение *ИСТИНА (TRUE)* или *ЛОЖЬ (FALSE)*.

## Пример:

$5 < 7$  - ИСТИНА;  $8 = 12$  -ЛОЖЬ (проверяем равно ли 8 12, именно проверяем, а не утверждаем, что  $8 = 12$ );

**Пример:** *определить, является ли  
треугольник со сторонами  $a, b, c$  -  
равносторонним*

Программа может иметь вид:

**1 вариант**

CLS

REM Равносторонний ли треугольник

INPUT «Введите сторону A:»; A

INPUT «Введите сторону B:»; B

INPUT «Введите сторону C:»; C

IF A=B=C THEN PRINT «Этот треугольник  
равносторонний»

IF A<>B<>C THEN PRINT «Этот треугольник  
равносторонним не является»

END

# Пример:

## 2 вариант

CLS

REM Равносторонний ли треугольник

INPUT «Введите стороны треугольника через  
запятую- A, B, C:»; A, B, C

IF A=B=C THEN PRINT «Этот треугольник  
равносторонний» ELSE PRINT «Этот  
треугольник равносторонним не является»

END

**Пример:** проверить, равно ли введенное число некоторому значению, и в случае равенства выдать на экран сообщение о равенстве чисел.

```
REM сравнить число со каким-то значением  
INPUT "Введите а", а  
IF а=7 THEN PRINT "Числа равны"  
END
```

После запуска программы проверяется равно ли введенное значение семи или нет. Если равно, то на экран выводится сообщение “Числа равны”



# Повторение: запишите выражения на языке QBasic

$$y = \frac{\sqrt{2x+1}}{3x}$$

$$y = \frac{\sqrt{\sin x}}{\cos x}$$

$$y = \frac{1}{x} + \frac{x^2}{x+1}$$

**Пример:** *определить, является ли треугольник со сторонами  $a, b, c$  - равнобедренным*

.....

.....Равнобедренный ли треугольник

..... «Введите .....-..., ..., ... :»; ..., ..., ..

..... «Этот треугольник .....»

..... PRINT «Этот треугольник .....»

END

*Задание: написать программу с условием вычисления выражения*

$$y = \frac{\sqrt{\sin x}}{\cos x}$$

☐  $x=0; y=0$

☐  $x=1; y=1.697785$

**Пример:** Решение квадратного уравнения.  
Решение квадратного уравнения зависит от значения дискриминанта.

```
REM Решение квадратного уравнения
INPUT "Введите коэффициент a: ", a
INPUT "Введите коэффициент b: ", b
INPUT "Введите коэффициент c: ", c
d=b*b-4*a*c
IF d<0 THEN PRINT "Корней нет"ELSE
END IF
IF d=0 THEN x=-b/(2*a) PRINT "корень уравнения: ", x
END IF
ELSE x1=(-b-SQR(d))/(2*a) x2=(-b+SQR(d))/(2*a)
PRINT "корни уравнения: ", x1, x2
END
```

*Задание: написать программу с условием вычисления выражения*

$$y = \frac{x^3 - 2}{x}$$

☐  $x=1; y=-1$

☐  $x=2; y=3$



# Задание: составить линейную и разветвляющуюся программы

1. Теоремы Пифагора:

$$a^2 + b^2 = c^2$$

2. Площади произвольного четырехугольника ABCD, равной половине произведения диагоналей и синуса угла между ними:

$$S_{ABCD} = \frac{1}{2} AC \cdot BD \cdot \sin \beta$$

# Тестирование

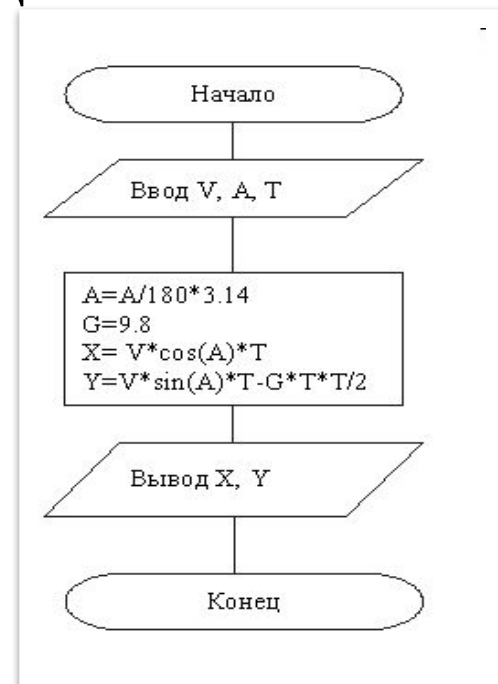
1. Понятное и точное предписание исполнителю при заданных начальных данных выполнить конечную последовательность команд, приводящую к искомому результату, называется

- 1) моделью
- 2) системой
- 3) алгоритмом
- 4) технологией

# Тестирование

2. Алгоритмическая конструкция какого типа изображена на блок-схеме?

- 1) линейная
- 2) циклическая
- 3) разветвляющаяся
- 4) вспомогательная



# Тестирование

## **3. Алгоритм включает в себя ветвление, если**

1. он составлен так, что его выполнение предполагает многократное повторение одних и тех же действий;
2. ход его выполнения зависит от истинности тех или иных условий;
3. его команды выполняются в порядке их естественного следования друг за другом независимо от каких-либо условий;
4. он представим в табличной форме;
5. он включает в себя вспомогательный алгоритм.



# Тестирование

4. Выберите верное представление арифметического выражения на алгоритмическом языке:

1.  $x + 3y / 5xy$

2.  $x + 3*y / 5*x*y$

3.  $(x + 3y) / 5xy$

4.  $(x + 3*y) / (5*x*y)$

5.  $x + 3*y / (5*x*y)$

# Тестирование

5. Алгоритм, записанный на «понятном» компьютеру языке, называется

1. исполнителем алгоритмов;
2. программой;
3. текстовкой;
4. протоколом алгоритма.

# Правильный ответ на тест

1) 3

2) 1

3) 2

4) 4

5) 2

# Задание:

Написать программу решения следующей шуточной задачи:

- 1) ЭВМ спрашивает: «Сколько Вам лет?».
- 2) Если ответ больше 100, то отвечает: «Вы хорошо сохранились!».
- 3) В противном случае желает: «Лет до 100 расти Вам без старости!».

# Домашнее задание

1. Уметь составлять блок-схемы и программы, имеющие разветвляющуюся структуру, записывать математические выражения на QBasic.
2. Написать программу вычисления площади треугольника по высоте и основанию.



# Алгоритмическая структура «выбор»

**Введение в программирование**

# Структура «выбор»

Структура IF... позволяет выбрать между двумя вариантами.

Если требуется осуществить выбор между большим числом вариантов, то это можно организовать используя лишь структуру IF... Но можно (что чаще проще) и с помощью структуры "Выбор".

# Структура «выбор»

Эта структура имеет вид:

```
SELECT CASE <Выражение>  
CASE <условие 1> <серия 1>  
CASE <условие 2> <серия 2> ...  
CASE ELSE <серия иначе>  
END SELECT
```

Выражение, заданное после ключевых слов **SELECT CASE**, сравнивается с определенными значениями - условиями и если они истинны, то выполняется соответствующая серия команд. Если не одно условие не истинно, то выполняется серия команд между *CASE ELSE* и *END SELECT*.

# **Пример:** выдать словесное значение числа

```
REM Преобразование чисел в слова  
INPUT "Введите число", a  
SELECT CASE a  
CASE 1 PRINT "один"  
CASE 2 PRINT "два"  
CASE 3 PRINT "три" ...  
CASE 10 PRINT "десять"  
END SELECT  
CASE ELSE PRINT "это число не могу перевести"  
END
```

В данном примере введенное число сравнивается с числами от 1 до 10 и если наше число равно одному из этих чисел, то на экран выводится словесное значение числа. Если это не так на экран выводится сообщение: "это число не могу перевести".

# **Задание:**

**Написать программу , которая позволяет  
выставлять оценку за работу в  
зависимости от количества сделанных  
ошибок.**