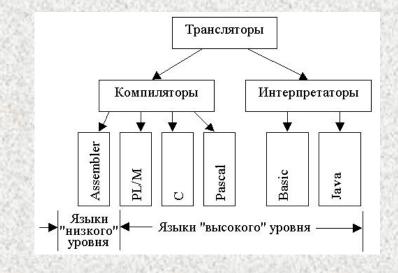
# Язык программирован ия QBasic



## Введение

**Язык программирования** - это совокупность средств и правил представления алгоритма в виде, понятном ЭВМ

Бейсик – один из языков программирования высокого уровня.

Разработан первый Бейсик в **1964** г. сотрудниками Дартмутского колледжа Дж. Кемени и Т. Курцем. Интересно происхождение названия языка. В прошлом веке один

английский миссионер выделил из английского языка триста наиболее

употребительных слов, назвал их Basic English и стал обучать туземцев. Опыт

оказался весьма успешным, и контакты с аборигенами значительно упростились.

Создатели языка Бейсик стремились достигнуть того же эффекта — облегчить

понимание между "туземцами" — начинающими программистами, и компьютерами.

Аббревиатура BASIC так и расшифровывается — "Beginner's All purpose Symbolic Instruction Code", что в переводе значит "многоцелевой язык

символических команд для начинающих ...

Большое достоинство Бейсика, из-за которого его изучение продолжается в школах и поныне — это возможность создавать диалоговые программы.

#### Арифметические операции

Операция	Обозначение	Пример	Результат
Сложение	+	2+5	7
Вычитание		10-8	2
Умножение	*	3*4	12
Деление	I	15/3	5
		15/4	3.75
Целочисленное деление	1	15\4	3
Возведение в степень	Λ	2^3	8
Остаток от деления	MOD	13 MOD 5	3

### Операции отношений

- > больше
- < меньше
- = равно
- <> не равно
- >= больше либо равно
- <= меньше либо равно

## Математические функции

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)	

#### Запись математических

$I=\frac{U}{R}$ выражений $I=U/R$		
$T = 2\pi \sqrt{\frac{l}{g}}$	T=2*3.14*SQR(L/G)	
$s = \upsilon_0 t + \frac{at^2}{2}$	s=v0*t+(a*t^2)/2	
2x <sup>3</sup>	2*X^3	
$\frac{3+4}{8-6}$	(3+4)/(8-6)	

### Переменные и константы в

Переменная - это имя ячейки в оперативной памяти компьютера, в котором в каждый момент времени может храниться только одно значение. Имена переменных могут иметь длину до 40 символов, начинаться с буквы, за которой могут следовать любые символы. Примеры правильных имен переменных: A, B, Z, IVAN; IVAN3, S1, T234, LOVE7, R6N8F43. Переменные различаются по типу хранимой в них информации:

числовой тип – для хранения различных чисел;

строковый тип – для хранения символов и строк (в таком случае к имени переменной добавляется обязательный символ \$, например, X\$ или QUIKE3\$).

Постоянная (константа) - величина, записанная в виде конкретного числа.

целые - 6; -18 вещественные 6.2; -18.1564; Строковые (символьные) "Петя", "X="

- 1. дробная часть отделяется от целой точкой, а не запятой (нельзя 3,14, надо 3.14);
- 2. в записи десятичной дроби ноль, стоящий перед точкой опускается (вместо 0.123 можно .123);
- 3. необходимо соблюдать приоритет выполнения действий
  - □ действия в скобках
  - □ вычисление функций
  - □ возведение в степень
  - □ умножение и деление
  - □ сложение и вычитание

Для печати больших и малых чисел используется запись с плавающей точкой.

Например: 1.234Е-05 равно 0.00001234

1.234Е05 равно 123400

## Оператор присваивания LET — задает значение переменной

LET A=1 - в ячейку с адресом А запишется значение 1 (читается так — переменной А присвоено значение 1) LET можно опускать и в программе писать A=1 Левая часть оператора - имя переменной, правая часть может быть константой или арифметическим выражением: LET C=A+B. Предварительно значения переменным А и В должны быть присвоены оператором LET.

Оператор вида LET X=X+1 добавляет 1 к текущему значению переменной X и увеличивает значение переменной на единицу. Новое значение запишется в эту же ячейку с адресом X

## Оператор ввода данных INPUT позволяет вводить данные в ходе выполнения

INPUT «Введите 3 значения переменных»; А,В,С

Встретив в программе INPUT компьютер приостанавливает работу программы, выводит на экран «?» и ждет от пользователя ввода 3-х числовых значений с клавиатуры через запятую.

? 5, -10, 456.87

Данные, введенные пользователем, будут последовательно присвоены переменным A,B,C и работа программы продолжится.

<u>ВАЖНО!</u> Пользователь должен ввести столько числовых значений, сколько переменных указано в INPUT. Если количество переменных больше или меньше количества значений, то на экране появится информация REDO FROM START (повторите сначала).

Не разрешается вводить арифметические выражения (875+7263), запятые, имена переменных.

**Оператор вывода результата PRINT** выводит результат работы программы на экран

PRINT «Значение C=»; С

PRINT без дополнительных данных выводит пустую строку. Это удобный способ выделить заголовок таблицы

### Средства автоматизации -

это операторы, которые изменяют порядок передачи управления в программе в зависимости от условий.

#### Оператор GO TO

Оператор безусловного перехода. Передает управление другому фрагменту программы по номеру строки (метке оператора) в прямом и обратном направлении.

#### Разберем программу

10 LET X=0

20 LET X=X+1

30 PRINT X

40 GO TO 20

**50 END** 

#### Оператор условного перехода IF ... THEN

изменяет порядок выполнения программы при определенных условиях

Условия, которые проверя **етимератюрить наприотребления** оператора IF...

- 1) > больше
- 2) < меньше
- 3) = равно
- 4) <> не равно
- 5) >= больше либо равно
- 6) <= меньше либо равно

- THEN
- 1) IF I=33 THEN GO TO 100
- 2) IF A+B < 16 THEN GO TO 100
- 3) IF A-2=C THEN GO TO 100
- 4) IF A>=C THEN GO TO 100
- 5) IF A<0 THEN GO TO 100

**Оператор GO TO можно опускать** IF A<0 THEN GO TO 100

#### Логические операции AND (И) OR (ИЛИ) в IF ... THEN

используются для сравнения двух и более отношений

#### Примеры

IF X=12 AND Y<0 THEN 100 (если одновременно X=12 и Y<0, то управление в программе передается строке 100)

IF X=12 OR Y<0 THEN 100 (если верно хотя бы одно из этих соотношений, то управление в программе передается строке 100)

### Оператор IF ... THEN ... ELSE

Параметр ELSE указывает что делать в том случае, если проверяемое условие не выполняется.

IF X=45 THEN 100 ELSE PRINT "Неверно"

Если X=45, то управление переходит на строчку 100, иначе (X<>45)

### Операторы FOR ... NEXT

эта пара операторов образует в программе циклы и управляет ими

FOR A=1 TO 10 STEP 1 приращения 1

(читать так : ДЛЯ А от единицы ДО 10 с шагом

ВЬ

выполнять действия и перейти к СЛЕДУЮЩЕМУ

значению А)

тело цикла (действия)

Если шаг приращения = 1, то STEP можно опускать

**NEXT A** 

NEXT увеличивает значение счетчика и определяет, не привысило ли оно установленной границы. Пока не достигнуто конечное значение переменной, действия в цикле повторяются.

Разберем программу

FOR X=1 TO 5
PRINT X
NEXT X

Оператор FOR может содержать в правой части переменные и арифметические выражения

#### Примеры:

FOR X=100 TO 1000 STEP 2

# Пример программы, реализующей линейный алгоритм

#### Нахождение периметра треугольника:

- 1 CLS
- 2 INPUT A,B,C
- 3 P=A+B+C
- 4 PRINT P
- 5 END

Для удобства пояснения пронумеруем строки программы и поясним каждую.

Внимание! При работе в оболочке QBasic строки нумеровать не обязательно.

#### Поясним работу программы:

- 1. Производится очистка экрана.
- Оператор ввода INPUT приостанавливает действие программы, выводит на экран «?», ожидая от пользователя ввода 3-х значений переменных A, B, C
- 3. Выполняется сложение 3-х введённых чисел и результат записывается в ячейку с именем Р.
- 4. Оператор вывода PRINT выводит значение переменной P на экран монитора.
- Оператор END заканчивает работу программы.

# Пример программы, реализующей разветвляющийся алгоритм

Дан фрагмент программы:

Скажите, что будет выведено на экран, если по запросу введено:

- а) число 5;
- б) число 12;
- в) число10