

ОСНОВЫ ЯЗЫКА QBasic

Часть 1

Введение в QBASIC

Язык программирования BASIC был разработан профессорами Дартмутского Колледжа (США) в 1963 году. Этот язык изначально задумывался как язык для людей, не являющихся профессиональными программистами, но имеющими дело с компьютерами.

Слово BASIC является аббревиатурой (Beginner's All-purpose Symbolic Instruction Code), что в переводе означает Многоцелевой Символический Код Для начинающих. С тех пор этот язык претерпел много изменений и усовершенствований, но остался одним из наиболее подходящих языков для изучения основ

Нынешний QBASIC – это Quick Basic, предложенный фирмой Microsoft.

Существуют новые версии этого языка, а так же Visual Basic, являющийся объектно-ориентированным языком. Еще один вариант Visual Basic – это VBA (Visual Basic for Application), то есть Visual Basic для приложений. VBA встроен в программные продукты Microsoft, тем самым расширяя их возможности.

При проектировании языка были сформулированы восемь требований к нему:

- ✓ простота в использовании для начинающих;
- ✓ общность назначения (отсутствие специализации);
- ✓ возможность расширения функциональности средствами, доступными программистам;
- ✓ интерактивность;
- ✓ четкие и понятные сообщения об ошибках;
- ✓ высокая скорость работы на небольших программах;
- ✓ отсутствие необходимости понимания работы аппаратного обеспечения для написания программ;
- ✓ эффективное посредничество между

ОГЛАВЛЕНИЕ

- ✓ Алфавит QBASIC
- ✓ Типы данных в QBASIC
- ✓ Переменные
- ✓ Константы
- ✓ Простейшая программа
- ✓ Переменные и оператор присваивания
- ✓ Ввод данных с клавиатуры
- ✓ Вывод данных на экран. Оператор PRINT
- ✓ Задания

Алфавит языка QBasic

содержит:

- ✓ Все латинские прописные и строчные буквы;
- ✓ Арабские цифры 0-9;
- ✓ Знаки арифметических операций:

^ * / + - \

- ✓ Знаки операций отношения:

= > < >= <= <>

<> - это знак **больше/меньше**

Разделители и прочие

СИМВОЛЫ:

- ✓ ■ – точка;
- ✓ , - запятая;
- ✓ ; - точка с запятой;
- ✓ : - двоеточие;
- ✓ - пробел;
- ✓ ! – признак вещественной величины;
- ✓ # - признак вещественной величины двойной точности;
- ✓ % - признак целой величины;

Разделители и прочие символы (продолжение):

- ✓ \$ - признак текстовой величины;
- ✓ () – круглые скобки;
- ✓ “ – кавычки;
- ✓ ‘ – апостроф.
- ✓ Используются также буквы русского алфавита, но в текстовых константах или комментариях.

Типы данных в QBASIC

Данные в программировании –

**это та информация, которая
обрабатывается в программе.**

**Данные могут быть числовые или
текстовые. Так же данные могут
быть константами или**

переменными

Таблица типов

данных

Тип	Символ	Содержание
INTEGER	%	целые числа в интервале от -32 768 до +32 768
LONG	&	целые числа (длинные целые) в интервале от -2 147483648 до +2 147483648
SINGLE	!	числа с плавающей запятой (вещественные) в интервале от $\pm 8,43 \cdot 10^{-37}$ до $\pm 3,37 \cdot 10^{38}$
DOUBLE	#	числа с плавающей запятой (вещественные) в интервале от $\pm 4,19 \cdot 10^{-307}$ до $\pm 1,67 \cdot 10^{308}$
STRING	\$	любые тексты с максимальной длиной 32 767 знаков

Например:

- ✓ **102!** – константа числовая, вещественная, обычной точности;
- ✓ **9566.46#** - константа числовая, вещественная, двойной точности;
- ✓ **4321%** - константа числовая, целая;
- ✓ **“План на неделю”** – константа текстовая. Заключается в кавычки.

Помните!

Разделителем целой и дробной частей
является точка!

45.345#

Переменные

Переменная – это область памяти, (размер области зависит от типа данных), в которой хранится значение, присвоенное переменной.

Каждая переменная имеет имя (идентификатор).

Имя переменной в **QBASIC** может содержать до 40 символов (букв латинского алфавита, цифр, специальных символов), однако всегда должно начинаться с буквы.

НАПРИМЕР:

A10

Создавая новые имена для своих переменных вы можете использовать как **строчные (маленькие)**, так и **заглавные (большие) буквы**, но QBASIC не различает **большие и маленькие буквы**

QBASIC поддерживает **5 типов** данных.

**Тип переменной распознается по
специальному символу, стоящему в
конце имени переменной.**

Например:

**Знак \$ в конце имени переменной
говорит о том, что переменная
относится к текстовой группе.**

MyName\$="Павел".

**Знаки % или &, стоящие в конце
имени переменной, говорят о том,
что это переменная целого типа
или длинного целого типа.**

SN% или NS&.

**Если имя переменной не имеет на
конце специального символа, то
по умолчанию такая переменная
имеет действительный
(вещественный) тип
(т. е. является числом с дробной
частью обычной точности).
То есть $A!=2.3$ и $A=2.3$ задают
значение одной и той же
переменной.**

**Если же мы хотим
обозначить переменную
как числовую
двойной точности,
то необходимо
в конце имени переменной
приписать знак «решетка».**

A#

Переменные

Символьные

Целые (A%)

Длинные целые (A&)

Вещественные обычной
точности (A!, A)

Вещественные двойной
точности (A#)

Примеры правильных и неправильных имен переменных:

AVc3e – правильно,

так как состоит из латинских букв и цифр и начинается с буквы ;

4AVCe – неправильно,

так как начинается с цифры;

Имя23 – неправильно,

так как включает в себя русские буквы;

Примеры правильных и неправильных имен переменных (продолжение):

- A B C D** – неправильно, так как буквы написаны через пробел;
- A&V** – неправильно, так как включает в себя знак амперсанд, являющийся признаком длинной целой величины;
- K** – правильно;
- Print** – неправильно, так как Print – это зарезервированное слово – название оператора

**Существует специальный
оператор, с помощью которого
можно задать переменным тип,
но мы рассмотрим его позже.**

Константы

Константы – числовые или строковые, не меняются в ходе выполнения программы. Значения констант записываются явно, числом или строкой символов.

Например: $A\% = 12$

$A\$ = \text{“Сумма равна”}$

$X! = 12.54$

Существует специальный оператор для описания констант.

Const имя переменной=константа

Например:

**Const G=100,
DNINED%=7**

Если переменная описана как константа с помощью оператора **Const**, то **QBASIC** не позволит ее случайно изменить.



Простейшая программа

Программа на языке QBASIC
состоит из последовательности
инструкций – команд компилятору.

Если в строке записано
несколько инструкций, то они
разделяются двоеточием.

В случае ошибки, компилятор
выдает соответствующее
сообщение и работа программы
прекращается.

Начнем с самой простой программы:

CLS

PRINT “Ура! Программа работает”

END

**Если нужно вывести на экран
несколько строчек, то оператор PRINT
можно повторить несколько раз.**

CLS

PRINT “Это моя первая программа”

PRINT “Она работает”

PRINT “УРА”

END

Задание 1: напишите программу, которая выводит на экран следующий текст:

**В лесу родилась елочка,
В лесу она росла.**

Задание 2: что выведет на экран следующая программа?

PRINT “Раз, два, три, четыре, пять”

END

PRINT “Вышел зайчик погулять”

Переменные и оператор присваивания

Пусть, например, нужно выполнить вычисления по формуле:

$$45(15+87)-53(64+8)$$

Для этого достаточно написать такую программу:

```
PRINT 45*(15+87)-53*(64+8)  
END
```

То есть, мы просто включили нужную формулу в инструкцию PRINT

Сначала был произведен расчет по данной формуле, а затем результат вывели на экран

В следующей задаче требуется перед выполнением вычислений задать переменным значения :

Задание 3 Вычислить значение выражения $(a+b)(2a+1)(b-1)$ для $a=12$, $b=17$ и для $a=-31$, $b=8$

Здесь можно предложить такую программу:

A=12

B=17

PRINT (A+B)*(2*A+1)*(B-1)

A=-31

B=8

PRINT (A+B)*(2*A+1)*(B-1)

END

В этой программе мы впервые используем переменные.

Каждая переменная в программе обозначается своим именем. В рассматриваемом случае это **A и **B**.**

Первые строчки в нашей программе – это и есть операторы присваивания:

A=12 – в область памяти, отведенной для хранения переменной A, записываем значение 12;

B=17 - в область памяти, отведенной для хранения - переменной B записываем значение 17;

Теперь в программе везде, где встретятся имена **A и **B**, используются значение 12 и 17 до тех пор, пока они не будут изменены другими операторами присваивания.**

**В общем виде оператор присваивания
МОЖНО записать так:
переменная = арифметическое выражение**

В этой записи **переменная** – это имя какой-нибудь переменной, а арифметическое выражение – формула, по которой необходимо вычислять требуемое значение

В арифметическое выражение могут входить
числа и имена переменных,
объединенные знаками арифметических операций.
Арифметическое выражение может быть достаточно
сложным, лишь бы оно **поместилось в одной строке.**

Одно из самых важных правил записи оператора присваивания!

**В операторе присваивания имя переменной,
в которую присваивается рассчитанное
значение выражения, всегда должно стоять
слева от знака равно (“=”).**

Примеры записи оператора присваивания:

Правильно:

$$D=B*B+4*A*C$$

$$S=A+B+3+D$$

$$REZ=D-S$$

Неправильно:

$$D+B=16$$

$$S-D=A+B+3$$

$$REZ+S=D$$

Примеры формул и соответствующих им арифметических выражений

Формула

Арифметическое выражение

$$\frac{A * B}{C} + D$$

$$A * B / C + D$$

$$\frac{A + 2B - 3C}{5A + 4}$$

$$(A + 2 * B - 3 * C) / (5 * A + 4)$$

Встроенные функции языка QBASIC

ABS() абсолютная величина (модуль)

ATN() арктангенс

COS() косинус

SIN() синус

TAN() тангенс

EXP() экспонента

LOG() натуральный логарифм

SQR() квадратный корень

Для тригонометрических функций используется радианная мера угла

Примеры

Формула

*Арифметическое
выражение*

$$\sin X + \sin 2X - \cos(5X+3)$$

$$\sin(X) + \sin(2 * X) + \cos(5 * X + 3)$$

$$\sqrt{B^2 - 4AC}$$

$$\text{SQR}(B * B - 4 * A * C)$$

$$\text{tg}^2 (X + Y) + 1$$

$$\text{TAN}(X + Y)^2 + 1$$

Какое значение будет иметь переменная **A после выполнения следующих операторов?**

1)

$$A=5$$

$$B=A+4$$

$$A=A+B+A*B$$

Ответ: 59

2)

$$A=1$$

$$B=A*4$$

$$A=A-B+A*B$$

Ответ: 1

Ввод данных с клавиатуры

Как правильно написать программу, чтобы она работала с разными значениями переменных?

Эту возможность обеспечивает инструкция ввода

INPUT

Пример: Вычислите сумму двух чисел.

```
INPUT A,B
```

```
PRINT A+B
```

```
END
```

При запуске этой программы на экране появится знак вопроса. Это означает, что программа ожидает ввода с клавиатуры.

Наберем два числа на клавиатуре, отделив одно от другого запятой, и нажмем клавишу Enter, чтобы отметить конец ввода.

?5,6

Инструкция **INPUT** воспримет эти два числа и запишет первое из них в переменную **A**, а второе в переменную **B**.

Общий вид оператора INPUT

INPUT [;][приглашение,] список переменных

Список переменных – это имена переменных, перечисленные через запятую. **Приглашение** – это любой текст, заключенный в двойные кавычки.

Если приглашение отсутствует, то на экран выводится вопросительный знак (?).

Если после слова INPUT поставить точку с запятой (;), То после ввода текущая позиция не переместится на следующую строчку.

Примеры: В предыдущей программе вместо инструкции **INPUT A, B** можно написать любую из следующих инструкций:

INPUT “Введите A и B”, A, B

INPUT “A и B ->”, A, B

INPUT ; A, B

INPUT ; “sum of A and B” , A, B

Вывод данных на экран. Оператор PRINT.

Запишем общий вид оператора PRINT:

PRINT [список вывода] [;]

Список вывода – это произвольные арифметические выражения или строки текста, заключенные в двойные кавычки. При этом для арифметических выражений вычисляются и печатаются их значения, а для строк выводятся все символы, заключенные в кавычки.

Одно выражение от другого в списке можно отделять запятыми или точками с запятой. Если в конце инструкции PRINT поставить точку с запятой, то после вывода **не будет** осуществлен переход на следующую строку.

Примеры. В программе суммирования двух чисел можно написать любую из следующих инструкций **PRINT**:

PRINT “сумма и произведение двух чисел”

INPUT “введите два числа”, A, B

S=A + B

P=A*B

PRINT “сумма”; A; “и”; B; “равна”; S

PRINT “произведение”; A; “на”; B; “равно”; P

END

Задания:

По образцу программы для суммы и произведения чисел составьте программы, которые печатают заголовок о своем назначении, вводят с клавиатуры необходимые данные и выводят результат с нужными пояснениями для следующих задач.

- 1) Вычислите площадь прямоугольника по двум сторонам.
- 2) Вычислите гипотенузу прямоугольного треугольника по двум катетам.
- 3) Вычислите площадь правильного треугольника по его стороне.
- 4) Вычислите площадь круга и длину окружности данного радиуса R .
- 5) Вычислите площадь треугольника по основанию и высоте.
- 6) Вычислите площадь треугольника по двум сторонам и углу между ними.
- 7) Вычислите площадь прямоугольного треугольника по катету и гипотенузе.
- 8) Вычислите площадь треугольника по трем сторонам (формула Герона).

Спасибо за внимание!