**Практическая работа №9. Назначение и особенности языка BASIC.**

**Основные элементы языка BASIC.**

Алфавит языка. Алфавит языка Basic представляет собой таблицу символов ASCII. Первая половина этой таблицы (символы с кодом 0-127) - стандартная. Вторая половина (символы с кодом 128-255) специфична для каждой страны. В этой таблице каждый символ имеет 8-битовое обозначение. Итак, в алфавит языка Basic входят все прописные и заглавные буквы английского и русского алфавитов, цифры, а также набор специальных символов, который имеется на клавиатуре компьютера.

Переменные. В Бейсике различают переменные следующих типов:

* числовые переменные;
* символьные переменные;
* переменные пользовательского типа (записи);
* переменные-массивы.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Тип переменной** | **Символ** | **Описание** | **Объём** | **Пример** |
| Целые числа | % | integer | 2 байта | 17; 123 |
| Действит. числа | нет | real | 4 байта | 3.1415 |
| Символьный | $ | string | min 1байт | язык |
| Пользовательский | нет | type |  |  |
| Целый 2 точности | # | double | 8 байт | 1.2543786 |

Переменная - это величина, которая может меняться при выполнении программы. Кроме указанных в таблице, целые числа записывают также в экспоненциальной записи, например, **2,55.105=2.55Е5;      7,15.10-7=7.15Е-7**

Программа на языке Бейсик обрабатывает данные двух типов – числовые и символьные. Каждое число представлено в машине некоторой комбинацией битов. Любое число можно по-разному представить в машине:

1. целое число в диапазоне от –32768 до 32768;
2. длинное целое в диапазоне от –2147483648 до 21474833648;
3. вещественные числа обычной точности;
4. вещественные числа двойной точности;

Данные можно представить переменными и константами.

**Переменные.** С понятием переменной величины вы уже знакомы по урокам алгебры. Например, в простом алгебраическом равенстве ***c = f + 2b – 5*** значение *переменной с* зависит от значения переменной f и b, указанных в правой части равенства. Например, при ***f=2*** и ***b=6, с=9***.

Такое же равенство можно записать в программе на Бейсике.

***c = f + 2\*b – 5***

В терминах языка Бейсик ***c***, ***f*** и ***b*** – это имена переменных. Такие имена также называют *идентификаторами*.

**Идентификаторы**. В языке Бейсик идентификатор – это произвольный набор символов, который может содержать от 1 до 40 символов, причём первый символ должен быть латинской буквой, а остальные – латинские буквы или цифры или символы типа @, #, % и.т.д.

***Пример:***

***A, Ds, SodRan, k1, n123, dlina!***

Тип идентификатора в Бейсике опознаётся по последнему символу в имени переменной.

% - целое число;   
& - длинное целое число;   
! – вещественное число обычной точности;   
# - вещественное число двойной точности;   
$ - символьный тип.

В Бейсике предусмотрен другой способ описания типов переменных (qbasic). Если в начале программы поместить команду: DEFINT I – L, то все переменные, имена которых начинаются с буквы I и лежат в диапазоне до буквы L будут считаться целыми (INTEGER).

**Константы.** Константы, как числовые так и символьные – это величины, которые не меняются в ходе выполнения программы.

***Например:***

***p=3.1415T$=“Скороднянская школа”***

***A=9575   
m$=”Расписание на неделю”***

Иногда записанные константы называют *литералами.* В отношении констант необходимо помнить правила:

* разделителем целой и дробной части является точка;
* значения символьных констант заключается в кавычки.

Числа можно записать в *экспоненциальной* форме, например:

***0.0285=2.85Е-2*** или ***0.0285=2.85D-2***

***784.527=7.84Е+2*** или ***784.527=7.84D+2***

Числа, записанные в экспоненциальной форме, представляют собой произведение мантиссы на порядок, т. е. На 10 в степени –2, +2 и т. п. Буква Е используется для вещественных чисел обычной точности, буква D – для вещественных чисел двойной точности.

Именованные константы. Если присвоить переменной некоторое значение (числовое или символьное) и в дальнейшем не изменять эти значения, например:

F3=60, n%=15

тогда переменные F3 и n% можно считать именованной константой. В qbasic существует специальный оператор для описания именованной константы:

CONST <имя\_переменной = константа, имя переменной = константе>

Например: CONST F2=60, n%=15

Выражения и операции. Последовательность операций, которые необходимо произвести над данными, чтобы получить требуемое значение, называется *выражением.* В Бейсике существует пять категорий операций:

1. Арифметические операции;
2. Операции отношений;
3. Логические операции;
4. Функциональные операции;
5. Строковые операции.

## Арифметические операции.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Название операции | знак в матем. | знак в Бейсике | Пример в математ. | Пример в Бейсике |
| Ввозведение в степень | - | ^ | 25 | 2^5 |
| Сложение | + | + | 5+10 | 5+10 |
| Вычитание | - | - | a-b | a-b |
| Умножение | x;  . | \* | 2.5 | 2\*5 |
| Деление | : | / | 10:2 | 10/2 |

## Операции отношений.

|  |  |  |
| --- | --- | --- |
| Значение операции | Знак операции | Выражение в Бейсике |
| Равенство | = | x=y |
| Неравенство | <> | x<>y |
| Меньше | < | x<y |
| Больше | > | x>y |
| Меньше или равно | <= | x<=y |
| Больше или равно | >= | x>=y |

Включенные в программу комментарии QBASIC просто игнорирует. Использование комментариев позволяет включать в программу любой текст, не нарушая ее семантики. Комментарии используются для объяснения тех или иных действий в программе. Если Вы вернетесь к своей программе через несколько недель (месяцев или даже лет), они помогут Вам быстрее в ней освоится. В QBASIC начало комментария обозначается знаком ' (одинарная кавычка). Остаток строки нельзя переносить в начало следующей строки, содержащей оператор.

***Пример:***  
'Это комментарий  
'PRINT без дополнений выводит на экран пустую строку   
PRINT   
PRINT 'За каждым оператором следует комментарий   
PRINT   
'Вывод пустой строки   
PRINT

***Примечание:*** Вся последняя строка интерпретируется QBASIC как комментарий, т.е. включенный в эту строку оператор PRINT не выполняется. Теперь перейдем к рассмотрению арифметических выражений.

Арифметическое выражение — это комбинация чисел, арифметических констант и некоторых функций, которые связаны между собой знаками арифметических операций. Числа — простейшая форма арифметических выражений. Число состоит из цифр "О...9" и специальных знаков "- + . D". Обратите внимание, что вместо запятой ",", отделяющей целую часть от дробной, используется точка ".".  
Примеры:

|  |  |  |
| --- | --- | --- |
| 4.23 | вместо | 4,23 |
| 9.23D+78 | вместо | 9,23\*1078 |
| -2.77 | вместо | -2,77 |
| -1.34D-23 | вместо | -1,34\*10-23 |

Сложные арифметические выражения можно получить, если связать в программе числа друг с другом, например, для вычислений. Символы, с помощью которых числа вступают друг с другом в некоторые отношения, называют знаками операций. Эти знаки имеют следующие значения:

|  |  |
| --- | --- |
| Знак | Значение |
| \* | умножение |
| / | деление |
| + | сложение |
| - | вычитание |
| ^ | возведение в степень |

Примечание: В QBASIC операции "\*", "/", "^" имеют больший приоритет, чем операции "+", "-". Впрочем, на последовательность вычислений можно влиять, применяя в нужных местах круглые скобки.

Новое представление оператора **PRINT** дает возможность работать с арифметическими выражениями. PRINT *[Выражение]*

**Выражение** - Арифметическое выражение. Значение выражения выводится на экран. Так как в операторе PRINT выражение — необязательный параметр, то c его помощью можно выводить на экран пустую строку.