

Язык программирование

Pascal ABC

Цель:

- *Познакомить с понятием язык программирования Pascal и основными конструкциями языка.*
- *Сформировать навыки использования разных типов данных при решении задач.*
- *Показать комплексное использование компьютера (источник информации, исполнитель алгоритмов)*

Язык программирования –

это совокупность набора символов системы (алфавит), правил образования (синтаксис) и истолкования конструкций из символов (семантика) для задания алгоритмов с использованием символов естественного языка.

Pascal – процедурный язык, имеющий блочную структуру. Набор операторов языка отражает принципы структурного программирования.

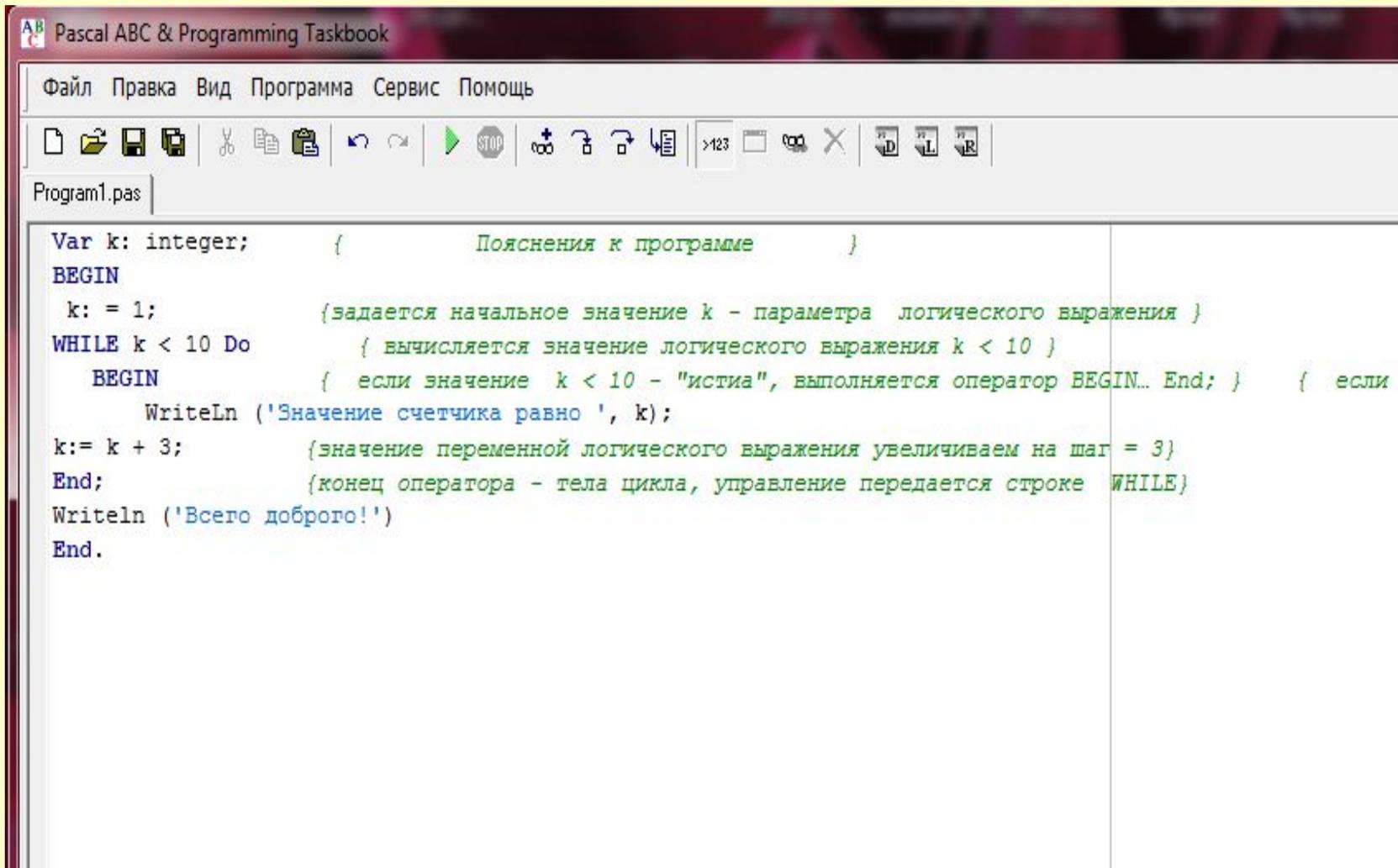
Существует два "вида" переводчиков языков программирования, основная задача которых, перевести программу с языка программирования в машинные коды понятные компьютеру:

- **Компилятор** переводит всю программу сразу и только после этого, если в ней нет ошибок, запускает её на выполнение.

- **Интерпретатор** переводит текст программы построчно и сразу построчно её выполняет.

Запуск Pascal ABC

C:\Program Files\РАВС\PascalABC.exe

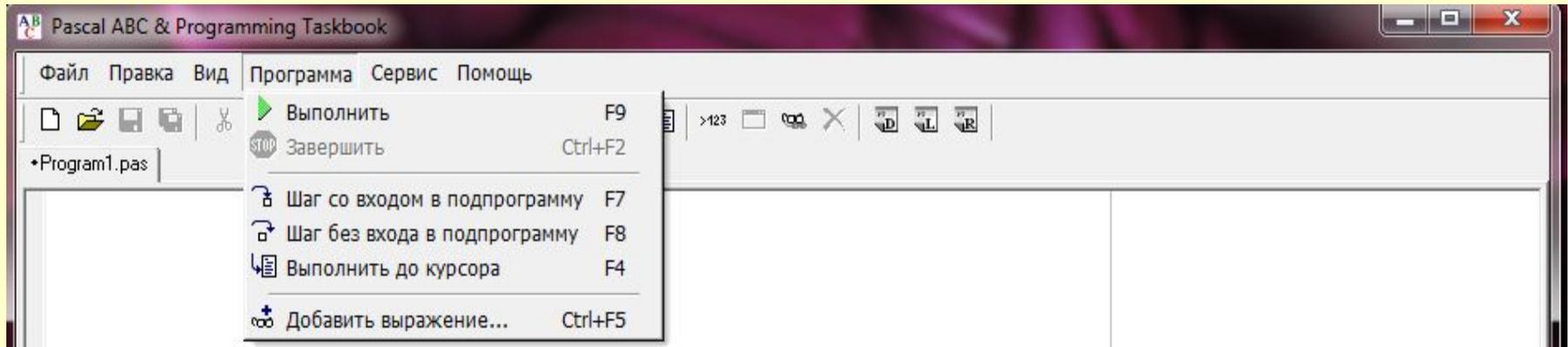


The screenshot shows the Pascal ABC IDE window titled "Pascal ABC & Programming Taskbook". The menu bar includes "Файл", "Правка", "Вид", "Программа", "Сервис", and "Помощь". The toolbar contains icons for file operations (New, Open, Save, Print), editing (Cut, Copy, Paste), execution (Run, Stop), and window management (Zoom, Close, Maximize, Restore). The active file is "Program1.pas". The code in the editor is as follows:

```
Var k: integer;      {      Пояснения к программе      }
BEGIN
  k: = 1;            {задается начальное значение k - параметра логического выражения }
  WHILE k < 10 Do    { вычисляется значение логического выражения k < 10 }
    BEGIN          { если значение k < 10 - "истина", выполняется оператор BEGIN.. End; } { если
      WriteLn ('Значение счетчика равно ', k);
    k:= k + 3;      {значение переменной логического выражения увеличиваем на шаг = 3}
  End;             {конец оператора - тела цикла, управление передается строке WHILE}
  WriteLn ('Всего доброго!')
End.
```

Управление средой

Управление средой осуществляется с помощью меню, которое аналогично меню текстового редактора



Алфавит языка

- 1) 26 латинских букв
- 2) Цифры
- 3) Арифметические операции: $+$, $-$, $/$, $*$,
MOD, DIV
- 4) Знаки отношений: $<$, $>$, $=<$, $>=$, $<>$,
OR, AND, NOT
- 5) Специальные символы:

.	отделяет целую часть от дробной
, ;	- отделяют данные в программе
'	(апостроф) используют для комментариев в программе
{ }	используют для комментариев в программе которые не влияют на результат выполнения программы
..	Разделитель в обозначении диапазона
:=	знак присваивания
()	в арифметических выражениях

Структура программы

Программа на языке Pascal состоит из двух основных частей:



описания всех

данных,

с которыми производятся
действия

описания

самих

действий

В разделе описания могут быть следующие объекты:

<code>Program</code>	имя программы ; {заголовков программы}
<code>Label</code>	{раздел описания меток}
<code>Const</code>	{раздел описания констант}
<code>Type</code>	{раздел описания типов}
<code>Var</code>	{раздел описания переменных}

В разделе действий программы

`Begin`

{тело программы, представляет собой последовательность операторов, разделенных символом «;»}

`End.`



Program1.pas

```
Var k: integer;  
BEGIN  
  k := 1;  
  WHILE k < 10 Do  
    BEGIN  
      WriteLn ('Значение k = ' + IntToStr(k));  
      k := k + 3;  
    End;  
  WriteLn ('Всего доброго!');  
End.
```

Описание данных

Описание действий

Конструкции языка

Из символов алфавита конструируются объекты называемые данными.

- *Данные* – величины, которые обрабатываются командами и операторами языка.

Данные Pascal :

- Модули;
- Константы;
- Переменные;
- Функции;
- Выражения;
- Массивы;
- Метки.

Типы Данных

В любой задаче используются, обрабатываются какие-либо данные числа, целые или вещественные, массивы, символы, слова и т.д. Все они характеризуются своим типом

Первичными в иерархии типов являются стандартные скалярные, представляющие традиционные типы в языках программирования:

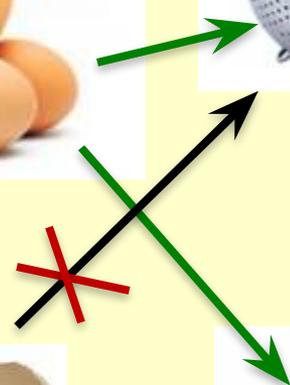
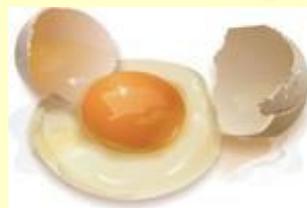
Стандартные скалярные типы данных

- целые типы (**Integer**, Диапазон допустимых значений -32768.. 32767, размер памяти 2 байта);
- вещественные типы (**Real**, диапазон допустимых значений $2.9 \text{ E } -39.. 1.7 \text{ E } 38$, размер памяти 6 байт)
- символьный тип (**Char**, Значениями символьного типа являются символы из множества ASCII, размер памяти, занимаемый символом, - 1 байт.
- строковый тип (**String**, символы из множества ASCII, занимаемый несколько символом.
- логический тип (**Boolean**, основу математической логики составляют две константы: True (Истина, 1) и False (Ложь, 0). Логические данные широко используются при сравнении величин.

Типы данных

integer – целые числа

1, 100, - 1000 и т.д.



real – действительные числа

1.5, 104, - 1000.5674 и т.д.

Д.3:

- Выучить лекцию

Описание данных в программе

Переменные - величины значение которых меняется в ходе выполнения действий над ними

Пример: (переменной F присвоено значение 56): **F := 56**

Имя переменной
(идентификатор)

Значение
переменной

Знак
присваивания

Имя переменной состоит из одного или нескольких символов латинского алфавита.

Недопустимые имена переменных:

21A	Начинается не с буквы
A C	Содержит пробелы
АНЯ	Содержит русские буквы
A&B	Содержит специальные символы
Char	Содержит команду

Раздел описания переменных начинается со служебного слова **var**, само описание переменной содержит два элемента:

- имя переменной (идентификатор) ;
- ее тип

Var имя переменной : тип переменной;

Эти элементы разделяются двоеточием, после указания типа ставится разделитель - СИМВОЛ «;».

Например:

Описание переменных

```
Var al: integer;  
    b,c: real;  
    x: char;  
    y: string;  
    a,b1: boolean;
```

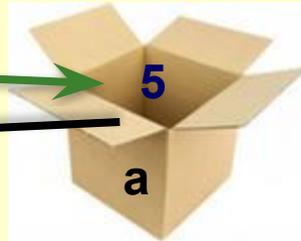
Действие над
переменными в
программе

```
al:=4;  
b:=3.5; c:=0.0034;  
x:='A'; y:='алф' ;  
a:=True;  
b1:=False;
```

Переменная в программе

•Program1.pas

```
var a: integer;  
begin  
a:=5;  
write(a);  
end.
```



CRT - программа завершена

5
_

Константы - постоянные величины изменение которых в программе не предусмотрено.

Константы по внешнему виду похожи на переменные, однако в отличие от них не могут изменить своего значения.

Описание констант начинается со служебного слова **const**

В языке имеются стандартные константы, которые можно использовать без предварительного объявления:

Пример:

const

```
max = 1000;
```

```
Pi = 3.1415;
```

```
C = Pi*2;
```

```
min = - max;
```

pi	3.141592
True	«ИСТИНА»
false	«ЛОЖЬ»

Выражения

Выражения используются для вычисления новых значений.

Операнды - это некоторые «элементарные» значения: переменные, константы, вызовы функций.

Примеры выражений:

$$a+b-x;$$

$$y -(2-y+c)*y;$$

$$\sin(x) -2 + 56;$$

При составлении выражений необходимо знать следующие правила:

- всё выражение должно быть записано в строку. Например:

$$\frac{ax - a^2}{x_1 - x_2} \Rightarrow ((a * x) - a^2) / (x_1 - x_2);$$

- в выражении в качестве разделителей можно использовать только круглые скобки;
- нельзя записывать подряд два знака.
Например, форма записи $a + b / -c$ неправильна, правильной является форма $a + b / (-c)$.

Стандартные функции языка

Pascal	Математическое определение и результат работы
SQR(X)	функция возведения в квадрат, где x - переменная целого или вещественного типа.
SQRT(X)	\sqrt{X} - функция извлечения квадратного корня, где x - переменная целого или вещественного типа.
ABS(X)	$ X $ -функция определения модуля числа x , где x переменная целого или вещественного типа
sin(x), cos(x), arctan(x)	тригонометрические функции, аргументы и значения которых всегда вещественного типа
round (x)	операция округления вещественного числа x до ближайшего целого (round (5 .67) =6; round (5 . 47) =5).
exp (x)	- <i>экспонента в степени x</i> , аргумент и значение всегда вещественного типа.
trunc (x)	- отбрасывание вещественной части x , x - вещественное значение (trunc(5.67)=5).
In (x)	- натуральный логарифм x , аргумент и значение всегда вещественного типа.

Арифметические операции

+, -, *, /, div, mod

Операцию « / » целых чисел использовать нельзя, так как результат при этом будет выходить за рамки целого типа. Вместо деления к целочисленным данным применяют две операции:

операция «деление нацело» с отбрасыванием дробной части, обозначается служебным словом **div**

$(14 \text{ div } 3 = 4, \quad 5 \text{ div } 8 = 0)$;

операция «взятие остатка от целочисленного деления», обозначается служебным словом **mod**

$(14 \text{ mod } 3 = 2, \quad 5 \text{ mod } 8 = 5)$.

11 div 5

$$\begin{array}{r} 11 \overline{) 5} \\ -10 \\ \hline 1 \end{array}$$

11 mod 5

Пример с отрицательными числами

$$-7 \text{ div } 2 = -3$$

$$-7 \text{ mod } 2 = -1$$

-3
-1

$$-7 = (-3) * 2 + (-1)$$

Функции $\ln(x)$ и $\exp(x)$ могут быть использованы для возведения в степень по правилу:

$$x^n = \exp(n \ln(x))$$

Например, значение x^9 вычисляется по формуле $\exp(9 \ln(x))$.

Д/з: ВYУЧИТЬ ЛЕКЦИЮ



<https://cloud.mail.ru/public/52Vd/3aq1J3TmL>