

## Пример программы

```
program First; // заголовок программы – необязательная строка
{ Программа вычисления длины окружности и площади круга
  Автор: Михалкович С.С. Дата написания: 2.09.13 }
const Pi = 3.14;
var
  r: real; // входные данные – радиус круга
  S,C: real; (* выходные данные – площадь круга и длина окружности *)
begin
  write('Введите радиус окружности: ');
  readln(r);
  S := Pi*r*r;
  C := 2*Pi*r;
  writeln('Длина окружности равна ',C);
  writeln('Площадь круга равна ',S);
end.
```

## Новые возможности PascalABC.NET (сайт [pascalabc.net](http://pascalabc.net))

```
var x: integer := 3; // инициализация при описании
begin
  var y: integer := x; // внутриблочное описание
  y += 2; // y := y + 2;
  var z := x; // внутриблочное описание с автоопределением типа
  z *= 2;
  writelnFormat('x={0} y={1} z={2}', x, y, z);
end.
```

## Типы данных

Целые знаковые: integer (4 байта), shortint (1), smallint (2), int64 (8)

Целые беззнаковые: byte (1), word (2), longword (4), uint64 (8)

Вещественные: real (8), single (4)

Символьный: char (2 байта) – Unicode

Строковый: string

Логический: boolean (1 байт)

## Стандартные функции

abs(x)	sqr(x)	sqrt(x)	ln(x)	log10(x)	logn(base, x)	exp(x)
sin(x)	cos(x)	tan(x)	arctan(x)			
int(x)	frac(x)	trunc(x)	round(x)	floor(x)	ceil(x)	
odd(i)	max(a,b)	min(a,b)	pow(a,b)			
succ(c)	pred(c)	ord(c)				

## Стандартные процедуры

Inc(x) Dec(x) Inc(x,n) Dec(x,n)

## Приоритет операций

1. + - (унарные) **not**
2. / **div mod shl shr and**
3. + - (бинарные) **or xor**
4. < > <= >= = <> **in**

## Операторы

### Операторы присваивания

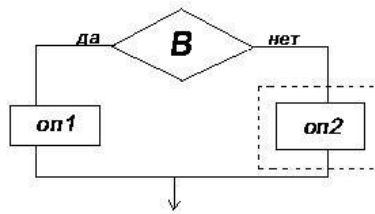
*переменная := выражение*  
*переменная += выражение*  
*переменная -= выражение*

*переменная \*= выражение*  
*переменная /= выражение*

### Условный оператор

**if** *B* **then** *оператор<sub>1</sub>*  
 [**else** *оператор<sub>2</sub>*]

(*B* – логическое выражение)



### Составной оператор (блок)

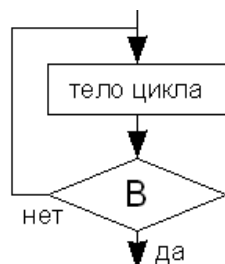
**begin**  
*операторы*  
**end**  
*операторы ::= [оператор {; оператор }]*

### Оператор выбора варианта

**case** *выражение-переключатель* **of**  
 { *список выбора: оператор;* }  
 [**else** *оператор;*]  
**end;**

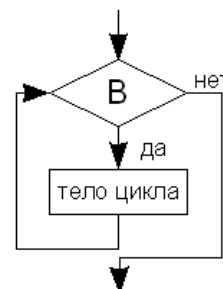
### Оператор цикла с предусловием (ПОКА)

**while** *B* **do**  
*оператор*



### Оператор цикла с постусловием (ДО)

**repeat**  
*операторы*  
**until** *B*



### Оператор цикла с параметром

*направление ::= to | downto*  
**for** *переменная := выражение<sub>1</sub> направление выражение<sub>2</sub>* **do**  
*оператор*  
**for var** *переменная := выражение<sub>1</sub> направление выражение<sub>2</sub>* **do**  
*оператор*  
**for var** *переменная: тип := выражение<sub>1</sub> направление выражение<sub>2</sub>* **do**  
*оператор*