

# ПРОГРАММИРОВАНИЕ ЛИНЕЙНЫХ АЛГОРИТМОВ

## НАЧАЛА ПРОГРАММИРОВАНИЯ

# Числовые типы данных

Стандартные функции языка Паскаль:

Функция	Назначение	Тип аргумента	Тип результата
<b>abs (x)</b>	Модуль $x$	<b>integer, real</b>	Такой же, как у аргумента
<b>sqr (x)</b>	Квадрат $x$	<b>integer, real</b>	Такой же, как у аргумента
<b>sqrt (x)</b>	Квадратный корень из $x$	<b>integer, real</b>	<b>real</b>
<b>round (x)</b>	Округление $x$ до ближайшего целого	<b>real</b>	
<b>frac (x)</b>	Целая часть $x$	<b>real</b>	
<b>int (x)</b>	Дробная часть $x$	<b>real</b>	
<b>random</b>	Случайное число от 0 до 1	-	<b>real</b>
<b>random (x)</b>	Случайное число от 0 до $x$	<b>integer</b>	<b>integer</b>

Исследуем работу функций **round**, **int** и **frac**, применив их к некоторому вещественному  $x$ .  
Соответствующая программа будет иметь вид:

```
program n_3;  
  var x: real;  
begin  
  writeln ('Исследование функций round, int, frac');  
  write ('Введите x>>');  
  readln (x);  
  writeln ('Округление - ', round(x));  
  writeln ('Целая часть - ', int(x));  
  writeln ('Дробная часть - ', frac(x))  
end.
```

# Целочисленный тип данных

Операции над целыми числами в языке Паскаль:

Операция	Обозначение	Тип результата
Сложение	+	integer
Вычитание	-	integer
Умножение	*	integer
Получение целого частного	div	integer
Получение целого остатка деления	mod	integer
Деление	/	real

# Операции **div** и **mod**

Трёхзначное число можно представить в виде следующей суммы:  $x = a \cdot 100 + b \cdot 10 + c$ , где  $a$ ,  $b$ ,  $c$  - цифры числа.

Программа нахождения суммы цифр вводимого с клавиатуры целого трёхзначного числа.

```
program n_4;  
  var x, a, b, c, s: integer;  
begin  
  writeln ('Нахождение суммы цифр трёхзначного числа');  
  write ('Введите исходное число>>');  
  readln (x);  
  a:=x div 100;  
  b:=x mod 100 div 10;  
  c:=x mod 10;  
  s:=a+b+c;  
  writeln ('s= ', s)  
end.
```

# Символьный и строковый типы данных

**Символы** – это все буквы и значки, которые есть на клавиатуре. Для ввода в программу символьных переменных необходимо указать для них символьный тип данных **char**.

Функция **ord** преобразовывает букву в её числовой код.

Функция **chr** преобразовывает числовой код символа в сам символ.



# Символьный и строковый типы данных

**Символы** – это все буквы и значки, которые есть на клавиатуре. Для ввода в программу символьных переменных необходимо указать для них символьный тип данных **char**.

Функция **ord** преобразовывает букву в её числовой код.

Функция **chr** преобразовывает числовой код символа в сам символ.



# Символьный и строковый типы данных

**Символы** – это все буквы и значки, которые есть на клавиатуре. Для ввода в программу символьных переменных необходимо указать для них символьный тип данных **char**.

Функция **ord** преобразовывает букву в её числовой код.

Функция **chr** преобразовывает числовой код символа в сам символ.



Значением строковой величины (тип **string**) является произвольная последовательность символов, заключенная в апострофы.

```
var c: string  
c:= chr(52)+chr(37)
```





# Символьный и строковый типы данных

```
program n_5;  
  var a: char; kod: integer; b: string;  
begin  
  writeln ('Код и строка');  
  write ('Введите исходную букву>>');  
  readln (a);  
  kod:=ord(a);  
  b:=chr(kod-1)+a+chr(kod+1);  
  writeln ('Код буквы ', a, '-', kod);  
  writeln ('Строка: ', b)  
end.
```

Вывод на экран кода буквы, введённой с клавиатуры

Вывод на экран строки из трёх букв. Каких?

# Логический тип данных

Величины логического типа принимают всего два значения: **false** и **true**;

**false** < **true**.

Логические значения получаются в результате выполнения операций сравнения числовых, символьных, строковых и логических выражений.

В Паскале логической переменной можно присваивать результат операции сравнения.

# Логический тип данных

Пример. Напишем программу, определяющую истинность высказывания «Число  $n$  является чётным» для произвольного целого числа  $n$ .

Пусть **ans** - логическая переменная,  **$n$**  - целая переменная.

В результате выполнения оператора присваивания  **$ans:=n \bmod 2=0$**  переменной **ans** будет присвоено значение **true** при любом чётном  **$n$**  и **false** в противном случае.

```
program n_6;  
  var n: integer; ans: boolean;  
begin  
  writeln ('Определение истинности высказывания о чётности числа');  
  write ('Введите исходное число>>');  
  readln (n);  
  ans:=n mod 2=0;  
  writeln ('Число ', n, ' является четным - ', ans)  
end.
```

# Логический тип данных

Логическим переменным можно присваивать значения логических выражений, построенных с помощью логических функций и (**and**), или (**or**), не (**not**).

Логическая операция в Паскале	Название операции
and	конъюнкция (логическое умножение)
or	дизъюнкция (логическое сложение)
not	отрицание (инверсия)

# Логический тип данных

Пример. Напишем программу, определяющую истинность высказывания «Треугольник с длинами сторон  $a$ ,  $b$ ,  $c$  является равнобедренным» для произвольных целых чисел  $a$ ,  $b$ ,  $c$ .

```
program n_7;  
  var a, b, c: integer; ans: boolean;  
begin  
  writeln ('Определение истинности высказывания  
           о равнобедренном треугольнике');  
  write ('Введите значения a, b, c>>');  
  readln (a, b, c);  
  ans:=(a=b) or (a=c) or (b=c);  
  writeln ('Треугольник с длинами сторон ', a, ',', b, ',', c, ' является равнобедренным - ', ans)  
end.
```

Задание 1. Запиши математическое выражение  $(20 + s)^2$  на языке Паскаль. Ответ записать в тетради.

Задание 2. Выражение записано на языке Паскаль:  $((a+b)/6$  . Какая ошибка допущена? Запишите ответ в тетради.

**Задание 3. стр.147 № 9 (а) Письменно  
в тетради.**



# Опорный конспект

**Типы данных** в языке Паскаль: вещественный, целочисленный, символьный, строковый, логический.

