

# **ОПЕРАТОРЫ**

## **ПРИСВОЕНИЯ,**

## **ВВОДА, ВЫВОДА**

ВВОДА, ВЫВОДА



# ИЗУЧЕНИЕ НОВОГО МАТЕРИАЛА

1. Что такое оператор?
2. Оператор присвоения
3. Операторы вывода
4. Операторы ввода
5. Закрепление материала



# ЧТО ТАКОЕ ОПЕРАТОР?

- **Операторы** - это служебные слова, написанные на английском языке и выполняющие определённые функции.
- **Их размещение в Программе** – в операторных скобках (begin – end.)

# ОПЕРАТОР ПРИСВОЕНИЯ



**<имя переменной> := <значение>;**

1. Вычисляется значение, стоящее в правой части оператора присваивания.
2. Результат записывается в переменную, имя которой указано в левой части.

**Пример:**

- ✓ C:=45; в C будет число "45"
- ✓ B:=C-5; в B будет число "40"
- ✓ X:=X+5; увеличивает значение X на 5 единиц
- ✓ Y:='Привет'; присваивает переменной Y текст



# ОПЕРАТОРЫ ВЫВОДА WRITE И WRITELN

- ✚ Операторы Writeln и Write служат для вывода информации на экран компьютера.
- ✚ Процедура Write после напечатанного оставляет курсор на той же строчке, а Writeln переводит его после печати на новую строку.

## Write (список вывода):

Например:

**Write** (A,B,T); на экране: **0, 0, 0**

**Writeln** (A,B,T); на экране: **0**

**0**

**0**

# ЧТО ВЫВОДИТ НА ЭКРАН ОПЕРАТОР WRITE

## 4 варианта:

1. Текст, заключённый в апострофы
2. Имя переменной
3. Арифметические выражения
4. Алгебраические выражения





# ТЕКСТ, ЗАКЛЮЧЁННЫЙ В АПОСТРОФЫ

1. В него могут входить любые символы, в том числе и русские буквы. При этом текст будет напечатан на экране точно в таком же виде, в каком указан в операторе.

## Например:

- ✓ в программе - `Write ('Привет!');`
- ✓ на экране - `Привет!`
- ✓ в программе - `Write ('2+2=4');`
- ✓ на экране - `2+2=4`
- ✓ в программе - `Write ('S=');`
- ✓ на экране - `S=`

# ИМЯ ПЕРЕМЕННОЙ

2. На экране будет напечатано ее значение.

**Например:** если  $X=675$ ,

- ✓ в программе - `Write (X);`
- ✓ на экране - `675`
- ✓ в программе - `Write ('X=',X);`
- ✓ на экране - `X=675`



# АРИФМЕТИЧЕСКОЕ ВЫРАЖЕНИЕ

в) Будет вычислено его значение и напечатан результат.

Например:

- ✓ в программе - `Write (34*2+102);`
- ✓ на экране - `170`

# АЛГЕБРАИЧЕСКОЕ ВЫРАЖЕНИЕ

с) Будет вычислено его значение и напечатан результат.

Например: если  $A=6$   $B=7$

в программе - `Write (2*A*B);`

на экране - **84**



# ОПЕРАТОРЫ ВВОДА READ И READLN

- ✦ Процедуры **READ** и **READLN** служат для ввода переменных в компьютер.
- ✦ Они отличаются только тем, что процедура **READ** после напечатанного оставляет курсор на той же строчке, а **READLN** переводит его после печати на новую строку.
- ✦ Например: **READ** (A) на экране: A = ?  
**READLN** (A) на экране: A  
= ?

# ОБЩИЙ ВИД

## **Readln (имя1,...,имяN);**

✓ Процедура **Readln** считывает значения, вводимые с клавиатуры, и записывает их в переменные, которые указаны.

✓ Например

Если на запрос **Readln (a, c, x);** с клавиатуры будут введены числа **2, 5, 4.6**, то переменные получат соответствующие значения:

**a=2, c=5, x=4.6.**



# ОСНОВНЫЕ ТИПЫ ПЕРЕМЕННЫХ

## 1). нат - **Word**

целые положительные числа  
[0, 65535]

## 2). цел - **Integer**

целые числа из интервала  
[-32768, 32767];

## 3). вещ - **Real**

дробные числа из интервала  
[-2.9\*10<sup>-39</sup>, - + 2.9\*10<sup>-39</sup>]

## 4). лит - **Char** - хранит 1 символ (буква, знак или код, занимает 1 байт)

## 5). лит - **Cstring** - хранит слова, занимает 16 байт)

# ЗАКРЕПЛЕНИЕ МАТЕРИАЛА

- См в программе Паскаль 3 программы - задания в папке **Vvod**