

Паскаль.

Типы данных

Переменная

Математические

операции

Оператор read, write

Оператор if

Паскаль. Типы данных.

Типы данных определяют:

- ☐ Возможные значения переменных
- ☐ Внутреннюю форму представления числа в памяти РС
- ☐ Операции на данными

В языке Паскаль (Делфи) типы данных (переменных) указывают в программе заранее!

Типы данных. Типы данных.

4

Идентификатор	Длина (байт)	Диапазон значений	Операции
integer	2	-32768..32767	+, -, /, *, Div, Mod, >=, <=, =, <>, <, >
byte	1	0..255	+, -, /, *, Div, Mod, >=, <=, =, <>, <, >
word	2	0..65535	+, -, /, *, Div, Mod, >=, <=, =, <>, <, >
real	6	$2,9 \times 10^{-39}$ - $1,7 \times 10^{38}$	+, -, /, *, >=, <=, =, <>, <, >
boolean	1	true, false	Not, And, Or, Xor, >=, <=, =, <>, <, >
char	1	все символы кода ASCII	+, >=, <=, =, <>, <, >

Варианты описание переменных в программе:

- a) var переменная:тип данных;
- b) var переменная1,переменная2:тип данных;

Пример программы:

```
program parimer;  
  var x:bute;  
begin  
  read(x);  
  write(x);  
end.
```

Описание типа
данных в
программе

Вопрос к аудитории:

Какие значения можно вводить с клавиатуры?

Пример программы:

```
program parimer;  
  var x:real;  
begin  
  read(x);  
  write(x);  
end.
```

Вопрос к аудитории:

Введено с клавиатуры значение:

- a) 125;
- b) true;
- c) Привет

Что будет происходить после ввода значений?

Паскаль. Переменная.

Переменной называют элемент программы, который предназначен для хранения, коррекции и передачи данных внутри программы.

Описание переменной:

См. в тетради тему “Типы данных”

Переменные. Работа с переменными⁹.

Присвоение значение переменной:

- a) переменная:= значение переменной;
- b) переменная:= переменная;

Переменная которой
ПРИСВАЕВАЕТСЯ
значение

Значение (или
значение
переменной) которое
ПЕРЕДАЕТСЯ
переменной

Передача переменной числового значения “3”:

```
var x:byte;  
begin  
x:=3;  
end.
```

Передача переменной числового значения “54666”:

```
var x:string;  
begin  
x:='Текс';  
end.
```

Передача переменной X числового значения
переменной Y. Вывод на экран значения
переменной Y:

```
var x,y:byte;  
begin  
    read(x);  
    y:=x;  
    write(y);  
end.
```

Задача. Увеличить переменную X на единицу

```
var x,y,c:byte;  
begin  
  x:=3;  
  x:=x+1;  
end.
```

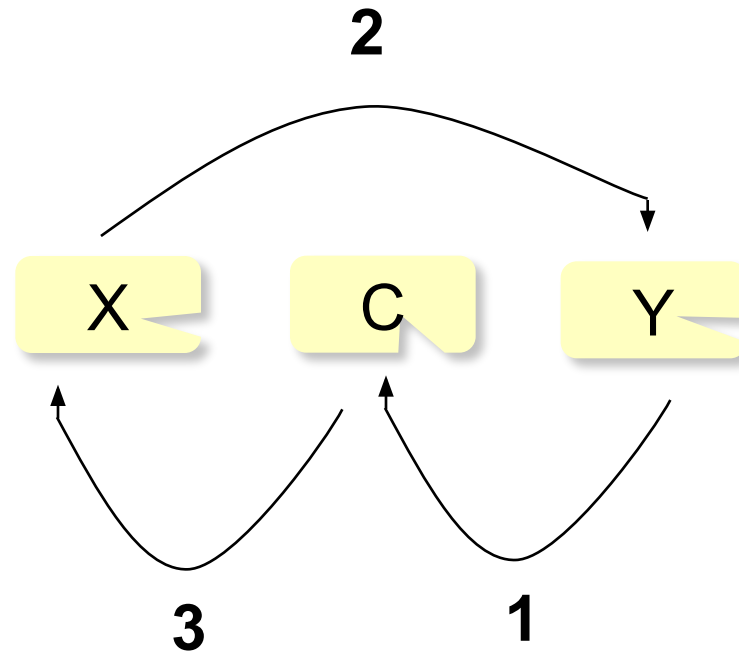
x:=x+1;

Новое
значение "x"

Старое
значение "x"

Задача. Поменять местами значения переменных X и Y:

```
var x,y,c:byte;  
begin  
    c:=y;  
    y:=x;  
    x:=c  
end.
```



Паскаль. Математические операции.

Символ операции	Название операции	Пример
*	умножение	$2 * 3$ (результат: 6)
/	деление	$30 / 5$ (результат: 6)
+	сложение	$2 + 3$ (результат: 5)
-	вычитание	$5 - 3$ (результат: 2)
div	целочисленное деление	$5 \text{ div } 2$ (результат: 2)
mod	остаток от деления	$5 \text{ mod } 2$ (результат: 1)

Математические операции. Функции¹⁶

Abs(X)

Возвращает абсолютное значение числа X.

Cos(X), Sin(X)

Возвращает косинус (синус) числа X, где X - угол в радианах.

Ln(x)

Возвращает число, равное натуральному логарифму от числа X.

Pi

Число Пи.

Sqr(X)

Возвращает число, равное квадрату числа X.

Sqrt(X)

Возвращает число, равное квадратному корню из числа X.

Математические операции. Примеры.¹⁷

Задача. Сложить два числа 2 и 6. Вывести результат на экран:

```
var x:byte;  
begin  
x:=2+6;  
write(x);  
end.
```

Математические операции. Примеры.¹⁸

Задача. Прибавить единицу к введенному с клавиатуры числу. Вывести результат на экран.

```
var x:integer;  
begin  
  read(x);  
  x:=x+1;  
  write(x);  
end.
```

Математические операции. Примеры.¹⁹

Задача. Возвести в квадрат число введенное с клавиатуры. Вывести результат на экран.

Решение 1:

```
var x:integer;  
begin  
  read(x);  
  x:=x*x;  
  write(x);  
end.
```

Решение 2:

```
var x:integer;  
begin  
  read(x);  
  x:=sqr(x);  
  write(x);  
end.
```

Паскаль. Оператор read, write.

Оператор чтения данных с экрана:

Read(x) – читает с экрана и присваивает его переменной x

Read(x,y) - читает поочередно с экрана значения и присваивает его переменной x и y соответственно.

Оператор чтения данных с экрана:

Write(x) – выводит на экран значение переменной x.

Write('text') – выводит на экран значение переменной текст “text”.

Write(x:10:3) – выводит значение переменной округленной до 3 знаков после запятой.

Write('Число',y) – выводит на экран текст “Число” и значение переменной y.

Write('Число',y,'отрицательное') – выводит на экран текст “Число”, значение переменной y и текст “отрицательное”

Паскаль. Оператор if.

Описание 1: *if условие then оператор 1;*

Логическое условие
(простое или составное)

Оператор 1

Определение: “Оператор 1” выполняется тогда и только тогда, когда “условие” является истинным!

Описание 2: *if условие then оператор 1 else оператор 2;*

Определение: “Оператор 2” выполняется тогда и только тогда, когда “условие” является ложным!

Оператор if. Другие определения.²⁵

Описание:

if условие **then** оператор 1 **else** оператор2;

Описание на русском языке:

Если условие истинно **то** оператор 1 **иначе** оператор 2;

Оператор if. Логические операции.²⁶

Логические операции используются для проверки различных условий. Если условие выполняется, то оно считается истинным (true), если нет – ложным (false).

Логическая операция	Описание	Логическая операция	Описание
=	равное	>=	Больше или равно
>	больше	<=	Меньше или равно
<	меньше	<>	Не равно
==	точное равно		

Пример.

- 1.) Если $x = 5$, то $x > 3$ – выражение будет истинно (true).
- 2.) Если $x = 8$, то $x \leq 7$ – выражение будет ложно (false).

Оператор if. Примеры.

Задача: Составить программу для вывода на экран сообщения о том является ли число отрицательным или положительным.

Решение 1:

```
program test;  
var x:real;  
begin  
  writeln('Введите число');  
  read(x);  
  if x>=0 then write('Число является положительным');  
end.
```

Решение 2:

```
....  
if x>=0 then write('Число является положительным') else  
write('Число является отрицательным');  
...
```

Задача: Даны два числа вывести наибольшее из них.

Решение:

```
program test;  
var x,y:integer;  
begin  
  writeln('Введите число 1');  
  read(x);  
  writeln('Введите число 2');  
  read(y);  
  if x>y then write('Число ',x,' больше числа ', y)  
  else write('Число ',y,' больше числа ', x);  
end.
```

Задача: Проверить является ли число четным.

Решение 1:

```
program chetno;  
var x:integer;  
begin  
  writeln('Введите число');  
  read(x);  
  if (x mod 2)=0 then write('Число четное') else  
    write('Число не четное');  
end.
```

Оператор if. Задачи.

Задача 1. Проверить является ли число больше 8.

Задача 2. Проверить делится ли число на 3.

Задача 3. Проверить является ли число нечетным (условие проверки “ $(x \bmod 2) = 0$ ” не использовать).

Задача 4. Даны два числа, вывести на экран наименьшее из них.

Задача 5. Дано целое число. Если оно является положительным, то прибавить к нему 5; в противном случае – вычесть 1. Результат вывести на экран.

Задача 6. Даны 3 целых числа, найти количество положительных чисел.

Оператор if. Составные лог. операторы.³¹

Логические операторы **and** (логическое “и”), **or** (логическое “или”) – позволяют создавать составные логические выражения.

Таблица истинности для and (“и”)

Условие 1	Условие 2	Выражение
true	true	true
true	false	false
false	true	false
false	false	false

Оператор if. Составные лог. операторы.³²

Таблица истинности для or (“или”)

Условие 1	Условие 2	Выражение
true	true	true
true	false	true
false	true	true
false	false	false

Оператор if. Составные лог. операторы.³³

Таблица истинности для not (отрицание)

Условие 1	Выражение
true	false
false	true

Оператор if. Примеры выражений.³⁴

Пример 1:

$X = 5$ и $Y = 6$

Выражение $(x < 7)$ and $(y > 5)$ – истинно

true

true

Выражение $(x < 1)$ and $(y > 5)$ – ложно

false

true

Оператор if. Примеры выражений.³⁵

Пример 2:

$X = 5$ и $Y = 6$

Выражение $(x < 7) \text{ or } (y > 5)$ – истинно

true

true

Выражение $(x < 1) \text{ or } (y > 5)$ – истинно

false

true

Оператор if. Примеры выражений.³⁶

Пример 3:

$X = 5$ и $Y = 6$

Выражение $(\text{not } x < 7)$ – ложно

true

Выражение $(\text{not } x < 1)$ – истинно

false

Задача: Составить программу для проверки является ли число больше 5 и меньше 20 одновременно.

Решение:

```
program test;  
var x:integer;  
begin  
  writeln('Введите число');  
  read(x);  
  if (x>5) and (x<20) then  
    write('Является')  
  else write('Не является');  
end.
```

Оператор if. Программные скобки.³⁸

Описание 3:

```
if условие then  
begin  
  оператор 1;  
  оператор 2;  
  ...  
  оператор n;  
end;
```

Открытие программных
скобок

Закрытие
программных скобок

Оператор if. Программные скобки.³⁹

Задача:

Поменять местами переменные, если первая переменная больше второй.

```
...  
If x > y then  
begin  
  c:=y;  
  y:=x;  
  x:=y;  
end;  
writeln(x, ' ', y);  
end.
```

Оператор if. Программные скобки.⁴⁰

Задача:

Составить программу для вычисления корней квадратного уравнения

```
...  
d:= B*B-4*A*C;  
If d > 0 then  
begin  
x1:=(-1)*B+sqr(d);  
x2:=(-1)*B-sqrt(d);  
writeln('Первый корень равен ',x1);  
writeln('Второй корень равен ',x2);  
end;  
If d = 0 then ...  
If d < 0 then ...  
end.
```


Паскаль. Циклы. Оператор for.

Определение 1:

Многократно повторяющийся участок вычислительного процесса называется **циклом**.

Определение 2:

Если заранее известно количество необходимых повторений, то цикл называется **арифметическим**.
Если же количество повторений заранее неизвестно, то говорят об **итерационном** цикле.

В **итерационных** циклах производится проверка некоторого условия, и в зависимости от результата этой проверки происходит либо выход из цикла, либо повторение выполнения тела цикла.

Если проверка условия производится перед выполнением блока операторов, то такой итерационный цикл называется циклом **с предусловием** (цикл "пока"), а если проверка производится после выполнения тела цикла, то это цикл **с постусловием** (цикл "до").

Особенность этих циклов заключается в том, что тело цикла с постусловием всегда выполняется хотя бы один раз, а тело цикла с предусловием может ни разу не выполниться. В зависимости от решаемой задачи необходимо использовать тот или иной вид итерационных циклов.

Описание:

***for переменная := значение 1 to значение 2 do
оператор;***

Оператор **for** вызывает *оператор*, находящийся после слова **do**, по одному разу для каждого значения в диапазоне от “*значения 1*” до “*значения 2*”.

Задача:

Вывести на экран числа от 1 до 16.

Решение:

```
...  
i:=0;  
for i:=1 to 16 do WriteLn(x);  
...
```

Задача:

Вывести на экран таблицу умножения на 8.

Решение:

```
...  
for i:=1 to 10 do WriteLn('8 * ',i,'= ',8*i);  
...
```

Задача:

Даны целые числа K и N ($N > 0$). Вывести N раз число K .

Даны два целых числа A и B ($A < B$). Вывести в порядке возрастания все целые числа, расположенные между A и B (включая сами числа A и B), а также количество N этих чисел.

Даны два целых числа A и B ($A < B$). Вывести в порядке убывания все целые числа, расположенные между A и B (не включая числа A и B), а также количество N этих чисел.

Циклы. Арифметические циклы.

49

Задача:

Дано вещественное число — цена 1 кг конфет. Вывести стоимость 1.2, 1.4, ..., 2 кг конфет.

Даны два целых числа A и B ($A < B$). Найти сумму всех целых чисел от A до B включительно.

Даны два целых числа A и B ($A < B$). Найти произведение всех целых чисел от A до B включительно.

Даны два целых числа A и B ($A < B$). Найти сумму квадратов всех целых чисел от A до B включительно.

