

Обучение программированию Pascal ABC

Автор :
учитель МБОУ «Завьяловской
СОШ 1 Завьяловского района»
Алтайского края
Киян Ирина Владимировна

Алфавит Я.П. Pascal ABC

Алфавитом языка называется совокупность всех допустимых символов, которые можно использовать в этом языке.

Алфавит языка Паскаль

1. Приписные и строчные латинские буквы от A до Z
2. Цифры 0 1 2 3 4 5 6 7 8 9
3. Одиночные знаки: + - / * = < > ^ & ; : #
4. Парные () { } []

Структура программы

Program имя программы; } заголовок

var переменные: ТИП; } раздел описания
переменных

Begin

операторы; } тело программы

End.

Ввод каждой строки заканчивается
СИМВОЛОМ ;

Заголовок

Имя программы – слово, записанное буквами латинского алфавита, возможно использовать цифры после букв. Нельзя использовать русские буквы, символы + - / = пробел, нельзя начинать имя с цифры.

Раздел описания переменных

Переменная – латинская буква, несколько переменных перечисляются через запятую.

Типы переменных

1. Целый тип – **integer**
2. Вещественный тип – **real**

Записываются в форме десятичной точки 42.8
96.45 либо в форме порядка 31.4 E 5=3140000
символ E является признаком десятичного
порядка.

Пример `var n, p: integer;`

Тело программы

Оператор присваивания :=

Пример: **a:= 100;**

b:=5;

a:=a+b;

Процедура ввода данных

read (a,b);

Процедура вывода данных

write (s);

Комментарии к программе

Текст для пояснения к программе

записывается в **{ } (* *) //**

Русские буква используются при вводе и выводе текста и заключены в апострофы ' '

Схема исполнения программы на языке Паскаль

- 1) Ввод исходных данных
- 2) Обработка данных с помощью операторов
- 3) Вывод результата на экран

Арифметические действия

+ - * / для целых и вещественных чисел

div и **mod** - для целых чисел

div - целочисленное деление

$$10 \text{ div } 3 = 3$$

$$27 \text{ div } 10 = 2$$

mod – остаток от деления

$$10 \text{ mod } 3 = 1$$

$$27 \text{ mod } 10 = 7$$

Порядок действий

- 1) Вычисления значений в скобках
- 2) Вычисления значений функций
- 3) Операции *, /, **mod**, **div**
- 4) Операции + -

Решение задач

Задача1. Найти результат сложения двух переменных $s=a+b$

Program Primer;

var a,b,s: integer;

Begin

read (a,b); **//введите два числа**

s:=a+b;

write ('сумма чисел= ',s);

End.

Задача2.(самостоятельно) Написать программу вычисления площади прямоугольного треугольника по двум катетам

$$S=a*b/2$$

Задача3. (самостоятельно) Написать программу вычисления площади трапеции по высоте и двум основаниям $S=(a+b)/2*h$

Стандартные математические функции

Функция	Действия	Тип переменной	Тип функции
abs(x)	Модуль числа x	I,R	I,R
sqr (x)	Квадрат числа x	I,R	I,R
sqrt (x)	Корень из числа x	R	R
power(x,y)	x в степени y	R	R
inc (x,n)	Значением является x увеличенное на n	I	I
dec (x,n)	x уменьшенное на n	I	I

Функция	Действия	Тип переменной	Тип функции
frac (x)	Дробная часть	R	R
int (x)	Целая часть	R	R
round (x)	Округление x до следующего целого числа		
trunc(x)	Округление x , отбрасывает дробную часть		
random (x)	Значением функции является случайное число из (0,1)		

Задача1. (самостоятельно) Найти длину гипотенузы прямоугольного треугольника, если известна длина катетов $a^2=b^2+c^2$

Задача2. (самостоятельно) Найти S треугольника по трем сторонам по формуле Герона $p = \frac{a+b+c}{2}$

$$s = \sqrt{(p * (p - a) * (p - b) * (p - c))}$$

Условный оператор

Это оператор с помощью которого можно в зависимости от условия выполняется одно из нескольких действий.

If условие **then** действие1 **else** действие
2

Читать

Если условие **то** действие1 **иначе** действие2

Если условие истинно то выполняется
действие1

в противном случае выполняется действие2

Неполная форма условного оператора

If условие then действие1

Если условие истинно выполняется действие 1, в противном случае это действие пропускается и выполняется действие, следующий за условием.

«Вложенные» условные операторы

В условном операторе за служебными словами **then** и **else** могут следовать любые операторы языка Паскаль, в том числе и условные. **Пример:**

If $x > 0$ **then** $x := x - y$ **else**

В условном операторе возможно использование **логических операций**.

Not (не)

And (и)

Or (или)

Отношения стоящие слева и справа от знака логической операции должны быть заключены в скобки, т.к. логические операции имеют более высокий приоритет.

If $(x > 0)$ and $(x < 10)$ then $y := \text{sqr}(x)$

Задача1 : Найти наименьшее из двух чисел введенных с клавиатуры. ($a \neq b$)

Program Primer;

var a, b: integer;

Begin

read (a,b); //ввести два числа

If $a < b$ **then** write (a) **else** write (b);

end.

Задача2 : Найти наибольшее из трех введенных с клавиатуры чисел.($a \neq b \neq c$)

Program Primer;

var a, b,c,max: integer;

Begin

read (a,b,c); // ВВОД ДАННЫХ

If $a > b$ **then** max:=a **else** max:=b;

If max<c **then** max:=c; //ПОИСК
МАКСИМАЛЬНОГО

write (max);// печать результата

end.

Задача3 : Определить, является ли
треугольник со сторонами a, b, c
равнобедренным.

Program Primer;

var a, b, c: integer;

Begin

read (a, b, c);

If (a=b) or (a=c) or (b=c) **then** write ('треугольник
равнобедренный') **else** write ('треугольник не
равнобедренный') ;// печать результата

end.

Задача1. (самостоятельно) Даны два угла треугольника, напишите программу, которая определяет существует ли треугольник с такими углами ($a+b < 180$).

Задача2. (самостоятельно) Даны два угла треугольника, напишите программу, которая определяет является ли треугольник прямоугольным.

Задача3. (самостоятельно) Написать программу решения квадратного уравнения $ax^2+bx+c=0$.

Цикл с параметром

Цикл предназначен для повторения одной и той же последовательности команд до тех пор, пока не выполнится какое-либо условие.

Каждое повторение цикла называется – **ИТЕРАЦИЯ**.

Цикл с известным количеством повторений называется **циклом с параметром**.

For параметра := Н.З. **to** К.З. **do**} заголовок

begin

оператор

end;



тело цикла

do – значение параметра увеличивается на 1

downto - значение параметра уменьшается
на 1

ЧИТАТЬ

Для параметра **от** Н.З. **до** К.З. **выполнять**
оператор

Параметр – это переменная которая принимает значения от начального до конечного

Н.З. и К.З. – могут быть представлены константами, переменными, арифметическими выражениями.

До тех пор пока $\text{Н.З.} < \text{К.З.}$ параметр принимает значения от Н. до К. и для каждого из них выполняется тело цикла.

Как только $\text{Н.З.} > \text{К.З.}$ выполнение цикла закончится.

Задача1 : Найти сумму чисел из промежутка от 1 до 10.

```
Program Primer;
```

```
var i, s: integer;
```

```
Begin
```

```
s:= 0; // обнуляем сумму
```

```
for i:=1 to 10 do
```

```
begin
```

```
s:=s+i;
```

```
end;
```

```
write ('сумма чисел = ',s);
```

```
End.
```

Задача2: (самостоятельно) Найти произведение четных чисел из промежутка от 10 до 20

Определение четности (нечетности) чисел

Если $a \bmod 2 = 0$, то a - четное число
Если $a \bmod 2 = 1$, то a - нечетное число

```
Program Primer2;
```

```
var n, p: integer;
```

```
Begin
```

```
p:= 1; // произведению присваиваем значение 1
```

```
For n:=10 to 20 do
```

```
begin
```

```
If n mod 2=0 then p:=p*n;
```

```
end;
```

```
write ('произведение четных чисел = ',p);
```

```
End.
```

Задача3: (самостоятельно) Вывести на экран все двухзначные числа, кратные 3 и заканчивающиеся на 2 и 4.

Program Primer3;

var x: integer;

Begin

for x:=10 **to** 99 **do**

begin

If (x mod 3=0) and (x mod 10=2) or (x mod 10=4) **then write** (x, ' ');

end;

End.

Цикл с предусловием

Цикл с неопределенным количеством повторений.

while условие **do**

begin

оператор

end;

ЧИТАТЬ:

Пока выполняется условие **выполнять**
оператор

Число повторений оператора цикла не известно, а задается некоторое условие его окончания

Условие - логическое выражение ($<$, $>$, $=>$, $=<$, $<>$)

Перед каждой итерацией цикла условие проверяется и если оно истинно, то выполняется оператор, в противном случае происходит выход из цикла.

Цикл с постусловием

repeat операторы **until** условие

ЧИТАТЬ:

повторять операторы **до тех пор, пока**
не выполнится условие

Проверка условия производится после очередного выполнения тела цикла (оператора), условие проверяется до тех пор пока оно не выполнится (т.е. пока условие ложно цикл будет повторяться)

Тело цикла должно содержать хотя бы один оператор, влияющий на условие окончания, иначе цикл будет бесконечным.

Из-за того что в циклах этого типа сначала выполняется тело цикла затем проверка условия, в них бывает трудно найти ошибку. Лучше использовать циклы с предусловием.

Задача1: (while) Найти сумму цифр натурального числа.

Program Primer1;

var a,n, s: integer;

Begin

read (n);

s:=0;

while n <> 0 **do** // пока n не равно 0 выполнять

begin

a:= n mod 10;

s:= s+a;

n:=n div 10;

end;

write ('сумма цифр числа', '=',s);

End.

Задача2: (repeat) Найти сумму цифр натурального числа.

Program Primer2;

var a,n, s: integer;

Begin

read (n);

s:=0;

repeat // повторять

a:= n mod 10;

s:= s+a;

n:=n div 10;

until n=0 // пока n не станет = 0

write ('сумма цифр числа', '=',s);

End.

Задача3: (while) Найти сумму четных чисел последовательности от 0 до 20.

Program Primer3;

var i, s: integer;

Begin

i:=0; //первая цифра последовательности

s:=0; // обнуляем сумму

while i < 20 **do** // пока i меньше 20 выполнять

begin

i:= i+2;

s:= s+i;

end;

write ('сумма четных чисел = ',s);

End.

Задача4: (repeat) Найти сумму четных чисел последовательности от 0 до 20.

Program Primer4;

var i, s: integer;

Begin

i:=0;

s:=0;

repeat // повторять

i:= i+2;

s:= s+i;

until i>19 // пока i не станет > 19

write ('сумма четных чисел = ',s);

End.

Задача5. Найти первую цифру числа (while)

Program Primer5;

var a: integer;

Begin

read (a);

while a>10 **do** //пока a больше 10 выполнять

begin

a:=a div 10;

end;

writeln (a);

End.

Вложенные циклы

Циклы могут быть простыми или **вложенными** (цикл в цикле). Внутренний и внешний циклы могут быть любыми из трех рассмотренных ранее видов.

Как работает вложенный цикл- задается начальное значение параметра внешнего цикла, затем начинает исполняться внутренний цикл при этом параметр второго цикла принимает все значения из указанного интервала.

Задача1: Вывести на экран
последовательность всех
положительных двухзначных чисел.

Любое число состоит из цифр от 0 до 9,
используя операторы вложенного цикла,
можно составить любое число

Program Primer1;

var i, j: integer;

Begin

for i:=1 **to** 9 **do** //внешний цикл

for j:=0 **to** 9 **do** // внутренней цикл

writeln (i,j);

End.

Задача2. Найти все целые трехзначные числа произведение цифр которых равно заданному числу.

Program Primer2;

var a,b,c,p,d: integer;

Begin

read (d); // заданное числа

for a:=1 **to** 9 **do** //первая цифра числа

for b:=0 **to** 9 **do** // вторая цифра числа

for c:=0 **to** 9 **do** // третья цифра числа

begin

p:=a*b*c;

if p=d **then writeln** (a,b,c);

end;

End.

Задача3. Найти сумму цифр двухзначных чисел (положительных)

```
Program Primer3;
```

```
var i, j, s: integer;
```

```
Begin
```

```
for i:=1 to 9 do //первая цифра числа
```

```
for j:=0 to 9 do // вторая цифра числа
```

```
begin
```

```
s:=i+j;
```

```
writeln (s);
```

```
end;
```

```
End.
```

Решение задач

Задача1. Найти все двухзначные числа, которые при умножении на 3 заканчиваются на 4

```
Program Primer1;
```

```
var a,b: integer;
```

```
Begin
```

```
for a:=10 to 99 do
```

```
begin
```

```
b: =a*3; //увеличиваем число в 3 раза
```

```
If b mod 10=4 then writeln (a);
```

```
end;
```

```
End.
```

Задача3. Найти все трехзначные числа, в которых ровно две одинаковых цифры.

```
Program Primer3;
```

```
var a,b,c: integer;
```

```
Begin
```

```
For a:=1 to 9 do //первая цифра числа
```

```
for b:=0 to 9 do // вторая цифра числа
```

```
for c:=0 to 9 do // третья цифра числа
```

```
Begin
```

```
If (a<>b) and (b<>c) then
```

```
If (a=b) or (a=c) or (b=c) then writeln (a,b,c);
```

```
end;
```

```
End.
```

Задача4. Найти все четырехзначные числа, в которых суммы двух первых цифр равна x , сумма третьей и четвертой цифры равна y (x, y вводятся с клавиатуры).

Program Primer3;

var a,b,c,d,s,s1,x,y: integer;

Begin

read (x, y);

For a:=1 **to** 9 **do** //первая цифра числа

for b:=0 **to** 9 **do** // вторая цифра числа

for c:=0 **to** 9 **do** // третья цифра числа

for d:=0 **to** 9 **do** // четвертая цифра числа

Begin

s:=a+b;

s1:=c+d;

If (s=x) and (s1=y) **then** **writeln** (a,b,c,d);

end;

End.

Контрольная работа

«Циклы, вложенные циклы»

Вариант 1

1. Найти все целые двухзначные числа, удовлетворяющие условию остаток от деления на 3 равен 2
2. Найти все трехзначные числа, которые при делении на 2 дают в остаток 1, при делении на 3 – остаток 2, при делении на 4 – остаток 3.
3. Найти все симметричные четырехзначные числа. Например: 7667, 1331

Вариант 2

1. Найти все целые двухзначные числа, разность цифр которых равна 3 (по модулю)
2. Найти все трехзначные числа, которые при увеличении на 1 делятся на 2, при увеличении на 2 делятся на 3 при увеличении на 3 делятся на 4.
3. Найти все трехзначные числа, которые состоят из разных цифр, а их сумма равна a (a вводится с клавиатуры)