

ВВЕДЕНИЕ

Основным направлением обучения является изучение базовых алгоритмических конструкций, основных операторов и их использования.

В качестве основного языка программирования выбран Паскаль, т. к. он наиболее удачно используется для переноса блок-схем алгоритмов на алгоритмический язык.

Алгоритмический язык программирования Паскаль не требует чрезмерных усилий для усвоения его основных синтаксических конструкций, легко усваивается в процессе практической работы на любом персональном компьютере. Вместе с тем Паскаль обладает большими возможностями для решения широкого круга практических задач и овладение этим языком можно рассматривать как ценный навык алгоритмизации и программирования.

По этой причине предлагаемое методическое пособие может быть использовано студентами младших курсов физического, математического факультетов для приобретения практических навыков алгоритмизации и программирования, может быть использовано студентами заочниками для самостоятельного овладения языком программирования.

Основная цель пособия - помочь студентам в проведении самостоятельной работы, которая может быть реализована в форме типового расчета.

Пособие содержит примеры решения задач и задачи по программированию на языке Паскаль. Всего предложено 25 вариантов, содержащих по 15 задач и охватывающих следующие темы: линейные программы, операторы выбора и условного перехода, циклы, процедуры, функции пользователя.

К каждому разделу приводятся несколько типовых задач с решениями, что позволяет студентам самостоятельно выполнить типовой расчет.

В настоящем пособии приводится список литературы, которая может быть использована при работе над типовым расчетом.

ВАРИАНТЫ ЗАДАНИЙ

№ варианта	Разделы														
	I			II				III				IV		V	
1	1	6	13	1а	2	8	22	1	7	12	18	1	11	1	11
2	2	7	14	1б	3	9	23	2	8	13	19	2	12	2	12
3	3	8	15	1в	4	10	24	3	9	14	20	3	13	3	13
4	4	9	16	1г	5	11	25	4	10	15	21	4	14	4	14
5	5	10	17	1д	6	12	26	5	11	16	22	5	15	5	15
6	1	11	18	1е	7	13	27	6	7	17	23	6	16	6	16
7	2	12	19	1ж	2	14	28	1	8	12	18	7	17	7	17
8	3	6	20	1з	3	15	29	2	9	13	19	8	18	8	18
9	4	7	21	1и	4	16	30	3	10	14	20	9	19	9	19
10	5	8	22	1к	5	17	22	4	11	15	21	10	20	10	20
11	1	9	23	1а	6	18	23	5	7	16	22	1	11	1	11
12	2	10	24	1б	7	19	24	6	8	17	23	2	12	2	12
13	3	11	25	1в	2	20	25	1	9	12	18	3	13	3	13
14	4	12	13	1г	3	21	26	2	10	13	19	4	14	4	14
15	5	6	14	1д	4	8	27	3	11	14	20	5	15	5	15
16	1	7	15	1е	5	9	28	4	7	15	21	6	16	6	16
17	2	8	16	1ж	6	10	29	5	8	16	22	7	17	7	17
18	3	9	17	1з	7	11	30	6	9	17	23	8	18	8	18
19	4	10	18	1и	2	12	22	1	10	12	18	9	19	9	19
20	5	11	19	1к	3	13	23	2	11	13	19	10	20	10	20
21	1	12	20	1а	4	14	24	3	7	14	20	1	11	1	11
22	2	6	21	1б	5	15	25	4	8	15	21	2	12	2	12
23	3	7	22	1в	6	16	26	5	9	16	22	3	13	3	13
24	4	8	23	1г	7	17	27	6	10	17	23	4	14	4	14
25	5	9	24	1д	2	18	28	1	11	12	18	5	15	5	15

I. ЛИНЕЙНЫЕ ПРОГРАММЫ

Типы переменных в Паскале:

INTEGER – целый тип (диапазон -32768...32767)

REAL – действительный, вещественный тип (диапазон $2.9 \cdot 10^{-39} \dots 1.7 \cdot 10^{+38}$)

CHAR – литерный, символьный тип

BOOLEAN – логический тип

BYTE – целые числа от 0 до 255

LONGINT – целый тип (диапазон -2147483648... 2147483647)

STRING – строковый тип (от 0 до 255 символов)

Стандартные операции и функции от скалярных данных.

Функция, операция	Назначение	Тип переменных	Тип результата
A+B	Сумма	REAL, INTEGER	REAL, INTEGER
A-B	Разность	REAL, INTEGER	REAL, INTEGER
A*B	Произведение	REAL, INTEGER	REAL, INTEGER
A/B	Частное	REAL, INTEGER	REAL
ABS(x)	Абсолютное значение	REAL, INTEGER	REAL, INTEGER
SQR(x)	Вычисление x^2	REAL, INTEGER	REAL, INTEGER
SIN(x)	Нахождение $\sin x$	REAL, INTEGER	REAL
COS (x)	Вычисление $\cos x$	REAL, INTEGER	REAL
ARCTAN(x)	Вычисление $\arctan x$	REAL, INTEGER	REAL
EXP(x)	Вычисление экспоненты e^x	REAL, INTEGER	REAL
LN(x)	Вычисление $\ln x$	REAL, INTEGER	REAL
SQRT(x)	Вычисление \sqrt{x}	REAL, INTEGER	REAL
A div B	Нахождение целой части при делении A на B	INTEGER	INTEGER
A mod B	Нахождение остатка при деле-	INTEGER	INTEGER

	нии А на В		
TRUNC(x)	Нахождение целой части x	REAL, INTEGER	INTEGER
ROUND(x)	Округление x в сторону ближайшего целого	REAL, INTEGER	INTEGER
PI	Зарезервированная константа число π		REAL

Некоторые математические функции, не реализованные в языке Паскаль

В математике	В программировании на языке Паскаль
x^n	exp(n*ln(x))
$\log_b a$	ln(a)/ln(b)
tg x	sin(x)/cos(x)
ctg x	cos(x)/sin(x)

ОПЕРАТОРЫ:

Оператор	Назначение	Синтаксис
Readln	ввод значений с клавиатуры с переводом курсора на новую строку	Readln(перемен1, ..., переменN)
Writeln	вывод значений на экран с переводом курсора на новую строку	Writeln('Текст') Writeln(перемен1, ..., переменN)
Read	ввод значений с клавиатуры	Read(перемен1, ..., переменN)
Write	вывод значений на экран	Write('Текст') Write(перемен1, ..., переменN)

ПРИМЕРЫ РЕШЕНИЯ ЗАДАЧ:

Пример 1.

Составить программу расчета значения функции.

$$Z = |\cos x^4 - 3 \operatorname{tg} x^2| + 0.8 \sin ux^2 + 10 \text{ при любых значениях } x \text{ и } u.$$

Результат вывести в виде: при $x=$ и $u=...$ $z=...$

Решение

Используемые переменные: x, u - аргументы, z – значение функции

```
Program pr1;
```

```
  Var x,u,z: real;
```

```
Begin
```

```
  writeln('введите X Y');
```

{вывод строки подсказки}

```

readln (x,y);                                {ввод аргументов x и y}
z:=abs(cos(sqrt(x)*sqrt(x)-3*sin(sqrt(x))/cos(sqrt(x))))+0.8*sin(y*sqrt(x))+10;
writeln('при x=',x:8:2,' y=',y:8:2,' z=',z:8:2);    {вывод результата}
readln;                                       {задержка до нажатия клавиши Enter}
End.

```

Результат

введите X Y

1 2

при x=1.00 y=2.00 z=11.59

Пример 2.

Вводится вещественное число a . Не пользуясь никакими арифметическими операциями, кроме сложения, получить $7a$ за четыре операции.

Решение

Используемые переменные: **a** – вводимое число,

b, c, d – вспомогательные переменные

```

Program pr2;
Var  a,b,c,d:real;
Begin
  write('введите a ');
  readln (a);                                {ввод исходного числа}
  b:=a+a;                                     {2a}
  c:=b+b;                                     {4a}
  d:=b+c;                                     {6a}
  a:=d+a;                                     {7a}
  writeln('7a=',a:8:2);                      {вывод результата}
  readln;                                    {задержка до нажатия клавиши Enter}
End.

```

Результат

введите a 2

7a= 14.00

Пример 3.

Найти площадь круга и длину окружности.

Решение

Используемые переменные: **r** - радиус, **d** – длина окружности,

s – площадь круга

```

Program pr3;
Var d,r,s:real;
Begin
  write('введите радиус окружности ');
  readln (r);           {ввод радиуса}
  d:= 2*Pi*r;          {вычисление длины окружности}
  s:=Pi*sqr(r);        { вычисление площади круга}
  writeln('длина окружности= ',d:4:2);  {вывод результата}
  writeln('площадь окружности= ',s:4:2);
  readln;              {задержка до нажатия клавиши Enter}
End.

```

Результат

введите радиус окружности 5
длина окружности=31.42
площадь окружности=78.54

Пример 4.

Вычисление суммы цифр введенного натурального двузначного числа.

Решение

Используемые переменные: **n** - двузначное число, **a, b** – цифры числа

```

Program pr4;
Var n, a, b: integer;
Begin
  write('n= '); readln(n);      {ввод исходного двузначного числа}
  a:=n div 10;                 {1-я цифра}
  b:=n mod 10;                 {2-я цифра}
  writeln('сумма = ', a+b);     {вывод результата}
  readln;                      {задержка до нажатия клавиши Enter}

End.

```

Результат

n=48
сумма=12

Пример 5.

Введенное натуральное 4-значное число изменить так, чтобы 2 и 3 цифры поменялись местами.

Решение:

Четырехзначное число N можно представить в виде суммы разрядных слагаемых: $N=n_1*1000+n_2*100+n_3*10+n_4$, где n_1, n_2, n_3, n_4 – цифры соответствующих разрядов. Например, $3562=3*1000+5*100+6*10+2$

Чтобы во введенном числе N поменять цифры местами, нужно выделить каждую цифру и записать число в виде $N=n_1*1000+n_3*100+n_2*10+n_4$

Используемые переменные: N – вводимое четырехзначное число,
 n_1, n_2, n_3, n_4 – цифры

```
Program pr5;
Var N, n1, n2, n3, n4:integer;
Begin
  write('введите n ');
  readln (n);           {ввод исходного 4-значного числа}
  n1:=N div 1000;      {1-я цифра числа}
  n2:=N div 100 mod 10; {2-я цифра числа}
  n3:=N div 10 mod 10; {3-я цифра числа}
  n4:=N mod 10;        {4-я цифра числа}
  n:= n1*1000+n3*100+n2*10+n4; {получение числа в виде суммы разрядных
                                слагаемых}
  writeln('результат ', n); {вывод результата}
  readln;               {задержка до нажатия клавиши Enter}
End.
```

Результат:

введите n 1234

результат 1324

Пример 6.

Обмен значениями переменных X и Y .

Решение:

Для того, чтобы переменные X и Y поменялись своими значениями, можно использовать вспомогательную переменную, например, T . Вспомогательная переменная нужна для того, чтобы сохранить временно значение переменной X . После этого в переменную X можно занести значение переменной Y , а Y присвоить значение X .

Используемые переменные: X, Y – вводимые числа,
 T – вспомогательная переменная

```
Program pr6;
Var X, Y, T: integer;
begin
  write('Введите X Y ');
  readln(X, Y);           {ввод исходных чисел}
  T:=X;
```

```

X:=Y;
Y:=T;
writeln('X=', X, 'Y=',Y); {вывод результата}
readln; {задержка до нажатия клавиши Enter}
end.

```

Результат:

Введите X Y 3 7
X=7 Y=3

Пример 7.

Вычислить значение $y = (3 \operatorname{tg} x^2 - e^x) / 2 \sin x$ для $x = -34, 0.89, 1.23$

Решение:

Используемые переменные: **X** – аргумент, **Y** – значение функции

```

Program pr7;
Var x,y:real;
Begin
  Write('введите x ');
  readln (x); {ввод аргумента}
  y:=(3*sin(sqrt(x))/cos(sqrt(x))-exp(x))/(2*sin(x)); {вычисление значения функции}
  Writeln('при x=',x:4:2,' y=',y:4:2); {вывод результата}
  readln; {задержка до нажатия клавиши Enter}

End.

```

Результат:

1 случай:

введите x 34
при x=34.00 y=-551389941339996.31

2 случай:

введите x 0.89
при x=0.89 y=0.39

3 случай:

введите x 1.23
при x=1.23 y=25.64

ЗАДАНИЯ

1. Составить программу расчета значения функции $Z = |3 e^x + 3 - 2 \ln xy| + 1,8x^2 + 1$ при любых значениях x и y . Результат вывести в виде: при $x = \dots$ и $y = \dots$ $z = \dots$
2. Составить программу расчета значения функции $Z = \operatorname{tg} x^3 - |2 \sin x^2 y + 7.8 \cos x| + 10$ при любых значениях x и y . Результат вывести в виде: при $x = \dots$ и $y = \dots$ $z = \dots$
3. Составить программу расчета значения функции $Z = (x^4 - 2 \operatorname{ctg} x^2 y^2) / |8x^2 - 5 \operatorname{arctg} y|$ при любых значениях x и y . Результат вывести в виде: при $x = \dots$ и $y = \dots$ $z = \dots$
4. Составить программу расчета значения функции $Z = \operatorname{tg} x^5 - 21 \operatorname{ctg} xy^2 / (18 \sin x - 13 \cos y + 11)$ при любых значениях x и y . Результат вывести в виде: при $x = \dots$ и $y = \dots$ $z = \dots$
5. Составить программу расчета значения функции $Z = (2e^x - 4 - |\sin x^2 y|) / (2.1 \operatorname{ctg} x^2 + 10.2)$ при любых значениях x и y . Результат вывести в виде: при $x = \dots$ и $y = \dots$ $z = \dots$
6. Вводится вещественное число a . Не пользуясь никакими арифметическими операциями, кроме умножения, получить a^4 за две операции.
7. Вводится вещественное число a . Не пользуясь никакими арифметическими операциями, кроме умножения, получить a^6 за три операции.
8. Вводится вещественное число a . Не пользуясь никакими арифметическими операциями, кроме умножения, получить a^7 за четыре операции.
9. Вводится вещественное число a . Не пользуясь никакими арифметическими операциями, кроме умножения, получить a^8 за три операции.
10. Вводится вещественное число a . Не пользуясь никакими арифметическими операциями, кроме умножения, получить a^9 за четыре операции.
11. Вводится вещественное число a . Не пользуясь никакими арифметическими операциями, кроме умножения, получить a^{10} за четыре операции.
12. Вводится вещественное число a . Не пользуясь никакими арифметическими операциями, кроме сложения, получить $6a$ за три операции.
13. Найти произведение цифр введенного 4-значного числа.
14. Найти сумму цифр введенного 4-значного числа.
15. Определить сумму квадратов цифр введенного 3-значного числа.
16. Введено 3-значное число. Вывести число в зеркальном отображении.
17. Введено 3-значное число. Вывести число в зеркальном отображении
18. Введенное 4-значное число изменить так, чтобы 1 и 4 цифры поменялись местами.
19. Найти среднее арифметическое цифр введенного 3-значного числа.
20. Введено 3-значное число. Поменять местами 1-ю и 3-ю цифры числа.
21. Введенное 4-значное число изменить так, чтобы 1-3 и 2-4 цифры поменялись местами.
22. Найти полусумму цифр введенного 4-значного числа.
23. Определить сумму квадратов цифр введенного 3-значного числа.

24. Введенное 4-значное число изменить так, чтобы 1 и 3 цифры поменялись местами.

25. Введенное 4-значное число изменить так, чтобы 3 и 4 цифры поменялись местами.

II. ОПЕРАТОРЫ ВЫБОРА И УСЛОВНОГО ПЕРЕХОДА.

ОПЕРАТОРЫ:

Оператор	Назначение	Синтаксис
if ... then...	проверка условий	if <условие> then <оператор1> else <оператор2> if <условие> then <оператор>
case	выбор	case <переменная> of <константа 1>: <операторы 1>; ... <константа n>: <операторы n> [else <операторы>] end;

ПРИМЕРЫ РЕШЕНИЯ ЗАДАЧ

Пример 1.

Выбор минимального из трех введенных целых чисел.

Решение

Для того, чтобы выбрать минимальное из трех введенных чисел, нужно выбрать наименьшее сначала из двух и запомнить в переменную *m*, а затем значение *m* сравнить с третьим числом.

Используемые переменные: **a**, **b**, **c** – вводимые числа, **m** – минимальное из них

```
Program pr1;
```

```
Var a, b, c, m: integer;
```

```
BEGIN
```

```
  Write('a, b, c='); readln(a, b, c);      {ввод исходных чисел}  
  if a<b then m:=a else m:=b;             {выбор минимального из a и b}  
  if c<m then m:=c;                       {сравнение с третьим числом}  
  writeln('Min=', m);                    {вывод результата}  
  readln
```

```
END.
```

Результат:

1 случай:

a, b, c= 15 6 9

Min=6

2 случай:

a, b, c= 1 45 4

Min=1

3 случай:

a, b, c= 25 16 9

Min=9

Пример 2.

Решение квадратного уравнения вида $Ax^2 + Bx + C = 0$.

Решение

Используемые переменные: **a, b, c** – коэффициенты квадратного уравнения, **d** - дискриминант, **x1, x2** – корни уравнения

```
Program pr2;
```

```
Var a, b, c, d, x1, x2: real;
```

```
Begin
```

```
  Write('a, b, c='); readln(a, b, c); {вводим коэффициенты квадратного уравнения}
```

```
  d:=sqr(b)-4*a*c; {вычисляем дискриминант}
```

```
  if d>0 then begin {если дискриминант положительный, то вычисляем два корня}
```

```
    x1:=(-b+sqr(d))/(2*a); writeln('x1=', x1:6:2);
```

```
    x2:=(-b-sqr(d))/(2*a); writeln('x2=', x2:6:2)
```

```
  end
```

```
  else if d=0 then begin { иначе, если дискриминант равен 0, то вычисляем один корень}
```

```
    x1:=-b/(2*a); writeln('x=', x1:6:2)
```

```
  end
```

```
  else writeln('Корней нет'); {иначе выводим сообщение 'Корней нет'}
```

```
  readln
```

```
End.
```

Результат:

1 случай:

a, b, c= 1 -2 1

x= 1.00

2 случай:

a, b, c= 1 -6 8

x1= 4.00

x2= 2.00

3 случай:

a, b, c= 5 1 2

Корней нет

Пример 3.

Определить, есть ли в записи трехзначного числа хотя бы одна нечетная цифра.

Решение

Используемые переменные: **a**- вводимое трехзначное число,

a1, a2, a3 – его цифры

Program pr3;

Var a, a1, a2, a3: integer;

Begin

Write('a='); readln(a); *{ввод исходного числа}*

a1:= a div 100; *{1-я цифра}*

a2:= (a div 10) mod 10; *{2-я цифра}*

a3:= a mod 10; *{3-я цифра}*

if (a1 mod 2=1) or (a2 mod 2=1) or (a3 mod 2=1) *{Если 1-я цифра нечетная, или 2-я нечетная, или 3-я нечетная, то выводим 'Yes', иначе выводим 'No'}*
then writeln('Yes') else writeln('No');

readln;

End.

Результат:

1 случай:

a=418

Yes

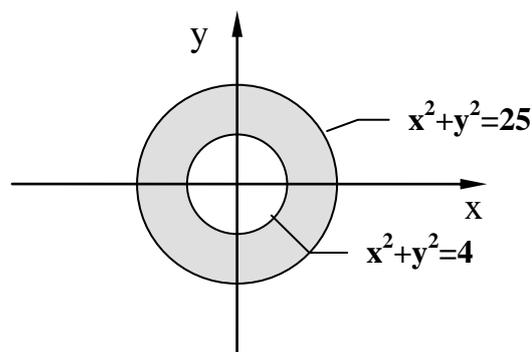
2 случай:

a=246

No

Пример 4.

Вводятся координаты точки. Определить попадает ли точка в заштрихованную область или нет.



Решение

Точка будет принадлежать заштрихованной области, если она лежит **внутри** большого круга ($x^2+y^2 < 25$) **и**, одновременно, **за** пределами малого круга ($x^2+y^2 > 4$).

```
Program pr4;
Var x,y: real;
Begin
  Write('введите x y ');
  readln (x,y);           {ввод координат точки}
  if (sqr(x)+sqr(y)<25) and (sqr(x)+sqr(y)>4) {проверка условия принадлежности}
  then writeln('точка попадает в заштрихованную область')
  else writeln('точка не попадает в заштрихованную область');
  readln;
end.
```

Результат:

1 случай:

введите x y 1 1

точка попадает в заштрихованную область

2 случай:

введите x y 2 5

точка не попадает в заштрихованную область

Пример 5.

Введено трехзначное число. Изменить число, увеличив все четные цифры на 2, а цифру 8 заменить на 0.

Решение

Для решения задачи необходимо разбить его на цифры. Каждую цифру следует изменить следующим образом: если цифра четная и меньше 8, то увеличить ее на 2, иначе, если цифра 8, то заменить ее на 0.

Используемые переменные: **n** – вводимое число, **a,b,c** – его цифры

```
Program pr5;
Var n,a,b,c:Integer;
begin
  write('введите n');   readln (n);           {ввод числа}
  a:=n div 100;         {1-я цифра}
  b:=n div 10 mod 10;  {2-я цифра}
  c:=n mod 10;         {3-я цифра}
  {проверяем каждую цифру и изменяем ее}
  if (a mod 2=0) and (a<8) then a:=a+2 else if a=8 then a:=0;
  if (b mod 2=0) and (b<8) then b:=b+2 else if b=8 then b:=0;
```

```

if (c mod 2=0) and (c<8) then c:=c+2 else if c=8 then c:=0;
if (d mod 2=0) and (d<8) then d:=d+2 else if d=8 then d:=0;
n:=a*1000+b*100+c*10+d;  {формируем число из измененных цифр}
writeln('n= ',n);      {вывод результата}
readln;
end.

```

Результат:

введите n1824
результат 1046

Пример 6.

Составить программу, которая выводит меню и выполняет указанные действия с заданными целыми числами:

- 1 - произведение двух чисел
- 2 - частное двух чисел
- 3 - сумма двух чисел

Используемые переменные: **a,b** – вводимые числа, **n** – номер операции

Решение

```

program pr6;
var a,b,n: integer;
begin
  writeln('1 – произведение двух чисел');
  writeln('2 – частное двух чисел');
  writeln('3 – сумма двух чисел');
  write('введите номер операции ');
  readln(n);
  write('введите два числа');  readln(a,b);
  case n of
    1: writeln('произведение=',a*b);
    2: writeln('частное=',a/b:4:2);
    3: writeln('сумма=',a+b);
    else writeln('не верный номер');
  end;
  readln;
end.

```

Результат:

1 – произведение двух чисел
2 – частное двух чисел
3 – сумма двух чисел
введите номер операции 2
введите два числа 3 5
частное=0.60

Пример 7.

Ввести первую букву времени года и вывести соответствующее название времени года на русском языке.

Решение:

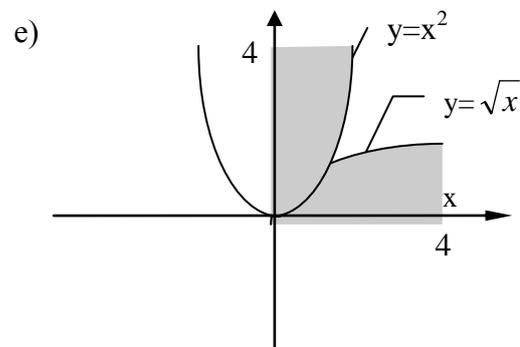
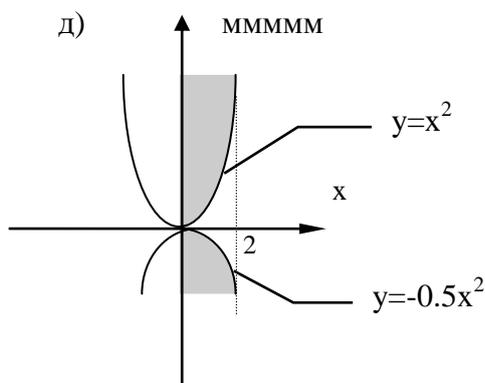
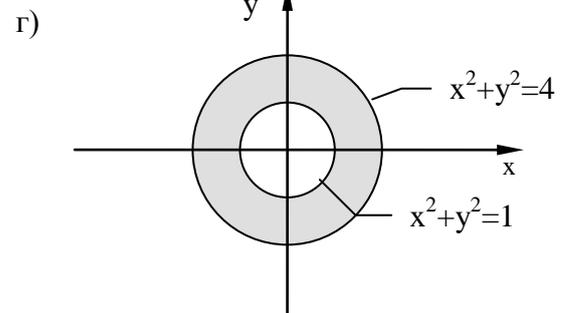
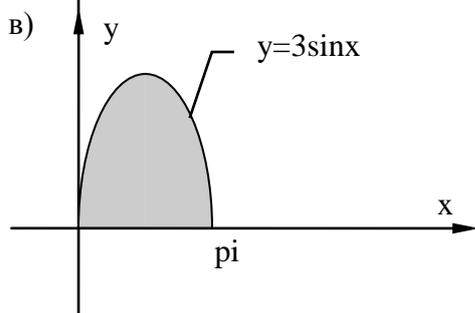
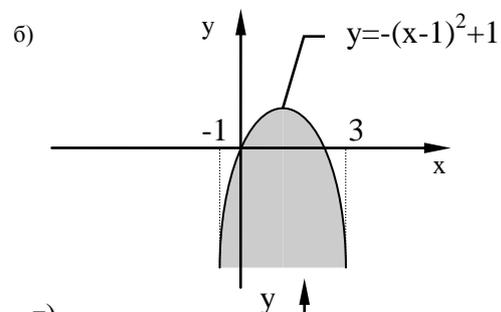
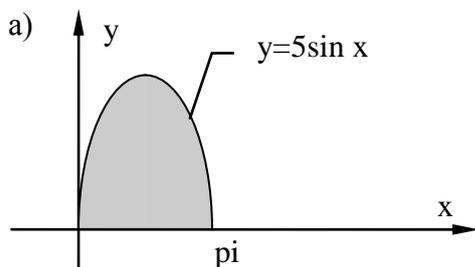
```
Program pr7;  
  Var N: char;  
Begin  writeln ('введи первую букву времени года ');  
      Readln (N);  
      Case N of  
        'з' : writeln ('зима');  
        'в' : writeln ('весна');  
        'л' : writeln ('лето');  
        'о' : writeln ('осень')  
      end.  
end.
```

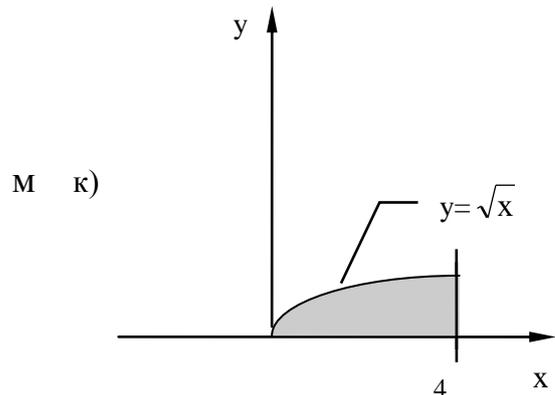
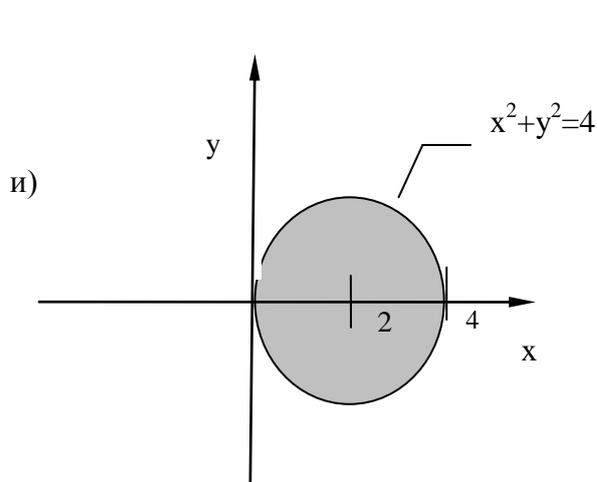
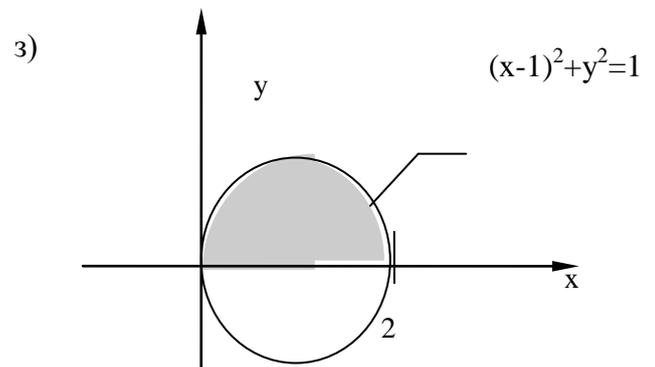
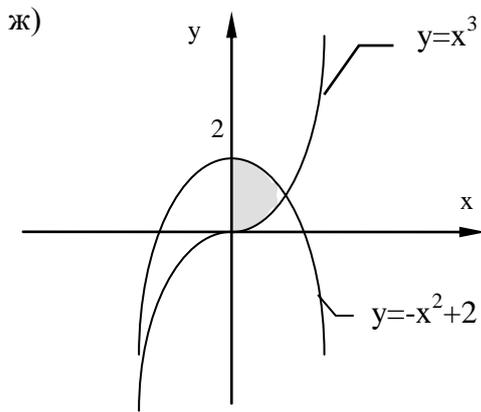
Результат:

введи первую букву времени года о
осень

ЗАДАНИЯ

1. Вводятся координаты точки. Определить попадает ли точка в заштрихованную область или нет.





2. Введено трехзначное число. Найти сумму четных цифр.
1. Введено четырехзначное число. Содержится ли в записи этого числа цифра 7?
2. Введено четырехзначное число. Найти среднее арифметическое нечетных цифр.
3. Введено трехзначное число. Если в записи числа встречается цифра 5, то записать число в зеркальном отображении.
4. Введено трехзначное число. Если сумма его цифр нечетна, то увеличить число вдвое.
5. Введено четырехзначное число. Найти сумму цифр, кратных 3-м.
6. Вводятся X и Y. Если хотя бы одно из этих чисел положительно, то найти их произведение. Иначе – найти их сумму.
7. Вводятся X и Y. Если X больше Y, то произвести их обмен.
8. Из чисел A, B, C, D выбрать максимальное.
9. Даны три действительных числа. Возвести в квадрат те из них, значения которых неотрицательны.
10. Даны два числа A и B. Выяснить, делится ли A на B.
11. Вводится число. Если оно положительное, то уменьшить его в 10 раз.
12. Вводятся X и Y. Заменить большее из этих чисел разностью большего и меньшего.
13. Определить, принадлежит ли введенное X интервалу (-10, 15)?
14. Вводится натуральное X. если оно четное, то уменьшить его вдвое, если нечетное – увеличить на 10.
15. Вводятся X и Y. Заменить меньшее из них суммой этих чисел.
16. Из чисел A, B, C, D выбрать минимальное.

17. Определить, сколько среди заданных чисел А, В, С, D отрицательных.
18. Сколько среди заданных чисел А, В, С, D нечетных.
19. Сколько среди заданных чисел А, В, С, D четных.
20. Составить программу, которая выводит меню и выполняет указанные действия с заданными целыми числами:
- 1 - сумма двух чисел
 - 2 - разность двух чисел
 - 3 - конец работы
23. Составить программу, которая по введенной начальной букве выводит название цветов радуги (красный, оранжевый, желтый, зеленый, голубой, синий, фиолетовый).
24. Составить программу, которая выводит меню и при нажатии нужной цифры вычисляет соответствующее выражение:
- 1 - $y = x^3 + x$
 - 2 - $y = (x + 1)^2$
 - 3 - $y = \sqrt{x + 4}$
25. Составить программу, которая по введенному порядковому номеру выводит название дня недели.
26. Составить программу, которая позволяет ввести номер месяца и вывести его название.
27. Составить программу, которая выводит меню и при нажатии вводе цифры вычисляет соответствующее выражение:
- 1 - $y = (a + b)^2$
 - 2 - $y = a^2 - b^2$
 - 3 - $y = 1 / (a + b)$
28. Составить программу, которая выводит меню и при вводе буквы вычисляет соответствующую функцию:
- T - $Tg x$
 - S - $Sin x$
 - C - $Cos x$
29. Составить программу, которая по введенной заглавной букве выводит название столиц мира: Москва, Париж, Лондон, Рим.
30. Составить программу, которая выводит меню и при вводе буквы вычисляет соответствующую функцию:
- A - $Abs x$
 - S - $Sin x$
 - C - $Cos x$

III. ЦИКЛЫ

ОПЕРАТОРЫ:

Оператор	Назначение	Синтаксис
For	Цикл с параметром	For <перем>:=<начальное знач> to <кон.знач> do <оператор>; Значение переменной изменяется от начального значения к конечному с шагом 1. For <перем>:=<нач. знач> downto <конеч. знач> do <оператор>; Значение переменной изменяется от начального значения к конечному с шагом -1
While	Цикл с предусловием	While <условие> do <оператор>; Цикл выполняется, пока условие истинно.
Repeat Until	Цикл с постусловием	Repeat <Операторы>; Until <условие>; Цикл выполняется, пока условие ложно.

Пример 1.

Вывести квадраты первых десяти натуральных чисел.

Решение

Используемые переменные: i – натуральные числа, x – их квадраты

```
Program pr1;
```

```
Var i, x: integer;
```

```
begin
```

```
  for i:=1 to 10 do begin      {перебираем натуральные числа от 1 до 10}
```

```
    x:=sqr(i);                {возводим очередное число в квадрат}
```

```
    write(x, ' ');           {выводим полученное значение}
```

```
  end;
```

```
  readln
```

```
end.
```

Результат

1 4 9 16 25 36 49 64 81 100

Пример 2.

Найти сумму $1 + 1/3 + 1/5 + \dots$ (N слагаемых).

Решение

```
Program pr2;
```

```
Var I, N: integer;
```

```
    S: real;
```

```
begin
```

```

    Write('N='); Readln(N); {вводим количество слагаемых}
    S:=0; {обнуляем сумму}
    {выполняем цикл N раз, добавляя к сумме по одному слагаемому}
    For I:=1 to N do
        S:=S+1/(2*I-1);
    Writeln('S=',S:5:2); {выводим результат с двумя десятичными знаками}
    Readln
end.

```

Результат:

N=4

S= 1.68

Пример 3.

Дано натуральное число n. Найти сумму $S=1+2/3+3/7\dots$ (n слагаемых)

Решение

```

Program pr3;
var i,b,n:integer;
    s:real;
begin
    write('Введите n ');
    readln(n); {вводим количество слагаемых}
    s:=1; {первоначальное значение суммы}
    b:=3; {первоначальное значение знаменателя}
    for i:=2 to n do
        begin
            s:=s+ i/b; { добавляем к сумме по очередное слагаемое}
            b:=b+4; { вычисляем следующий знаменатель}
        end;
    writeln('сумма=',s:4:2); {выводим результат с двумя десятичными знаками}
    readln;
end.

```

Результат:

Введите n 5

сумма=2.79

Пример 4.

Дано натуральное число n. Вычислить $3^1+3^2+\dots+3^n$

Решение:

```

program pr4;
var i,n:integer;
    s:real;

```

```

begin
  write('введите n ');
  readln(n);
  s:=0;
  for i:=1 to n do
    s:=s+ exp(i*ln(3));
  writeln('сумма=',s:4:0);
  readln;
end.

```

Результат:

введите n 5

сумма= 363

Пример 5.

Найти сумму ряда: $S = x - x^2/4 + x^3/9 - x^4/16...$ (n слагаемых).

Решение:

Program Pr5;

Var i, n, z: integer;

p, s: real;

```

begin
  writeln ('введите n'); readln ( n );
  writeln('введите x'); readln( x );
  z:=1; p:=x; s:=0;
  for i:=1 to n do begin
    s:= s + z*p/sqr(i); p:=p*x; z:=-z
  end;
  writeln ('S=', S :6:2);
  readln
end.

```

Пример 6.

Вычислить произведение:

$(1 + \sin 2) * (2 + \sin 3) * ...$ (n сомножителей).

Найти сумму ряда: $S = x - x^2/4 + x^3/9 - x^4/16...$ (n слагаемых).

Решение:

Program Pr6;

Var n, i: integer;

p: real;

```

begin
  write('n='); readln(n); p:=1;
  for i:=1 to n do
    p:=p* (i + sin(i+1));

```

```
writeln('p=', p:6:3);
readln
end.
```

Результат:

```
n=5
p=131.645
```

Пример 7.

Задана арифметическая прогрессия $-21; -16; \dots$. Определить номер первого положительного члена прогрессии.

a – очередной член прогрессии, n – его порядковый номер

Решение:

```
Program Pr7;
```

```
var
```

```
  a, n: integer;
```

```
begin
```

```
  a:= -21; n:=1;           {задаем начальные значения}
```

```
  while a<=0 do begin    {пока очередной член прогрессии меньше либо ра
    a:=a+5; n:=n+1;      равен 0, вычисляем следующий член прогрессии и
                        и его порядковый номер}
```

```
end;
```

```
  writeln('n=', n);      {выводим номер 1-го положительного члена
                        прогрессии}
```

```
  readln;
```

```
end.
```

Результат:

```
n=6
```

Пример 8.

Протабулировать функцию $y:=\sin(x)*x$ интервале $[-\pi/2, \pi/2]$ с шагом $\pi/10$.

Решение:

```
Program pr8;
```

```
Var x, y: real;
```

```
begin
```

```
  x:=-pi/2;
```

```
  while x<=pi/2 do
```

```
    begin
```

```
      y:=sin(x)*x;
```

```
      writeln('x=',x:8:2,' y=',y:8:2);
```

```
      x:=x+pi/10
```

```
    end;
```

```
end.
```

Пример 9.

Вычислить сумму:

$$S = \cos(1+x) + \cos^2(1+x) + \cos^3(1+x) + \dots + \cos^n(1+x) \quad (n \text{ слагаемых}).$$

Решение:

```

Program Pr9;
Var n, i: integer;
    s,x,t: real;
begin
  write(' n='); readln(n);
  write(' x='); readln(x);
  s:=0; t:=cos(1+x);
  for i:=1 to n do begin
    s:=s+t;
    t:=t*cos(1+x);
  end;
  writeln('s=', s:4:2)
end.

```

Пример 10.

Подсчитать количество двузначных чисел, у которых сумма цифр нечетна.

Решение:

```

Program Pr10;
Var n, a, b, k: integer;
begin
  k:=0; for n:=10 to 99 do begin
    a:= n div 10; b:=n mod 10;
    if (a+b) mod 2=1 then k:=k+1;
  end;
  writeln('k=',k)
end.

```

ЗАДАНИЯ

1. Дано натуральное число n . Вычислить $2^1 + 2^2 + \dots + 2^n$.
2. Дано натуральное число n . Вычислить $n!$ ($n! = 1 \cdot 2 \cdot 3 \cdot \dots \cdot n$).
3. Дано натуральное число n . Вычислить $\left(1 + \frac{1}{1^2}\right) \left(1 + \frac{1}{2^2}\right) \dots \left(1 + \frac{1}{n^2}\right)$.
4. Дано натуральное число n и действительное число x . Вычислить $\sin x + \sin^2 x + \dots + \sin^n x$.
5. Даны натуральные числа a, n . Вычислить $a(a-n)(a-2n)\dots(a-n \cdot n)$.
6. Найти произведение: $P = (1+x) \cdot (3+2x) \cdot (5+3x) \cdot \dots$ (n множителей)

7. Задана арифметическая прогрессия. 7,6; 6,3; Сколько членов прогрессии нужно сложить, чтобы полученная сумма стала <0 .
8. Задана арифметическая прогрессия 2; 2,8; Сколько членов прогрессии нужно сложить, чтобы полученная сумма стала >20 .
9. Задана арифметическая прогрессия. 7,1; 5,3; Сколько членов прогрессии нужно сложить, чтобы полученная сумма стала <0 .
10. Задана арифметическая прогрессия. 8,4; 6,2 Сколько членов прогрессии нужно сложить, чтобы полученная сумма стала <0 .
11. Задана арифметическая прогрессия. 2; 3,8; Сколько членов прогрессии нужно сложить, чтобы полученная сумма стала >30 .
12. Подсчитать количество и сумму четных трехзначных чисел.
13. Подсчитать сумму двухзначных чисел, сумма цифр которых не превышает 7.
14. Подсчитать количество двузначных чисел, кратных 3.
15. Подсчитать сумму двузначных чисел, сумма цифр которых не превышает 10.
16. Подсчитать количество и сумму трехзначных чисел, кратных 3.
17. Подсчитать количество и сумму четных трехзначных чисел.
18. Протабулировать функцию $y=x^3-1$ на интервале $[-1,3]$ с шагом 0.2.
19. Протабулировать функцию $y=x^2-1/x$ на интервале $[1,2]$ с шагом 0.05.
20. Протабулировать функцию $y= \sin x - \cos x$ на интервале $[-\pi,\pi]$ с шагом $\pi/10$.
21. Протабулировать функцию $y= \cos(x)*x$ на интервале $[-\pi,\pi]$ с шагом $\pi/10$.
22. Протабулировать функцию $y=\sin^2x$ на интервале $[0,\pi]$ с шагом $\pi/20$.
23. Протабулировать функцию $y= \cos^2x+\sin^2x$ на интервале $[-\pi/2,\pi/2]$ с шагом $\pi/10$.
24. Дано натуральное число n. Найти сумму $S=1-3+5-\dots$ (n слагаемых)
25. Дано натуральное число n. Найти сумму $S=1+2/3+3/5 \dots$ (n слагаемых)
26. Дано натуральное число n. Найти сумму $S=1-1/5+1/9-1/13+\dots$ (n слагаемых)
27. Дано натуральное число n. Найти сумму $S=1-3+5-\dots$ (n слагаемых)

IV. ПРОЦЕДУРЫ

Процедура – это относительно самостоятельная часть программы, имеющая собственное имя. Процедура описывается перед основной программой.

Структура процедуры:

Procedure <имя_процедуры>(<параметры>); *-заголовок процедуры*

Label <метки>;	}	<i>раздел объявления меток, констант, типов данных,</i>
Const <константы>;		
Type <типы данных>;		<i>переменных (может отсутствовать)</i>
Var <переменные>;		
Begin		
<операторы >		<i>- тело процедуры</i>
End;		

Параметры, указанные в заголовке процедуры называются *формальными*. Вызов процедуры осуществляется из основной программы указанием ее имени. При вызове процедуры формальные параметры заменяются *фактическими*. Количество и тип формальных и фактических параметров должны совпадать. Для передачи исходных данных в процедуру используются *параметры-значения*. Фактические параметры-значения могут быть константами, переменными, выражениями. Полученный в процедуре результат передается в основную программу с помощью *параметров-переменных*. В заголовке описания процедуры перед параметрами-переменными пишется Var.

Пример 1.

Создать процедуру вычисления среднего арифметического и среднего квадратичного натуральных чисел a и b .

Решение:

```

Program pr1;
Var
  a, b: integer;
  SrAr, SrKv: real; {SrAr- среднее арифметическое, SrKv – среднее квадратичное}
Procedure Sredn(a,b:integer; Var SrAr, SrKv: real);
  {a,b – входные данные (параметры-значения}
  {SrAr, SrKv – результат выполнения процедуры (параметры-переменные)}
Begin
  {тело процедуры}
  SrAr:=(a+b)/2;
  SrKv:=SQRT(a*b);
End;
BEGIN
  {основная программа}
  Write('a='); Readln(a);      {ввод a}
  Write('b='); Readln(b);     {ввод b}
  {вызов процедуры и передача фактических параметров a,b }
  Sredn(a, b, SrAr, SrKv);
  {вывод результата, полученного в процедуре}
  Writeln('Ср. арифм.= ', SrAr:6:2, ' Ср. квадр.е =', SrKv:6:2);
  Readln;
END.

```

Результат:

```

a=2
b=8
Ср. арифм.= 5.00 Ср. квадр.= 4.00

```

Пример 2.

Создать процедуру для вычисления периметра и площади прямоугольного треугольника по его катетам a , b .

Решение:

```

Program pr2;
Var
  a, b, p, s: real;           { a, b- катеты, p - периметр, s - площадь }
Procedure Treug(a,b: real; Var p, s: real);
  {a, b- входные данные(параметры-значения),}
  {p, s-выходные данные(параметры-переменные)}
Begin
  p:= a + b + Sqrt(a*a + b*b); {периметр треугольника с катетами a, b}
  s:= a*b/2;                   { площадь треугольника с катетами a, b }
End;
Begin                          {основная программа}
Write('a, b =');Readln(a, b);   {ввод исходных данных}
  Treug(a, b, p, s);           {вызов процедуры и передача исходных данных a, b}
  Writeln('p=', p:4:1, ' s=', s:4:1); {вывод результата, полученного из процедуры}
  Readln;
End.

```

Результат:

```

a, b= 3 4
p=12.0 s= 6.0

```

Пример 3.

Создать процедуру, определяющую возможность построения треугольника со сторонами x, y, z . Из чисел a, b, c, d выбрать тройки таких, которые позволяют построить треугольник.

Для решения задачи используется «правило треугольника»: каждая сторона треугольника меньше суммы двух других сторон.

Решение:

```

Program pr3;
Var
  a,b,c,d: integer;
Procedure Treug(x, y, z:integer);
Begin
  if (x<y+z) and (y<x+z) and (z<x+y)
    then writeln ('Треугольник со сторонами ' , x, ', ', y, ', ', z, ' можно
    построить')
    else writeln('Треугольник со сторонами ' x, ', ', y, ', ', z, ' нельзя
    построить ')
End;
Begin                          {основная программа}
Write('Введите a b c d: '); Readln(a,b,c,d); {ввод исходных данных}
{вызываем процедуру и проверяем возможность построения треугольника со
сторонами a,b,c}

```

```

    Треуг(a,b,c);
    {вызываем процедуру и проверяем возможность построения треугольника со
    сторонами a,b,d}
    Треуг(a,b,d);
    {вызываем процедуру и проверяем возможность построения треугольника со
    сторонами a,c,d}
    Треуг(a,c,d);
    {вызываем процедуру и проверяем возможность построения треугольника со
    сторонами b,c,d}
    Треуг(b,c,d);
    Readln;
end.

```

Результат:

Введите a b c d: 1 3 4 5

Треугольник со сторонами 1,3,4 нельзя построить

Треугольник со сторонами 1,3,5 нельзя построить

Треугольник со сторонами 1,4,5 нельзя построить

Треугольник со сторонами 3,4,5 можно построить

Пример 4.

Упорядочить по возрастанию числа А, В, С, создав процедуру обмена значениями 2-х переменных.

Решение:

```
Program pr4;
```

```
Var
```

```
  A, B, C: real;
```

```
Procedure Obmen(Var X,Y:real);
```

```
{X,Y являются и входными, и выходными данными}
```

```
Var
```

```
  T: real;          { T –вспомогательная переменная}
```

```
Begin
```

```
  T:=X;
```

```
  X:=Y;
```

```
  Y:=T;
```

```
End;
```

```
begin          {основная программа}
```

```
  Write('A, B, C ='); Readln(A,B,C);          {ввод исходных
  данных}
```

```
  If A>B then Obmen(A,B);          {если A больше B, то производим их обмен}
```

```
  If A>C then Obmen(A,C);          {если A больше C, то производим их обмен}
```

```
  If B>C then Obmen(B,C);          {если B больше C, то производим их обмен}
```

```
  Writeln('A=', A:5:2, ' B=', B:5:2, ' C=', C:5:2);          {вывод результата}
```

```
  Readln;
```

```
end.
```

Результат:

A, B, C = 6 18 3

A= 3.00 B= 6.00 C= 18.00

Пример 5.

Создать процедуру для вывода первых N членов арифметической прогрессии, заданной значением первого члена **a** и разностью **d**. Вывести первые 7 членов прогрессии 2, 5, ... (a=2, d=3) и первые 10 членов прогрессии 20, 19,(a=20, d= -1)

Решение:

Program pr5;

Procedure progres(a,d,n:integer);

Var

i: integer; *{i – номер очередного члена прогрессии }*

Begin

for i:=1 to n do *{перебираем i от 1 до n}*

begin

write(a, ' '); *{выводим очередной член прогрессии}*a:=a+d; *{вычисляем следующий член прогрессии}*

end;

End;

begin *{основная программа}**{вызов процедуры для вывода 7 членов арифметической прогрессии 2, 5,...}*

progres(2,3,7);

writeln;

{вызов процедуры для вывода 10 членов прогрессии 20,19,...}

progres(20,-1,10);

Readln;

end.

Результат:

2 5 8 11 14 17 20

20 19 18 17 16 15 14 13 12 11

Пример 6

Выяснить, какие натуральные числа от 2 до 10 являются простыми, а какие - составными (создать процедуру, определяющую простым или составным является данное число).

Число называется *простым*, если оно не имеет делителей кроме 1 и самого себя. Если у него есть другие делители, то число – *составное*.

Решение:

Program pr6;

```

Var
  i: integer;
Procedure prostoe(x:integer);
Var
  k,n: integer;  {n - возможные делители числа, k – количество делителей}
Begin
  k:=0;
  for n:=2 to x div 2 do  {перебираем числа от 2 до x div 2 для поиска делителей}
    if x mod n = 0      {если n –делитель числа x}
      then k:=k+1;     {увеличиваем количество делителей на 1}
  if k=0                {если у числа нет делителей}
    then writeln (x,' - prostoe')  {то число простое}
    else writeln (x,' - sostavnoe'); {иначе число составное}
End;
begin                  {основная программа}
  for i:=2 to 10 do    {перебираем числа от 2 до 10}
    prostoe(i);        {вызываем процедуру для проверки очередного числа}
  Readln;
end.

```

Результат:

```

2 - prostoe
3 - prostoe
4 - sostavnoe
5 - prostoe
6 - sostavnoe
7 - prostoe
8 - sostavnoe
9 - sostavnoe
10 - sostavnoe

```

ЗАДАНИЯ

1. Создать процедуру для вычисления периметра и площади квадрата по длине его стороны.
2. Создать процедуру для вычисления объема и площади поверхности куба по длине его ребра.
3. Создать процедуру для вычисления длины окружности и площади круга по заданному значению радиуса. ($L = 2\pi R$, $S = \pi R^2$)
4. Создать процедуру для вычисления периметра и площади прямоугольника по длинам его сторон.
5. Создать процедуру для вычисления периметра и площади треугольника по длинам его сторон ($S = \sqrt{p \cdot (p-a) \cdot (p-b) \cdot (p-c)}$, a, b, c – стороны треугольника, p – полупериметр)

6. Создать процедуру для нахождения корней квадратного уравнения по его коэффициентам **a, b, c**.
7. Вводятся A, B, C, D. Поменять местами A и C, B и D, создав процедуру обмена значениями 2-х переменных.
8. Создать процедуру для вывода первых N членов арифметической прогрессии, заданной первым членом **a** и разностью **d**. С помощью этой процедуры вывести 8 первых членов прогрессии 1, 4, ... и 10 первых членов прогрессии 24, 22,
9. Создать процедуру для вывода первых N членов арифметической прогрессии, заданной первым членом **a** и разностью **d**. С помощью этой процедуры вывести 5 первых членов прогрессии 3, 7, ... и 7 первых членов прогрессии 14, 11,
10. Создать процедуру для вывода первых N членов арифметической прогрессии, заданной первым членом **a** и разностью **d**. С помощью этой процедуры вывести 7 первых членов прогрессии 12, 10, ... и 9 первых членов прогрессии 2, 5,
11. Создать процедуру для вывода и нахождения суммы первых N членов арифметической прогрессии, заданной формулой **$a_n = 2n + 1$** .
12. Создать процедуру для вывода и нахождения суммы первых N членов арифметической прогрессии, заданной формулой **$a_n = 3n - 2$** .
13. Создать процедуру для вывода и нахождения суммы первых N членов арифметической прогрессии, заданной формулой **$a_n = 5n - 3$** .
14. Определить, какие из целых чисел от -3 до 8 являются решениями неравенства **$(x-5)(x+1) > 0$** (создать процедуру, определяющую является ли или не является данное целое число решением неравенства **$(x-5)(x+1) > 0$**)
15. Определить, какие из целых чисел от -10 до 2 являются решениями неравенства **$(x+4)(x+1) < 0$** (создать процедуру, определяющую является ли или не является данное целое число решением неравенства **$(x+4)(x+1) < 0$**)
16. Создать процедуру для вывода и подсчета суммы последовательных целых чисел от **a** до **b**. С помощью этой процедуры вывести и найти сумму целых чисел: 1) от 10 до 20; 2) от -5 до 15.
17. Создать процедуру для вывода таблицы значений функции **$y = \sin 3x + 1$** для x от **a** до **b** с шагом **h**. Вывести две таблицы: а) для x от 1 до 2 с шагом 0.1 и б) для x от 5 до 10 с шагом 0.5
18. Создать процедуру для вывода таблицы значений функции **$y = \operatorname{tg} x + 1/x$** для x от **a** до **b** с шагом **h**. Вывести две таблицы: а) для x от 2 до 4 с шагом 0.2 и б) для x от 1 до 10 с шагом 0.5
19. Создать процедуру для вывода таблицы значений функции **$y = 3x^2 + 1$** для x от **a** до **b** с шагом **h**. Вывести две таблицы: а) для x от 1 до 2 с шагом 0.1 и б) для x от 2 до 5 с шагом 0.5
20. Создать процедуру для вывода таблицы значений функции **$y = \sin(x + 1)^2$** для x от **a** до **b** с шагом **h**. Вывести две таблицы: а) для x от 2 до 3 с шагом 0.1 и б) для x от -2 до 3 с шагом 0.5

V. ФУНКЦИИ ПОЛЬЗОВАТЕЛЯ

Структура функции:

```
Function <имя_функции>(<параметры>) : <тип результата>;
  Label <метки>;
  Const <константы>;
  Type <типы данных>;
  Var <переменные>;
Begin
  <операторы, составляющие тело функции>;
End;
```

раздел объявления меток, констант, типов данных, переменных (может отсутствовать)

Для того, чтобы значение функции было определено и передано в основную программу, в теле функции обязательно должен быть хотя бы один оператор присваивания вида:

<имя_функции>:=<значение>.

Вызов функции, определенной пользователем, осуществляется так же, как и любой стандартной функции Паскаля.

Пример 1.

Создать функцию пользователя, вычисляющую $x!$

Решение:

```
Program pr1;
Var
  x: integer; y: real;
Function Fact(x: integer): real;
  Var
    i: integer; f: real;
begin
  f:=1;
  for i:=1 to x do f:=f*i;
  Fact:=f;
end;
begin
  Write ('x='); Readln(x);
  y:=Fact(x);
  Writeln('y=', y:4:0);
  Readln;
end.
```

{вычисляем произведение}
{имени функции присваиваем полученное значение}
{основная программа}
{ввод значения x}
{вызов функции с аргументом x}
{вывод результата}

Результат:

x= 4
y= 24

Пример 2.

Создать функцию пользователя, вычисляющую гиперболический косинус

$$gc = \frac{e^x + e^{-x}}{2}$$

Решение:

```

Program pr2;
Var  x: integer;
     y: real;
Function gc(x:integer):real;
begin
    gc:=(exp(x)+exp(-x))/2;
end;
begin      {основная программа}
    Write ('x='); Readln(x);          {ввод значения x}
    y:=gc(x);                          {вызов функции и присваивание ее значения переменной
y}
    writeln('Гиперболический косинус=', y:8:4);          {вывод результата}
    Readln;
end.

```

Пример 3.

Два треугольника длинами своих сторон. Определить, площадь какого из них больше (создать функцию для вычисления площади треугольника по длинам его сторон).

Для решения задачи используем формулу Герона $S = \sqrt{p \cdot (p-x) \cdot (p-y) \cdot (p-z)}$, где x, y, z – стороны треугольника, p – полупериметр)

Решение:

```

Program pr3;
var
    a1, b1, c1, s1, a2, b2, c2, s2: real;
    {функция вычисления площади треугольника со сторонами x,y,z}
Function PITr(x ,y, z: real): real;
Var
    p: real;
Begin
    p:= (x + y + z)/2;          {полупериметр треугольника}
    PITr:=Sqrt(p*(p-x)*(p-y)*(p-z));
End;
BEGIN      {основная программа}
    {ввод исходных данных}
    Write('Стороны 1-го треугольника:'); Readln(a1, b1, c1);
    Write('Стороны 2-го треугольника:'); Readln(a2, b2, c2);
    S1:=PITr(a1, b1, c1);{вызов функции для нахождения площади 1-го треуголь-
ника}

```

```

S2:=PI*Tr(a2, b2, c2);{вызов функции для нахождения площади 2-го треуголь-
ника}
if S1>S2 then Writeln('S1>S2')
    else if S1<S2 then Writeln('S1<S2')
        else Writeln('S1=S2');

Readln;
END.

```

Результат:

```

Стороны 1-го треугольника:3 4 5
Стороны 1-го треугольника:6 7 8
S1<S2

```

Пример 4.

Создать функцию нахождения максимального из двух чисел

Решение:

```

Program Pr4;
Var
    a,b,m:real;
Function Max(a,b:real):real;
begin
    if a>b                                {если a больше b}
        then Max:=a                       {то функция принимает значение a}
        else Max:=b;                       {иначе функция принимает значение b}
end;
BEGIN
    write('a,b='); readln(a,b);            {ввод исходных данных}
    m:=Max(a,b);                           {вызов функции}
    write('Max=',m:5:1);                    {вывод результата}
    readln;
END.

```

Результат:

```

a,b=14 9
Max= 14.0

```

Пример 5.

Создать функцию нахождения наибольшего общего делителя двух натуральных чисел.

Решение:

```

Program Pr5;
Var
    a,b:integer;

```

```

Function NOD(a,b:integer):integer;
begin
  while a<>b do
    if a>b then a:=a-b {заменяем большее из них разностью большего и меньшего}
    else b:=b-a;
  NOD:=a;
end;
BEGIN
  write('a,b='); readln(a,b);      {ввод исходных данных}
  write('NOD=', NOD(a,b));      {вызов функции и вывод ее значения}
  readln;
END.

```

Результат:

a,b=24 16
NOD=8

Пример 6.

Составить функцию для вычисления x^n , где x, n – целые числа
Степень принимает отрицательное значение, если основание степени $X < 0$ и показатель степени – нечетное число.

Решение:

```

Program Pr6;
Var
  x,n:integer;
Function stepen(x,n:integer):real;
begin
  if (n mod 2 <> 0) and (x<0) then stepen:=-exp(n*ln(abs(x)))
  else stepen:=exp(n*ln(abs(x)));
end;
BEGIN
  write('x,n='); readln(x,n);      {ввод исходных данных}
  write('stepen=',stepen(x,n):6:2); {вызов функции и вывод ее значения}
  readln;
END.

```

Результат:

1 случай
x,n=2 3
stepen= 8.00
2 случай
x,n=-2 3
stepen= -8.00
3 случай

x,n=2 -2
 stepen= 0.25
 4 случай
 x,n=-2 -2
 stepen= 0.25

Пример 7.

Вычислить, используя функцию нахождения знаменателя:

$$\frac{1}{2!} + \frac{2}{3!} + \frac{3}{4!} + \dots \text{(n слагаемых)}$$

Решение:

```

Program pr7;
Var s: real;
    i, n: integer;
Function Fakt(x: integer): real;      {функция вычисления факториала}
Var
    i: integer; f: real;
begin
    f:=1;
    for i:=1 to x do f:=f*i;
    Fakt:=f;
end;
BEGIN
    write('n= '); Readln(n);      {вводим количество слагаемых суммы}
    s:=0;
    For i:=1 to n do              {перебираем i от 1 до n}
        s:=s+i/Fakt(i+1);        {добавляем к сумме очередную дробь, вызывая}
                                { функцию для вычисления знаменателя}
    writeln('s=',s:6:4);          {вывод результата}
    Readln;
END.
  
```

Результат:

n=3
 s=0.9583

Пример 8.

Вычислить, используя функцию нахождения знаменателя:

$$\frac{5}{1+2} + \frac{10}{1+2+3} + \frac{15}{1+2+3+4} + \dots \quad \text{(n слагаемых)}$$

Решение:

```

Program pr8;
Var s: real;
    i, n: integer;
  
```

{ функция вычисления суммы первых m натуральных чисел }

Function Sum(m:integer):integer;

Var

k, S: integer;

begin

S:=0;

for k:=1 to m do

S:=S+k;

Sum:=S;

end;

BEGIN

write('n= '); Readln(n); *{ вводим количество слагаемых суммы }*

s:=0;

for i:=1 to n do *{ перебираем i от 1 до n }*

s:=s+i*5/sum(i+1); *{ добавляем к сумме очередную дробь, вызывая }*

{ функцию для вычисления знаменателя }

writeln('s=',s:6:4)

{ вывод результата }

END.

Результат:

n=3

s=4.8333

ЗАДАНИЯ

1. Найти минимальное из A, B, C, создав функцию выбора минимального из двух произвольных чисел.
2. Найти максимальное из чисел A, B, C, D, создав процедуру выбора максимального из 2-х произвольных чисел.
3. Сократить дробь вида a/b (a, b – вводимые натуральные числа), создав функцию для нахождения наибольшего общего делителя двух натуральных чисел.
4. Найти наибольший общий делитель чисел A, B, C, создав функцию для нахождения НОД двух натуральных чисел.
5. Два прямоугольных треугольника заданы своими катетами. Определить, у какого из них площадь больше (создать функцию для вычисления площади прямоугольного треугольника по его катетам).
6. Два прямоугольника заданы координатами вершин. Определить, площадь какого из них больше (использовать функцию нахождения площади прямоугольника).
7. Два прямоугольных треугольника заданы своими катетами. Определить, у какого из них периметр больше (создать функцию для вычисления периметра прямоугольного треугольника по его катетам).

8. Вводятся 2 натуральных числа. Найти среднее арифметическое цифр каждого из них (создать функцию для нахождения среднего арифметического цифр произвольного натурального числа).
9. Вводятся 3 натуральных числа. Найти сумму цифр каждого из них (создать функцию для нахождения суммы цифр произвольного натурального числа).
10. Вычислить $3^{-2} + 4^2 + 5^{-2}$, создав функцию для вычисления степени.
11. Вычислить $(-3^2 + 2^4)/4^{-2}$, создав функцию для вычисления степени.
12. Найти $(a! + b!)/(a+b)!$, создав функцию для вычисления факториала произвольного натурального числа.
13. Найти $m! + (m+n)!$, создав функцию для вычисления факториала произвольного натурального числа.
14. Вычислить, используя функцию нахождения знаменателя:

$$\frac{1}{1+2} + \frac{3}{1+2+3} + \frac{5}{1+2+3+4} + \dots \quad (15 \text{ слагаемых})$$
15. Вычислить, используя функцию нахождения знаменателя:

$$\frac{1}{2!} + \frac{4}{3!} + \frac{9}{4!} + \dots \quad (n \text{ слагаемых})$$
16. Вычислить, используя функцию нахождения знаменателя:

$$\frac{2}{3!} + \frac{4}{5!} + \frac{6}{7!} + \dots \quad (n \text{ слаг.})$$
17. Вычислить, используя функцию нахождения знаменателя:

$$\frac{2}{1+2} + \frac{4}{1+2+3} + \frac{6}{1+2+3+4} + \dots \quad (15 \text{ слаг})$$
18. Вычислить, используя функцию нахождения знаменателя:

$$\frac{1}{3!} + \frac{2}{4!} + \frac{3}{5!} + \dots \quad (n \text{ слагаемых})$$
19. Вычислить, используя функцию нахождения знаменателя:

$$\frac{2}{3!} + \frac{4}{5!} + \frac{6}{7!} + \dots \quad (n \text{ слаг.})$$
20. Вычислить, используя функцию нахождения знаменателя:

$$\frac{1}{1+3} + \frac{2}{1+3+5} + \frac{3}{1+3+5+7} + \dots \quad (15 \text{ слаг})$$

ЛИТЕРАТУРА

1. Фаронов В.В. Турбо Паскаль 7.0. Начальный курс. Учебное пособие.-М.: «Нолидж», 1999-616с., ил.
2. Д. Праймс. Программирование на языке Паскаль. Практическое руководство М.: Мир, 1997.
3. Васюкова Н.Д., Тюляева В.Р. Практикум по основам программирования языка Паскаль. Учебное пособие. М.: Высшая школа. 1991.
4. Фаронов В.В. Турбо Паскаль 7.0. Практика программирования. Учебное пособие.-М.: «Нолидж», 1999-432с., ил.
5. Д. Праймс. Программирование на языке Паскаль. Практическое руководство М.: Мир, 1997.
6. Вирт Н. Алгоритмы и структуры данных. – М.,1989.