

### ЛАБОРАТОРНАЯ РАБОТА № 3. ПРОГРАММИРОВАНИЕ АЛГОРИТМОВ ВЕТВЛЕНИЯ

*Цель работы:* освоение навыков программирования алгоритмов ветвления при решении задач.

*Задания по теме «Программирование алгоритмов ветвления»:*

1. Дано  $N$  троек действительных положительных чисел. Проверить, сколько из них могут являться длинами сторон треугольника.

Решение на Бейсике:

```
10 INPUT N 'Количество троек
20 K=0 'Первоначально количество треугольников равно
21     '0
30 FOR I=1 TO N
40 INPUT A,B,C
45 'Если одна сторона меньше суммы двух других, то
46 'K увеличиваем на 1
50 IF A<(B+C) AND B<(A+C) AND C<(A+B) THEN K=K+1
60 NEXT I
70 PRINT K
```

Решение на Паскале:

```
var
    i,                {вспомогательная переменная}
    n,                {количество троек}
    k : byte;         {количество треугольников}
    a,b,c : real;     {длины сторон треугольника}
begin
    readln(n);
    for i:=1 to n do
        begin
            readln(a,b,c);
            {Если одна сторона меньше суммы двух других, то K увеличиваем
            на 1}
            if (a<(b+c)) and (b<(a+c)) and (c<(a+b)) then
                k:=k+1;
            end;
        writeln(k)
    end.
```

2. Даны действительные положительные числа X, Y и Z. Выяснить, существует ли треугольник с длинами сторон X, Y и Z. Если он существует, то ответить, является ли он остроугольным.

Решение на Бейсике:

```

10 INPUT X,Y,Z
12 'Если сумма двух любых сторон не больше третьей,
13 'то, треугольника не существует
15 IF (X+Y)<=Z OR (X+Z)<=Y OR (Y+Z)<=X THEN
    PRINT "Треугольника не существует" : STOP
17 'Если сумма квадратов двух любых сторон больше
18 'квадрата третьей, то, треугольник является остро-
19 'угольным
20 SQRX=X^2
30 SQRY=Y^2
40 SQRZ=Z^2
50 IF (SQRY+SQRZ)>SQRX AND (SQRX+SQRZ)>SQRY AND
    (SQRY+SQRX)>SQRZ then PRINT "Треугольник остро-
    угольный" ELSE PRINT "Треугольник не остроуголь-
    ный"
```

Решение на Паскале:

```

var
  x,y,z           :real;      {длины сторон}
  sqr_X, sqr_Y, sqr_Z:real;    {их квадраты}
  b_X,b_Y,b_Z     :Boolean;  {булевские переменные
                               для упрощения логического выражения}

begin
  Readln(x,y,z);
  {Если сумма двух любых сторон не больше третьей,
   то, треугольника не существует}
  if ((x+y)<=z) or ((x+z)<=y) or ((y+z)<=x) then
    writeln('Треугольника не существует')
  else
    begin
      {Если сумма квадратов двух любых сторон больше
       квадрата третьей, то треугольник является
       остроугольным}
      sqr_X:=sqr(x);
      sqr_Y:=sqr(y);
      sqr_Z:=sqr(z);
      b_X:=( sqr_Y + sqr_Z ) > sqr_X;
      b_Y:=( sqr_X + sqr_Z ) > sqr_Y;
```

```

b_Z:=( sqr_Y + sqr_X ) > sqr_Z;
Write('Треугольник ');
if b_X and b_Y and b_Z then
    Writeln('остроугольный')
else
    Writeln('не остроугольный')
end
end.

```

3. Даны действительные положительные числа A,B,C,D. Выяснить, можно ли прямоугольник со сторонами A,B уместить внутри прямоугольника со сторонами C,D так, чтобы каждая из сторон одного прямоугольника была параллельна или перпендикулярна каждой стороне второго прямоугольника.

Решение на Бейсике:

```

10 PRINT "Введи стороны 1-го прямоуг. ";
20 INPUT A,B
30 PRINT " Введи стороны 2-го прямоуг. ";
40 INPUT C,D
45 'Меньшие стороны прямоугольников помещаем в пе-
46 'ременные Min1, Min2, а большие - в Max1, Max2
50 IF A>B THEN Max1=A : Min1=B ELSE Max1=B : Min1=A
60 IF C>D THEN Max2=C : Min2=D ELSE Max2=D : Min2=C
70 IF (Min1<Min2) AND (Max1<Max2) THEN
    PRINT "1-ый прям-к можно уместить внутри 2-го"
    ELSE PRINT "1-ый прям-к нельзя уместить внутри
        2-го"

```

Решение на Паскале:

```

var
    a,b,c,d:integer;
    min1,max1,min2,max2:integer;
begin
    Writeln('Введи стороны 1-го прямоуг. ');
    Readln(a,b);
    Writeln('Введи стороны 2-го прямоуг. ');
    Readln(c,d);
    {Меньшие стороны прямоугольников помещаем в пере-
    менные min1, min2, а большие - в max1, max2}
    if a>b then
        begin
            max1:=a;
            min1:=b

```

```

    end
else
    begin
        max1:=b;
        min1:=a
    end;
if c>d then
    begin
        max2:=c;
        min2:=d
    end
else
    begin
        max2:=d;
        min2:=c
    end;
if (min1 < min2) and (max1 < max2) then
    Writeln('1-ый пр-к можно уместить внутри 2-го')
else
    Writeln('1-ый пр-к нельзя уместить внутри 2-го')
end.

```

4. Составить программу, которая вычисляет площадь треугольника, если человек задал длины трех сторон А,В,С. При вычислении использовать формулу Герона:

$$S = \sqrt{(P(A+B-C))(P(A-B+C))(P(-A+B+C))}$$
 , где  $P$  — полупериметр. Перед вычислением площади проверить, существует ли треугольник с заданными сторонами.

Решение на Бейсике:

```

10 INPUT "ВВЕДИТЕ ДЛИНЫ ТРЕХ СТОРОН";A,B,C
15 'если длина одной из сторон больше или равна
16 'сумме двух других, то треугольник не существует
20 IF A>=(B+C) OR B>=(A+C) OR C>=(A+B) THEN
    PRINT "ТРЕУГОЛЬНИК ПОСТРОИТЬ НЕЛЬЗЯ" : STOP
30 P=(A+B+C)/2
40 S=SQR(P*(P-A)*(P-B)*(P-C))
50 PRINT "ПЛОЩАДЬ РАВНА ";S

```

Решение на Паскале:

```

var
    a,b,c,d,p,s : real;
begin

```

```

readln(a,b,c);
write('введите длины трех сторон ');
{Если длина любой из сторон треугольника больше
или равна сумме двух других, то треугольник пост-
роить нельзя}
if (a>=(b+c)) or (b>=(a+c)) or (c>=(a+b)) then
    writeln('треугольник построить нельзя')
else
    begin
        p:=(a+b+c)/2;
        s:=sqrt(p*(p-a)*(p-b)*(p-c));
        writeln('площадь = ',s:6:2)
    end;
end.

```

5. Задано N треугольников координатами их вершин. Определить, сколько из них остроугольных, прямоугольных и тупоугольных.

Решение на Бейсике:

```

10 'Функция расчета длины стороны треугольника
20 DEF FNM(X1,X2,Y1,Y2)=SQR((X1-X2)^2+(Y1-Y2)^2)
30 INPUT N : 'Число треугольников
31 DIM X1(N),Y1(N),X2(N),Y2(N),X3(N),Y3(N) 'Массив
32 'координат
33 'Начальное кол-во соответствующих треугольников
40 K1=0 : K2=0 : K3=0
41 'Циклическая часть программы выполняется N раз
50 FOR I=1 TO N
60 INPUT "КООРДИНАТЫ ПЕРВОЙ ВЕРШИНЫ";X1(I),Y1(I)
70 INPUT "КООРДИНАТЫ ВТОРОЙ ВЕРШИНЫ";X2(I),Y2(I)
80 INPUT "КООРДИНАТЫ ТРЕТЬЕЙ ВЕРШИНЫ";X3(I),Y3(I)
81 'Нахождение длин сторон
90 A=FNM(X1(I),X2(I),Y1(I),Y2(I))
100 B=FNM(X1(I),X3(I),Y1(I),Y3(I))
110 C=FNM(X2(I),X3(I),Y2(I),Y3(I))
111 'Если сумма квадратов длин двух сторон больше
112 'квадрата третьей, то к переменной k1 добавить
114 '1,если сумма квадратов длин любых сторон равна
115 'квадрату третьей, к переменной k2 добавить 1,
116 'иначе k3 увеличить на 1
120 IF (A^2+B^2)>C^2 AND (A^2+C^2)>B^2 AND
    (B^2+C^2)>A^2 THEN K1=K1+1 ELSE
    IF (A^2+B^2)=C^2 OR (B^2+C^2)=A^2 OR
    (A^2+C^2)=B^2 THEN K2=K2+1 ELSE K3=K3+1

```

```

130 NEXT I
140 PRINT "ОСТРОУГОЛЬНЫХ"; K1, "ПРЯМОУГОЛЬНЫХ";K2,
    "ТУПОУГОЛЬНЫХ"; K3

```

Решение на Паскале:

```

var
    a,b,c : real;           {Длины сторон треугольников}
    x1,x2,x3,y1,y2,y3 : real; {Координаты вершин}
    n,i,k1,k2,k3 : byte;
    {Функция вычисления длины стороны}
function ds(a1,b1,a2,b2 : real) : real;
begin
    ds:=sqrt (sqr(a1-a2)+sqr(b1-b2))
end;
    {Основная программа}
begin
    readln(n);               {Ввести число треугольников}
    {Начальное кол-во соответствующих треугольников}
    k1:=0; k2:=0; k3:=0;
    for i:=1 to n do {Организация цикла для обработки
                        данных}
        begin
            write('Координаты первой вершины ');
            readln(x1,y1);
            write('Координаты второй вершины ');
            readln(x2,y2);
            write('Координаты третьей вершины ');
            readln(x3,y3);
            {Вычисление длины сторон}
            a:=ds(x1,y1,x2,y2);
            b:=ds(x1,y1,x3,y3);
            c:=ds(x3,y3,x2,y2);
            {Если сумма квадратов длин двух сторон равна
квадрату третьей, то к переменной k1 добавить 1}
            if ((a*a+b*b)>c*c) and ((a*a+c*c)>b*b) and
                ((c*c+b*b)>a*a)
            then
                k1:=k1+1
            else
                {Если сумма квадратов длин двух любых сторон равна
квадрату третьей, то к переменной k2 добавить 1, иначе k3
увеличить на 1}
                if ((a*a+b*b)=c*c) or ((a*a+c*c)=b*b) or

```

```

        ((c*c+b*b)=a*a)
    then k2:=k2+1 else k3:=k3+1
end;
writeln('осторог.' , k1, 'прямоуг. ', k2, ' тупоуг.', k3)
end.

```

6. Задано N треугольников координатами их вершин. Определить, сколько в этом списке тупоугольных треугольников с периметром больше заданного числа P.

Решение на Бейсике:

```

10 'Функция вычисления длин сторон треугольника
20 DEF FNM(X1,X2,Y1,Y2)=SQR((X1-X2)^2+(Y1-Y2)^2)
30 INPUT N : 'Ввод числа треугольников
31 DIM X1(N),Y1(N),X2(N),Y2(N),X3(N),Y3(N) 'Массив
32 'координат вершин
40 INPUT P 'Ввести периметр
50 K=0 'Начальное число треугольников
60 FOR I=1 TO N 'Организация цикла на N повторов
70 INPUT "КООРДИНАТЫ ПЕРВОЙ ВЕРШИНЫ";X1(I),Y1(I)
80 INPUT "КООРДИНАТЫ ВТОРОЙ ВЕРШИНЫ";X2(I),Y2(I)
90 INPUT "КООРДИНАТЫ ТРЕТЕЙ ВЕРШИНЫ";X3(I),Y3(I)
91 'Вызов функции вычисления длин сторон тр-ка
100 A=FNM(X1(I),X2(I),Y1(I),Y2(I))
110 B=FNM(X1(I),X3(I),Y1(I),Y3(I))
120 C=FNM(X2(I),X3(I),Y2(I),Y3(I))
121 'Если периметр треугольника больше заданного P
122 'и сумма квадратов двух любых сторон меньше
123 'квадрата третьей, то k увеличить на 1
130 IF A+B+C>P AND ((A^2+B^2)<C^2 OR (B^2+C^2)<A^2
    OR (A^2+C^2)<B^2) THEN K=K+1
140 NEXT I 'Конец цикла
150 PRINT K 'Вывод числа треугольников

```

Решение на Паскале:

```

var
  a,b,c,p,per : real;
  x1,x2,x3,y1,y2,y3 : real; {Координаты вершин}
  i,k,n : byte;
  {Функция вычисления длин сторон треугольника}
function ds(a1,b1,a2,b2 : real) : real;
begin
  ds:=sqrt(sqr(a1-a2)+sqr(b1-b2))

```

```

end;
{Основная программа}
begin
  readln(n); {Ввести число треугольников}
  readln(p); {Ввести периметр}
  k:=0; {Начальное количество треугольников}
  for i:=1 to n do {Организуем цикл на N повторов}
    begin
      write('Координаты первой вершины ');
      readln(x1,y1);
      write('Координаты второй вершины ');
      readln(x2,y2);
      write('Координаты третьей вершины ');
      readln(x3,y3);
      {Расчет длин и периметра треугольника}
      a:=ds(x1,y1,x2,y2);
      b:=ds(x1,y1,x3,y3);
      c:=ds(x3,y3,x2,y2);
      per:=a+b+c;
      {Если сумма квадратов любых сторон больше квадрата
      третьей, а периметр больше заданного числа p, то k
      увеличить на 1}
      if (((a*a+b*b)>c*c) or ((a*a+c*c)>b*b) or
          ((c*c+b*b)>a*a)) and (per>p) then k:=k+1
    end;
    writeLn(k) {Вывод числа треугольников}
  end.

```

7. Задано N точек на плоскости, не находящихся на осях координат. Для тех из них, которые лежат в круге с центром в начале координат и радиусом R, определить, сколько принадлежит каждой из четвертей координатной плоскости.

Решение на Бейсике:

```

20 INPUT N : 'Ввести число точек
21 DIM X(N),Y(N) 'Массив координат точки
30 INPUT R 'Ввести радиус окружности
31 'Начальное количество точек в каждой четверти
40 K1=0 : K2=0 : K3=0 : K4=0
50 FOR I=1 TO N 'Организация цикла на N повторов
60 PRINT "КООРДИНАТЫ";I;"-ОЙ "; "ТОЧКИ";
70 INPUT X(I),Y(I) 'Ввести координаты точки
71 'Если точка вне круга, переходим на метку 100,
73 'иначе, определяем в какой четверти точка

```



```

77  'находится
80  IF X(I)^2+Y(I)^2>R^2 GOTO 100
90  IF X(I)>0 AND Y(I)>0 THEN K1=K1+1 ELSE IF X(I)>0
    AND Y(I)<0 THEN K2=K2+1 ELSE
    IF X(I)<0 AND Y(I)<0 THEN K3=K3+1 ELSE K4=K4+1
100 NEXT I          'Конец цикла
110 PRINT "1-я ЧЕТВ."; K1, "2-я ЧЕТВ."; K2, "3-я
    ЧЕТВ."; K3, "4-я ЧЕТВ."; K4

```

Решение на Паскале:

```

var
  x,y,r : real;
  n,i,k1,k2,k3,k4 : byte;
begin
  readln(n);          {Ввести число точек}
  readln(r);          {Ввести радиус окружности}
  {Начальное количество точек в каждой четверти}
  k1:=0; k2:=0; k3:=0; k4:=0;
  for i:=1 to n do {Организация цикла на n
                    повторов}
    begin
      write('Координаты ',i,'-ой точки ');
      readln(x,y);    {Ввести координаты точки}
      {Если сумма квадратов координат точки не больше
      квадрата радиуса круга - точка находится внутри круга}
      if (x*x+y*y)<=r*r then
        {Оценка координат точек, расчет количества точек в
        каждой четверти}
        if (x>0) and (y>0) then k1:=k1+1 else
          if (x>0) and (y<0) then k2:=k2+1 else
            if (x<0) and (y<0) then k3:=k3+1 else
              k4:=k4+1;
        end;          {Конец цикла}
        {Вывод количества точек}
      writeln('1-я ч. ', k1, ' 2-я ч. ', k2,
        ' 3-я ч. ', k3, ' 4-я ч. ', k4);
    end.

```

8. Задано  $N$  точек на плоскости и два прямоугольника со сторонами, параллельными осям координат. Определить, сколько точек принадлежит только первому прямоугольнику, сколько только второму и сколько обоим вместе.

Решение на Бейсике:

```

20 PRINT "ПЕРВЫЙ ПРЯМОУГОЛЬНИК"
30 INPUT "КООРДИНАТЫ ЛЕВОГО НИЖНЕГО УГЛА";P1X1,P1Y1
40 INPUT "КООРДИНАТЫ ПРАВОГО ВЕРХНЕГО УГЛА";P1X2,P1Y2
50 PRINT "ВТОРОЙ ПРЯМОУГОЛЬНИК"
60 INPUT "КООРДИНАТЫ ЛЕВОГО НИЖНЕГО УГЛА";P2X1,P2Y1
70 INPUT "КООРДИНАТЫ ПРАВОГО ВЕРХНЕГО УГЛА";P2X2,P2Y2
80 INPUT N           'Ввести число точек
81 DIM X(N),Y(N)
90 K1=0 : K2=0 : K3=0
100 FOR I=1 TO N      'Цикл на n повторов
110 PRINT "КООРДИНАТЫ";I;"-й "; "ТОЧКИ";
120 INPUT X(I),Y(I)
121 'Если точка принадлежит обоим прямоугольникам,
122 'к k3 добавить 1
130 IF X(I)>=P1X1 AND X(I)<=P1X2 AND X(I)>=P2X1 AND
    X(I)<=P2X2 AND Y(I)>=P1Y1 AND Y(I)<=P1Y2 AND
    Y(I)>=P2Y1 AND Y(I)<=P2Y2 THEN K3=K3+1 :
    GOTO 140 'Если выражение верно,
131 'переходим в конец цикла
132 'Если точка принадлежит первому прямоугольнику,
133 'к k1 добавить 1, иначе k2 увеличить на 1
135 IF X(I)>=P1X1 AND X(I)<=P1X2 AND Y(I)>=P1Y1 AND
    Y(I)<=P1Y2 THEN K1=K1+1 ELSE
    IF X(I)>=P2X1 AND X(I)<=P2X2 AND Y(I)>=P2Y1
    AND Y(I)<=P2Y2 THEN K2=K2+1
140 NEXT I           'Конец цикла
150 PRINT "1-й ПРЯМ.";K1,"2-й ПРЯМ.";K2,"ОБА";K3

```

Решение на Паскале:

```

var
  x,y : real;           {Координаты точки}
  n,i,k1,k2,k3 : byte;
  p1x1,p1y1,p1x2,p1y2,p2x1,p2y1,p2x2,p2y2 : real; {Координаты
  вершин}
  pr1,pr2 : boolean;    {Логические переменные}
begin
  readln(n);             {Ввести число точек}
  writeln('1-й прямоугольник');
  write('Координаты левого нижнего угла ');
  readln(p1x1,p1y1);
  write('Координаты правого верхнего угла ');
  readln(p1x2,p1y2);
  writeln('2-й прямоугольник');

```

```

write('Координаты левого нижнего угла ');
readln(p2x1,p2y1);
write('Координаты правого верхнего угла ');
readln(p2x2,p2y2);
{Начальное число точек в каждом и в обоих прямо-
угольниках}
k1:=0; k2:=0; k3:=0;
for i:=1 to n do           {Цикл на n повторов}
  begin
    write('Координаты ',i,'-ой точки ');
    readln(x,y);
    {Используя логические выражения, определяем местоположение
    точки}
    pr1:=(x>p1x1) and (x<p1x2) and (y>p1y1) and
      (y<p1y2);
    pr2:=(x>p2x1) and (x<p2x2) and (y>p2y1) and
      (y<p2y2);
    {Если оба логических выражения верны, к k3 доба-
    вить 1, если истинно первое выражение (точка на-
    ходится в первом прямоугольнике), k1 увеличить на
    1, иначе k2 увеличить на 1}
    if pr1 and pr2 then k3:=k3+1 else
      if pr1 then k1:=k1+1 else
        if pr2 then k2:=k2+1
    end;
    {Конец цикла}
    writeln('1-й пр. ',k1,' 2-й пр. ',k2,' оба ',k3);
  end.

```

9. Заданы своими координатами N точек на плоскости и два круга радиусами R1 и R2 ( $R1 < R2$ ) с центром в начале координат. Определить, сколько точек находится одновременно в обоих кругах и сколько вне их.

Решение на Бейсике:

```

10 INPUT "Введите количество точек";N
20 INPUT "Введите радиусы";R1,R2
30 K1=0 'Начальное количество точек внутри кругов
40 K2=0 'Начальное количество точек вне
50 FOR I=1 TO N 'Цикл на N повторов
60 PRINT "Введите координаты";I;"-ой точки"
70 INPUT X,Y
71 'Определяем положение точки
80 P1=X^2+Y^2<=R1^2
90 P2=X^2+Y^2<=R2^2
91 'Если истинно первое выражение, k1 увеличить на

```

```

92 '1, иначе увеличиваем k2
100 IF P1 THEN K1=K1+1 ELSE IF NOT P2 THEN K2=K2+1
110 NEXT I           'Конец цикла
120 PRINT "Принадлежит обоим";K1
130 PRINT "Вне кругов";K2

```

Решение на Паскале:

```

Var
  R1,R2 : real;           {Радиус окружностей}
  N,i,k1,k2 : word;
  p1,p2 : boolean;       {Переменные логического типа}
  x,y : real;             {Координаты точек}
Begin
  Write('Введите количество точек ');
  readln(N);
  Write('Введите радиусы ');
  readln(R1,R2);
  k1:=0; {Начальное количество точек внутри кругов}
  k2:=0; {Начальное количество точек вне}
  for i:=1 to N do {Цикл на N повторов}
    begin
      write('Введите координаты ',i,'-ой точки ');
      readln(x,y);
      p1:=(x*x+y*y)<=R1*R1;
      p2:=(x*x+y*y)<=R2*R2;
      {Если первое логическое выражение истинно, k1
       увеличь на 1, иначе увеличиваем k2}
      if p1 then inc(k1) {Процедура увеличения
                        аргумента на 1}
      else if not p2 then inc(k2)
      end;
      {Конец цикла}
      writeln('Принадлежит обоим ',k1);
      writeln('Вне кругов ',k2);
    end.

```

10. Заданы своими координатами N точек на плоскости, круг радиусом R с центром в начале координат и вписанный в него квадрат со сторонами, параллельными осям координат. Определить, сколько точек находится одновременно и в круге, и в квадрате, а сколько вне их.

Решение на Бейсике:

```

10 INPUT "Введите количество точек";N
20 INPUT "Введите радиус";R
30 K1=0 'Начальное количество точек и в круге, и в

```

```

31 'квдрате
40 K2=0 'Начальное количество точек вне круга
50 FOR I=1 TO N 'Цикл на N повторов
60 PRINT "Введите координаты";I;"-ой точки"
70 INPUT X,Y
80 P1=X^2+Y^2<=R^2
90 A=R*SQR(2)/2 'Найти половину стороны квадрата
100 P2 = X>=-A AND X<=A AND Y>=-A AND Y<=A
101 'Если истинно второе логическое выражение k1
102 'увеличить на 1, иначе k2 увеличить на 1
110 IF P2 THEN K1=K1+1 ELSE IF NOT P1 THEN K2=K2+1
120 NEXT I 'Конец цикла
130 PRINT "Принадлежит обоим";K1
140 PRINT "Вне";K2

```

Решение на Паскале:

```

Var
  R,a : real;
  N,i,k1,k2 : word;
  p1,p2 : boolean; {Переменные логического типа}
  x,y : real; {Координаты точки}
Begin
  Write('Введите количество точек ');
  readln(N);
  Write('Введите радиус ');
  readln(R);
  k1:=0; {Начальное количество точек и в круге, и в
         квадрате}
  k2:=0; {Начальное количество точек вне круга}
  for i:=1 to N do {Цикл на N повторов}
    begin
      write('Введите координаты ',i,'-ой точки ');
      readln(x,y);
      {Первой логической переменной присваивается выражение}
      p1:=(x*x+y*y)<=(R*R);
      a:=r*sqr(2)/2; {Находим половину стороны
                     квадрата}
      p2:=(x>=-a) and (x<=a) and (y>=-a) and
          (y<=a);
      {Если истинно второе логическое выражение, k1
       увеличить на 1, иначе k2}
      if p2 then inc(k1)
      else if not p1 then inc(k2)
    end
  end

```

```

        end;                                {Конец цикла}
    {Вывод результатов}
    writeln('Принадлежит обоим ',k1);
    writeln('Вне ',k2);
end.

```

11. Заданы своими координатами N точек на плоскости, круг радиусом R с центром в начале координат и описанный вокруг него квадрат со сторонами, параллельными осям координат. Определить, сколько точек находится одновременно и в круге, и в квадрате, а сколько вне их.

Решение на Бейсике:

```

10 INPUT "Введите количество точек";N
20 INPUT "Введите радиус";R
30 K1=0'Начальное кол-во точек в круге и в квадрате
40 K2=0 'Начальное количество точек вне квадрата
50 FOR I=1 TO N                'Цикл на N повторов
60 PRINT "Введите координаты";I;"-ой точки"
70 INPUT X,Y
80 P1=X^2+Y^2<=R^2
90 P2 = X>=-R AND X<=R AND Y>=-R AND Y<=R
91 'Если истинно первое логическое выражение, k1
92 'увеличить на 1, иначе увеличиваем k2
100 IF P1 THEN K1=K1+1 ELSE IF NOT P2 THEN K2=K2+1
110 NEXT I                    'Конец цикла
120 PRINT "Принадлежит обоим";K1
130 PRINT "Вне";K2

```

Решение на Паскале:

```

Var
    R : real;                                {Радиус круга}
    N,i,k1,k2 : word;
    p1,p2 : boolean;                        {Логические переменные}
    x,y : real;                            {Координаты точки}
Begin
    Write('Введите количество точек ');
    readln(N);
    Write('Введите радиус ');
    readln(R);
    k1:=0; {Нач. кол-во точек в круге и в квадрате}
    k2:=0; {Начальное количество точек вне квадрата}
    for i:=1 to N do                    {Цикл на N повторов}
        begin
            write('Введите координаты ',i,'-ой точки ');

```

```

    readln(x, y);
    p1:=(x*x+y*y)<=(R*R);
    p2:=(x>=-R) and (x<=R) and (y>=-R) and (y<=R);
    {Если истинно первое логическое выражение, k1
      увеличить на 1, используя процедуру увеличе-
      ния аргумента, иначе k2 увеличить на 1}
    if p1 then inc(k1)
    else if not p2 then inc(k2)
  end;
  {Конец цикла}
writeln('Принадлежит обоим ',k1);
writeln('Вне ',k2);
end.

```

12. Заданы координатами своих концов  $N$  отрезков на плоскости. Опреде-  
 лить, сколько отрезков имеют длину больше  $L1$  и меньше  $L2$  ( $L1 < L2$ ).

Решение на Бейсике:

```

10 INPUT "Введите количество отрезков";N
20 INPUT "Введите L1 и L2";L1,L2
30 K=0      'Начальное количество отрезков с длиной
31 'больше L1 и меньше L2
40 FOR I=1 TO N
50 PRINT "Введите координаты концов"; I;
   " -ого отрезка"
60 INPUT X1,Y1,X2,Y2
70 L=SQR((X1-X2)^2+(Y1-Y2)^2) 'Длина I-ого отрезка
71 'Если длина I-ого отрезка больше L1 и меньше L2,
72 'k1 увеличить на 1
80 IF L>L1 AND L<L2 THEN K=K+1
90 NEXT I      'конец цикла
100 PRINT K

```

Решение на Паскале:

```

Var
  L1,L2,l : real;
  N,i,k : word;
  x1,y1,x2,y2 : real;
Begin
  Write('Введите количество отрезков ');
  readln(N);
  Write('Введите L1 и L2 ');
  readln(L1,L2);
  k:=0; {Начальное количество отрезков с длиной
        больше L1 и меньше L2}

```

```

for i:=1 to N do           {Цикл на N повторов}
begin
  write('Введите координаты концов ',i,'-го
    отрезка ');
  readln(x1,y1,x2,y2);
  l:=sqrt(sqr(x1-x2)+sqr(y1-y2));
  {Если длина I-ого отрезка больше L1 и меньше L2, k1
  увеличить на 1}
  if (l>L1) and (l<L2) then inc(k);
end;                       {Конец цикла}
writeln(k);
end.

```

13. Заданы координатами своих концов  $N$  отрезков на плоскости. Определить, сколько отрезков имеют хотя бы одну общую точку с осью  $X$ , а и сколько хотя бы одну общую точку с осью  $Y$ .

Решение на Бейсике:

```

10 INPUT "Введите количество отрезков";N
20 K1=0 'Начальное количество отрезков, имеющих
21 'общую точку с осью X
30 K2=0 'Начальное количество отрезков имеющих
31 'общую точку с осью Y
40 FOR I=1 TO N           'Цикл на N повторов
50 PRINT "Введите координаты концов"; I;
   "о-ого отрезка"
60 INPUT X1,Y1,X2,Y2
61 'Если произведение координат X обоих концов
62 'отрезка не больше 0, то отрезок пересекает ось
63 'Y, если произведение координат Y обоих концов
64 'отрезка не больше 0, то он пересекает ось X
70 PX=Y1*Y2<=0
80 PY=X1*X2<=0
81 'При истинности одного из логических выражений
82 'к начальному числу добавить 1
90 IF PX THEN K1=K1+1
100 IF PY THEN K2=K2+1
110 NEXT I
120 PRINT "Пересекаются с осью X";K1
130 PRINT "Пересекаются с осью Y";K2

```

Решение на Паскале:

```

Var
  N,i,k1,k2 : word;

```



```

x1,y1,x2,y2 : real; {Координаты отрезков}
px,py : boolean;    {Переменные логического типа}
Begin
  Write('Введите количество отрезков ');
  readln(N);
  k1:=0; {Начальное количество отрезков, имеющих
          общую точку с осью X}
  k2:=0; {Начальное количество отрезков, имеющих
          общую точку с осью Y}
  for i:=1 to N do {Цикл на N повторов}
    begin
      write('Введите координаты концов ',i,'-го
            отрезка ');
      readln(x1,y1,x2,y2);
      {Если произведение координат X обоих концов отрезка
      не больше 0, то отрезок пересекает ось Y, если
      произведение координат Y обоих концов отрезка не
      больше 0, то отрезок пересекает ось X}
      px:=y1*y2<=0;
      py:=x1*x2<=0;
      {При истинности одного из логических выражений к
      начальному числу добавить 1, используем процедуру
      увеличения аргумента на 1}
      if px then inc(k1);
      if py then inc(k2);
    end;
    writeln('Пересекаются с осью x ',k1);
    writeln('Пересекаются с осью y ',k2);
  end.

```

14. Заданы координатами своих вершин  $n$  треугольников. Определить, сколько треугольников имеет периметр больше  $P1$  и меньше  $P2$ . ( $P1 < P2$ ).

Решение на Бейсике:

```

10 INPUT "Введите количество треугольников";N
20 INPUT "Введите P1 и P2";P1,P2
30 K=0 'Начальное количество треугольников,
31 'имеющих периметр больше P1 и меньше P2
40 FOR I=1 TO N 'Цикл на N повторов
50 PRINT I;"-й треугольник"
60 INPUT "Введите координаты точки A";AX,AY
70 INPUT "Введите координаты точки B";BX,BY
80 INPUT "Введите координаты точки C";CX,CY
81 'Найти длину сторон и периметр I-ого тр-ка

```

```

90 AB=SQR( (AX-BX) ^2+(AY-BY) ^2)
100 AC=SQR( (AX-CX) ^2+(AY-CY) ^2)
110 BC=SQR( (BX-CX) ^2+(BY-CY) ^2)
120 P=AB+AC+BC
121 'Если периметр больше P1 и меньше P2 ,то, k1
122 'увеличить на 1
130 IF P>P1 AND P<P2 THEN K=K+1
140 NEXT I      'Конец цикла
150 PRINT K      'Вывод количества треугольников

```

Решение на Паскале:

```

Var
  P1,P2 : real;
  p, : real;           {Периметр треугольника}
  ax,ay,bx,by,cx,cy : real; {Координаты вершин}
  i,N,k : word;
Begin
  Write('Введите количество треугольников ');
  readln(N);
  Write('Введите P1 и P2 ');
  readln(P1,P2);
  {Начальное количество треугольников с периметром >
  P1 и < P2}
  k:=0;
  for i:=1 to N do {Цикл на N повторов}
    begin
      writeln(i,'-й треугольник');
      write('Введите координаты точки А ');
      readln(ax,ay);
      write('Введите координаты точки В ');
      readln(bx,by);
      write('Введите координаты точки С ');
      readln(cx,cy);
      {Расчет периметра i-ого треугольника}
      p:=sqrt(sqr(ax-bx)+sqr(ay-by)) +
        sqrt(sqr(ax-cx)+sqr(ay-cy)) +
        sqrt(sqr(cx-bx)+sqr(cy-by));
      {Если периметр больше P1 и меньше P2, то k1
      увеличить на 1}
      if (p>P1) and (p<P2) then inc(k)
      end;
      {Конец цикла}
      writeln(k);
    end.

```

15. Заданы координатами своих вершин  $n$  прямоугольных треугольников с прямым углом в начале координат и катетами, лежащими на осях. Определить, сколько треугольников имеет гипотенузу больше  $L1$  и меньше  $L2$ . ( $L1 < L2$ ).

Решение на Бейсике:

```

10 INPUT "Введите количество треугольников";N
20 INPUT "Введите L1 и L2";L1,L2
21 'Начальное количество треугольников, имеющих
22 'гипотенузу согласно условию
30 K=0
40 FOR I=1 TO N 'Цикл на N повторов
50 PRINT I;"-й треугольник"
60 INPUT "Введите длину катета, лежащего на оси X";
   X
70 INPUT "Введите длину катета, лежащего на оси Y";
   Y
80 P=SQR(X^2+Y^2) 'Найти длину гипотенузы, по
81 'теореме Пифагора
82 'Если P больше L1 и меньше L2, k увеличить на 1
90 IF P>L1 AND P<L2 THEN K=K+1
100 NEXT I 'Конец цикла
110 PRINT K

```

Решение на Паскале:

```

Var
  L1,L2 : real;
  p,x,y : real; {Длина гипотенузы и двух катетов }
  i,N,k : word;
Begin
  Write('Введите количество треугольников ');
  readln(N);
  Write('Введите L1 и L2 ');
  readln(L1,L2);
  {Начальное количество треугольников, имеющих гипотенузу согласно условию}
  k:=0;
  for i:=1 to N do {Цикл на N повторов}
    begin
      writeln(i,'-й треугольник');
      write('Введите длину катета, лежащего на оси
        X ');
      readln(x);

```

```

write('Введите длину катета, лежащего на оси
      Y ');
readln(y);
{Найти длину гипотенузы, как корень от суммы
квадратов катетов. Если P больше L1 и меньше L2, k
увеличить на 1}
p:=sqrt(sqr(x)+sqr(y));
if (p>L1) and (p<L2) then inc(k)
end;
writeln(k);
end.

```

16. Заданы координатами своих вершин А,В,С n треугольников. Определить, сколько треугольников имеет сторону АВ, больше сторон АС и ВС. Считать, что треугольник не имеет сторон одинаковой длины.

Решение на Бейсике:

```

10 INPUT "Введите количество треугольников";N
11 'Начальное число треугольников, где сторона АВ
12 'больше двух других
20 K=0
30 FOR I=1 TO N           'Цикл на N повторов
40 PRINT I;"-й треугольник"
50 INPUT "Введите координаты точки А";AX,AY
60 INPUT "Введите координаты точки В";BX,BY
70 INPUT "Введите координаты точки С";CX,CY
71 'Находим длины сторон треугольников, если АВ
72 'больше ВС и АС, к k добавить 1
80 AB=SQR((AX-BX)^2+(AY-BY)^2)
90 AC=SQR((AX-CX)^2+(AY-CY)^2)
100 BC=SQR((BX-CX)^2+(BY-CY)^2)
110 IF AB>BC AND AB>AC THEN K=K+1
120 NEXT I 'Конец цикла
130 PRINT K

```

Решение на Паскале:

```

Var
  ax,ay,bx,by,cx,cy : real; {Координаты вершин}
  ab,ac,bc : real;          {Длины сторон}
  i,N,k : word;
Begin
  Write('Введите количество треугольников ');
  readln(N);
  k:=0;{Начальное число треугольников, где сторона

```

```

        АВ больше двух других}
for i:=1 to N do          {Цикл на N повторов}
begin
    writeln(i,'-й треугольник');
    write('Введите координаты точки А ');
    readln(ax,ay);
    write('Введите координаты точки В ');
    readln(bx,by);
    write('Введите координаты точки С ');
    readln(cx,cy);
    {Находим длины сторон треугольников, если АВ больше
    ВС и АС, к k добавить 1}
    ab:=sqrt(sqr(ax-bx)+sqr(ay-by));
    ac:=sqrt(sqr(ax-cx)+sqr(ay-cy));
    bc:=sqrt(sqr(cx-bx)+sqr(cy-by));
    if (ab>bc) and (ab>ac) then inc(k)
end;                      {Конец цикла}
writeln(k);
end.

```

17. Заданы своими координатами N точек на плоскости и два квадрата со сторонами L1 и L2 ( $L1 < L2$ ), параллельными осям координат и с центром симметрии в начале координат. Определить, сколько точек находится одновременно в обоих квадратах и сколько вне их.

Решение на Бейсике:

```

10 INPUT "Введите количество точек";N
20 INPUT "Введите длины сторон квадратов";L1,L2
30 K1=0      'Начальное число точек в обоих квадратах
40 K2=0      'Начальное число точек вне квадратов
50 FOR I=1 TO N          'Цикл на N повторов
60 PRINT "Введите координаты";I;"-ой точки"
70 INPUT X,Y
71 'Используя логическое выражение, определяем
72 'местоположение точки
80 P1 = ABS(X)<=L1/2 AND ABS(Y)<=L1/2
90 P2 = ABS(X)<=L2/2 AND ABS(Y)<=L2/2
91 'Если истинно первое выражение точка, принадле-
92 'жит обоим квадратам, иначе, если второе выраже-
93 'ние неверно, то ни одному
100 IF P1 THEN K1=K1+1 ELSE IF NOT P2 THEN K2=K2+1
110 NEXT I
120 PRINT "Принадлежит обоим";K1

```

```
130 PRINT "Вне";K2
```

Решение на Паскале:

```
Var
  L1,L2 : real;
  N,i,k1,k2 : word;
  p1,p2 : boolean;      {Переменные логического типа}
  x,y : real;            {Координаты точки}
Begin
  Write('Введите количество точек ');
  readln(N);
  Write('Введите длины сторон квадратов ');
  readln(L1,L2);
  k1:=0; {Начальное число точек в обоих квадратах}
  k2:=0; {Начальное число точек вне квадратов}
  for i:=1 to N do      {Цикл на N повторов}
    begin
      write('Введите координаты ',i,'-ой точки ');
      readln(x,y);
      {Используя логическое выражение, определяем
      местоположение точки}
      p1:=(abs(x)<=L1/2) and (abs(y)<=L1/2);
      p2:=(abs(x)<=L2/2) and (abs(y)<=L2/2);
      {Если истинно первое выражение, точка принадлежит
      обоим квадратам, иначе, если второе выражение не
      верно, то ни одному}
      if p1 then inc(k1)
      else if not p2 then inc(k2)
    end;      {Конец цикла}
    writeln('Принадлежит обоим ',k1);
    writeln('Вне ',k2)
  end.
```

18. Задано N четырехугольников со сторонами A, B, C, D. Определить, сколько среди них параллелограммов и ромбов.

Решение на Бейсике:

```
10 INPUT "ВВЕДИТЕ КОЛ-ВО ЧЕТУРЕХУГОЛЬНИКОВ";N
15 K1=0 : K2=0 'Начальное количество ромбов
17          'и параллелограммов
20 FOR I=1 TO N 'Цикл на N повторов
30 PRINT "ВВЕДИТЕ СТОРОНЫ";I;"-ГО ЧЕТЫРЕХУГОЛЬНИКА";
40 INPUT A,B,C,D 'Если длины всех сторон равны, òî
45          'k1 увеличиваем на 1
```

```

50 IF A=C AND B=D AND A=B THEN K1=K1+1 ELSE IF A=C AND B=D
    THEN K2=K2+1
60 NEXT I 'Конец цикла
70 PRINT "РОМБОВ-";K1;" ПАРАЛЛЕЛОГРАММОВ";K2

```

Решение на Паскале:

```

Var
  N: Word;
  a, b, c, d: Real;
  k1, k2: Word;
  i: Word;
Begin
  Write('Введите количество четырехугольников ');
  Readln(N);
  k1:=0; k2:=0; {Начальное количество ромбов и
                  параллелограммов}
  For i:=1 to N do {Цикл на N повторов}
    begin {Начало цикла}
      Write('Введите стороны ',i,'-го четырехугольника ');
      Readln(a,b,c,d);
      if (a=c) and (b=d) and (a=b) then inc(k1) {Если длины
                                                    всех сторон равны, òî k1 увеличиваем на 1}
      else {Иначе}
        if (a=c) and (b=d) then inc(k2) {Если равны длины
                                           противоположащих сторон, òî k1 увеличить на 1}
      end; {Конец цикла}
      Writeln('Ромбов-',k1,' параллелограммов-',k2) {Вывод
                                                         результата}
    end;
  End. {Окончание }

```

19. На плоскости задан N-угольник координатами своих вершин. Определить, сколько ребер имеют длину в диапазоне от D1 до D2.

Решение на Бейсике:

```

10 INPUT "ВВЕДИТЕ КОЛ-ВО УГЛОВ";N
20 INPUT "ВВЕДИТЕ D1 И D2";D1,D2
30 INPUT "ВВЕДИТЕ КООРДИНАТЫ 1-ОЙ ВЕРШИНЫ";XP,YP
40 XN=XP : YN=YP
45 K=0 'Начальное количество ребер, принадлежащее диапазону
50 FOR I=2 TO N 'Цикл на N повторов
60 PRINT "ВВЕДИТЕ КООРДИНАТЫ";I;"-Й ВЕРШИНЫ";
70 INPUT X,Y
80 L=SQR( (X-XP)^2+(Y-YP)^2) 'Вычисляем длину ребра
90 IF L>=D1 AND L<=D2 THEN K=K+1 'Если длина принадлежит

```

```

95           'диапазону от D1 до D2, òî k увеличить на 1
100 XP=X : YP=Y 'Присваиваем значения X,Y переменным XP,YP
105           'и возвращаемся в начало цикла
110 NEXT I
120 L=SQR((XN-XP)^2+(YN-YP)^2) 'Вычисляем длину последнего
125           'ребра N-угольника
130 IF L>=D1 AND L<=D2 THEN K=K+1 'Если длина принадлежит
135           'диапазону, òî K увеличить на 1
140 PRINT "В ДАННОМ ДИАПАЗОНЕ";K;" РЕБЕР"

```

Решение на Паскале:

```

Var
  N: Word;
  d1, d2: Real; {Диапазон}
  l: Real;
  k: Word;
  i: Word;      {Параметр цикла}
  x, y: Real;
  xp, yp, xn, yn: Real;
Begin
  Write('Введите количество углов ');
  Readln(N);
  Write('Введите D1 и D2 ');
  Readln(d1,d2);
  Write('Введите координаты 1-ой вершины ');
  Readln(xp,yp);
  xn:=xp;
  yn:=yp;
  k:=0;{Начальное количество ребер, принадлежащее диапазону}
  For i:=2 to N do {Цикл на N повторов}
    begin
      Write('Введите координаты ',i,'-ой вершины ');
      Readln(x,y);
      l:=sqrt(sqr(x-xp)+sqr(y-yp)); {Вычисляем длину ребра}
      {Если длина принадлежит диапазону от D1 до D2, òî k увеличить
на 1}
      if (l>=d1) and (l<=d2) then inc(k);
      xp:=x;      {Присваиваем значения x, y переменным xp,}
                  {yp и возвращаемся в начало цикла}
      yp:=y
    end;
  l:=sqrt(sqr(xn-xp)+sqr(yn-yp)); {Вычисляем длину}
                                  {последнего ребра N-угольника}

```



```

    if (l>=d1) and (l<=d2) then inc(k); {Если длина}
        {принадлежит диапазону то, k увеличить на 1}
    Writeln('В данном диапазоне всего ',k,' ребер')
End.

```

20. На плоскости задано N отрезков координатами своих концов. Определить, сколько отрезков не имеют ни одной общей точки с осями координат.

Решение на Бейсике:

```

10 INPUT "ВВЕДИТЕ КОЛ-ВО ОТРЕЗКОВ";N
15 K=0 'Начальное количество отрезков
20 FOR I=1 TO N 'Цикл на N повторов
30 PRINT "ВВЕДИТЕ КООРДИНАТЫ КОНЦОВ";I;"-ГО ОТРЕЗКА";
40 INPUT X1,Y1,X2,Y2
50 IF X1*X2>0 AND Y1*Y2>0 THEN K=K+1 'Если произведение
55 'координат X и произведение координат Y больше 0 то,
57 'отрезок не имеет ни одной точки с осями
60 NEXT I
70 PRINT K;"ОТРЕЗКОВ"

```

Решение на Паскале:

```

Var
    N: Word;
    x1, y1, x2, y2: Real;
    k: Word;
    i: Word;
Begin
    Write('Введите количество отрезков ');
    Readln(N);
    k:=0; {Начальное количество отрезков }
    For i:=1 to N do {Цикл на N повторов}
        begin
            Write('Введите координаты концов ',i,'-ого отрезка ');
            Readln(x1,y1,x2,y2);
            {Если произведение координат x и произведение координат y
            больше 0 то, отрезок не имеет ни одной точки с осями}
            if (x1*x2>0) and (y1*y2>0) then inc(k)
            end;
            Writeln(k,' отрезков')
        end.

```

21. На плоскости задано  $N$  отрезков координатами своих концов. Определить, сколько отрезков полностью принадлежат первой четверти координатной плоскости.

Решение на Бейсике:

```
10 INPUT "ВВЕДИТЕ КОЛ-ВО ОТРЕЗКОВ";N
15 K=0 'Начальное количество отрезков,
17 'принадлежащих первой четверти
20 FOR I=1 TO N 'Цикл на N повторов
30 PRINT "ВВЕДИТЕ КООРДИНАТЫ КОНЦОВ";I;"-ГО ОТРЕЗКА";
40 INPUT X1,Y1,X2,Y2
41 'Если все координаты концов отрезка >0 то, отрезок
42 'полностью принадлежит первой четверти
50 IF X1>0 AND X2>0 AND Y1>0 AND Y2>0 THEN K=K+1
60 NEXT I
70 PRINT K;"ОТРЕЗКОВ" 'Вывод результата
```

Решение на Паскале:

```
Var
  N: Word;
  x1, y1, x2, y2: Real;
  k: Word;
  i: Word;
Begin
  Write('Введите количество отрезков ');
  Readln(N);
  k:=0; {Начальное количество отрезков, принадлежащих
        первой четверти}
  For i:=1 to N do {Цикл на N повторов}
    begin
      Write('Введите координаты концов ',i,'-ого отрезка ');
      Readln(x1,y1,x2,y2);
      if (x1>0) and (x2>0) and (y1>0) and (y2>0) then inc(k)
      {Если все координаты концов отрезка >0 то, отрезок полностью
       принадлежит первой четверти}
    end;
    Writeln(k, ' отрезков') {Вывод результата}
  end.
```

22. На плоскости задано  $N$  отрезков координатами своих концов. Определить, сколько отрезков параллельны осям координат.

Решение на Бейсике:

```
10 INPUT "ВВЕДИТЕ КОЛ-ВО ОТРЕЗКОВ";N
```

```

15 K=0                                'Начальное кол-во отрезков
20 FOR I=1 TO N                        'Цикл на N повторов
30 PRINT "ВВЕДИТЕ КООРДИНАТЫ КОНЦОВ";I;"-ГО ОТРЕЗКА";
40 INPUT X1,Y1,X2,Y2
50 IF X1=X2 OR Y1=Y2 THEN K=K+1 'Если координаты X обоих
52 'концов отрезка или координаты Y равны, то отрезок
53 'параллелен одной из осей, K увеличить на 1
60 NEXT I
70 PRINT K;"ОТРЕЗКОВ"

```

Решение на Паскале:

```

Var
  N: Word;
  x1, y1, x2, y2: Real;
  k: Word;
  i: Word;
Begin
  Write('Введите количество отрезков ');
  Readln(N);
  k:=0;                                {Начальное кол-во отрезков }
  For i:=1 to N do                      {Цикл на N повторов}
    begin
      Write('Введите координаты концов ',i,'-ого отрезка ');
      Readln(x1,y1,x2,y2);
      if (x1=x2) or (y1=y2) then inc(k) {Если координаты x
      обоих концов отрезка или координаты y равны, то отрезок
      параллелен одной из осей, k увеличить на 1 }
    end;
    Writeln(k, ' отрезков')
  end.

```

23. На плоскости задан N-угольник координатами своих вершин. Определить, сколько его ребер имеют хотя бы одну общую точку с осью абсцисс.

Решение на Бейсике:

```

10 INPUT "Введите количество углов ";N
20 INPUT "Введите координаты 1-ой вершины "; XР, YР
30 XN=XР : YN=YР
40 K=0 'Начальное кол-во ребер, имеющих общие точки с
42 'осью абсцисс
50 FOR I=2 TO N 'Цикл на N повторов
60 PRINT "Введите координаты";I;"-ой вершины";

```

```

70 INPUT X, Y
80 IF YP*Y<=0 THEN K=K+1      'Если произведение координаты Y,
82 'обоих концов ребра <= 0, то ребро имеет общую точку с
83 'осью абсцисс
90 XP=X : YP=Y                'Значения координат X и Y передаются
92                             'переменным XP и YP
100 NEXT I
101 'Если произведение координаты Y, обоих концов последнего
102 'ребра N угольника <= 0, то ребро имеет общую точку с
103 'осью абсцисс
110 IF Y*YN<=0 THEN K=K+1
120 PRINT K;"ребер"          'Вывод результата

```

Решение на Паскале:

```

Var
  N: Word;
  k: Word;
  i: Word;
  x, y: Real;
  xp, yp, xn, yn: Real;
Begin
  Write('Введите количество углов ');
  Readln(N);
  Write('Введите координаты 1-ой вершины ');
  Readln(xp,yp);
  xn:=xp;
  yn:=yp;
  k:=0; {Начальное кол-во ребер, имеющих общие точки с
        осью абсцисс}
  For i:=2 to N do {Цикл на N повторов}
    begin
      Write('Введите координаты ',i,'-ой вершины ');
      Readln(x,y);
      if yp*y<=0 then inc(k); {Если произведение координаты
                              y, обоих концов ребра <= 0,
                              то ребро имеет общую точку с
                              осью абсцисс}
      xp:=x; {Значения координат x и y передаются
              переменным xp и yp}
      yp:=y
    end;
  if y*yn<=0 then inc(k); {Если произведение координаты y,
                          обоих концов последнего ребра

```

```

N-угольника <= 0, то ребро имеет
общую точку с осью абсцисс}
Writeln(k, ' ребер') {Вывод результата}
End.

```

24. На плоскости задан N-угольник координатами своих вершин. Определить, сколько его ребер имеют хотя бы одну общую точку с осью ординат.

Решение на Бейсике:

```

10 INPUT "Введите количество углов ";N
20 INPUT "Введите координаты 1-ой вершины "; XP, YP
30 XN=XP : YN=YP
40 K=0 'Начальное кол-во ребер, имеющих общую точку
50 FOR I=2 TO N 'Цикл на N повторов
60 PRINT "Введите координаты";I;"-ой вершины";
70 INPUT X, Y
80 IF XP*X<=0 THEN K=K+1 'Если произведение координат X
81 'обоих концов ребра меньше либо равно 0, то ребро имеет
82 'общую точку с осью ординат
90 XP=X : YP=Y
100 NEXT I
110 IF X*XN<=0 THEN K=K+1 'Если произведение координат X
111 'обоих концов последнего ребра N угольника меньше либо
112 'равно 0, то ребро имеет общую точку с осью ординат
120 PRINT K;"ребер" 'Вывод результата

```

Решение на Паскале:

```

Var
  N: Word;
  k: Word;
  i: Word;
  x, y: Real;
  xp, yp, xn, yn: Real;
Begin
  Write('Введите количество углов ');
  Readln(N);
  Write('Введите координаты 1-ой вершины ');
  Readln(xp, yp);
  xn:=xp;
  yn:=yp;
  k:=0; {Начальное кол-во ребер, имеющих общую точку}
  For i:=2 to N do {Цикл на N повторов}
    begin

```

```

Write('Введите координаты ',i,'-ой вершины ');
Readln(x,y);
if x*y<=0 then inc(k); {Если произведение координат x
обоих концов ребра меньше либо равно 0, то ребро имеет общую
точку с осью ординат}
  xp:=x;
  yp:=y
end;
{Конец цикла}
if x*yn<=0 then inc(k); {Если произведение координат x обоих
концов последнего ребра N угольника меньше либо равно 0, то
ребро имеет общую точку с осью ординат}
  Writeln(k,' ребер')
{Вывод результата}
End.

```

25. На плоскости задан N-угольник координатами своих вершин. Определить, сколько его вершин принадлежат осям координат.

Решение на Бейсике:

```

10 INPUT "Введите количество углов ";N
20 K=0 'Счетчик вершин, принадлежащих осям
30 FOR I=1 TO N 'Цикл на N повторов
40 PRINT "Введите координаты";I;"-ой вершины";
50 INPUT X, Y
60 IF Y*X=0 THEN K=K+1 'Если произведение координат
62 'равно 0, то K увеличить на 1
70 NEXT I
80 PRINT K;"вершин" 'Вывод результата

```

Решение на Паскале:

```

Var
  N: Word;
  k: Word;
  i: Word; {Счетчик вершин, принадлежащих осям}
  x, y: Real;
Begin
  Write('Введите количество углов ');
  Readln(N);
  k:=0;
  For i:=1 to N do {Цикл на N повторов}
    begin
      Write('Введите координаты ',i,'-ой вершины ');
      Readln(x,y);
      if x*y=0 then inc(k); {Если произведение координат
      равно 0, то k увеличить на 1}
    end;
  end;
  Writeln(k);
End.

```

```
end;  
Writeln(k, ' вершин')      {Вывод результата}  
End.
```