

Задание 2

На плоскости задано множество точек с целочисленными координатами, количество точек не превосходит 1000. Необходимо найти максимально возможную площадь невырожденного (то есть имеющего ненулевую площадь) треугольника, одна вершина которого расположена в начале координат, а две другие лежат на биссектрисах углов, образованных осями координат, и при этом принадлежат заданному множеству. Если такого треугольника не существует, необходимо вывести соответствующее сообщение.

Напишите эффективную, в том числе по используемой памяти, программу для решения этой задачи.

Перед текстом программы кратко опишите алгоритм решения и укажите язык программирования и его версию.

Входные данные

В первой строке задаётся N – количество точек в заданном множестве. Каждая из следующих строк содержит два целых числа – координаты очередной точки.

Пример входных данных:

```
3
6 6
-8 8
9 7
```

Выходные данные

Если искомый треугольник существует, программа должна напечатать одно число: максимально возможную площадь треугольника, удовлетворяющего условиям. Если искомый треугольник не существует, программа должна напечатать сообщение: «Треугольник не существует».

Пример выходных данных для приведённого выше примера входных данных:

```
48
```

Содержание верного ответа и указания по оцениванию (допускаются иные формулировки ответа, не искажающие его смысла)

Биссектрисами углов, образованных осями координат, служат две прямые: $y = x$ и $y = -x$. Очевидно, что вершины невырожденного треугольника должны лежать на разных биссектрисах, их координаты должны иметь вид (a, a) и $(b, -b)$. Площадь такого треугольника равна $ a \cdot b $. Эта площадь будет максимальной при максимальных значениях $ a $ и $ b $. Язык программирования – Паскаль; версия Free Pascal 2.6.

Пример правильной программы на Паскале

```
var
    N: integer; {количество точек}
    x,y: integer; {координаты очередной точки}
    amax, bmax: integer;
    s: integer; {площадь}
    i: integer;
begin
    readln(N);
    amax:=0; bmax:=0;
    for i:=1 to N do begin
        readln(x,y);
        if (x=y) and (abs(x)>amax) then amax:=abs(x);
        if (x=-y) and (abs(x)>bmax) then bmax:=abs(x);
    end;
    s:=amax*bmax;
    if s=0 then writeln('Треугольник не существует')
    else writeln(s)
end.
```

Указания по оцениванию	Баллы
Программа правильно работает для любых входных данных произвольного размера и находит ответ, не сохраняя входные данные в массиве. Допускается наличие в тексте программы одной синтаксической ошибки: пропущен или неверно указан знак пунктуации, неверно написано или пропущено зарезервированное слово языка программирования, не описана или неверно описана переменная, применяется операция, недопустимая для соответствующего типа данных (если одна и та же ошибка встречается несколько раз, то это считается за одну ошибку).	4
<p>Не выполнены условия, позволяющие поставить 4 балла, при этом программа работает верно, время работы линейно зависит от N, но размер используемой памяти зависит от количества точек. Например, входные данные запоминаются в массиве или другой структуре данных, размер которой соответствует числу N.</p> <p>Допускается одна из следующих ошибок:</p> <ol style="list-style-type: none"> 1) Пропущенная или неверная инициализация максимумов. В некоторых языках и системах программирования (например, в Бейсике) все переменные после создания имеют значение 0. В этом случае отсутствие инициализации нельзя считать ошибкой. 2) Вместо абсолютных значений (модулей) используются непосредственные значения координат. 3) Ошибка в сравнении, в результате которой в одном или нескольких местах находится минимальное значение вместо максимального. 4) Использование вещественных вычислений и операции извлечения 	3

корня для нахождения площади. 5) Неверная обработка случая отсутствия требуемого треугольника. Допускается наличие от одной до трёх синтаксических ошибок, описанных в критериях на 4 балла.	
Не выполнены условия, позволяющие поставить 3 или 4 балла, при этом программа работает верно, эффективно или нет. В частности, в 2 балла оцениваются переборные решения, в которых все исходные данные сохраняются в массиве, рассматриваются все возможные треугольники, из которых выбираются подходящие. Допускается наличие нескольких содержательных ошибок, описанных в критериях на 3 балла, и до пяти синтаксических ошибок, описанных в критериях на 4 балла.	2
Не выполнены условия, позволяющие поставить 2, 3 или 4 балла, но программа работает в отдельных частных случаях. 1 балл также ставится, если программа неработоспособна или не написана, но из пояснений видно, что экзаменуемый в целом верно представляет путь решения.	1
Не выполнены условия, позволяющие поставить 1, 2, 3 или 4 балла.	0
<i>Максимальный балл</i>	<i>4</i>

Решение экзаменуемого (работа № 3)

Точки, которые нас интересуют, лежат на прямых $x = y$ и $x = -y$, причём одна вершина должна лежать на одной прямой, а другая на другой. Треугольник будет прямоугольный, длины его сторон будут $|x_1| \cdot \sqrt{2}$ и $|x_2| \cdot \sqrt{2}$. Площадь треугольника будет $(|x_1| \cdot \sqrt{2}) \cdot (|x_2| \cdot \sqrt{2}) / 2 = |x_1| \cdot |x_2|$. Это значение максимально, когда каждое из чисел $|x_1|$, $|x_2|$ максимально возможное.

В программе читаем координаты точек и, если точка лежит на одной из биссектрис, записываем значение $|x|$ в соответствующий массив. Потом находим максимальное значение в каждом из массивов.

```
var
  N, ND1, ND2: integer;
  D1, D2: array [1..1000] of integer;
  Max1, Max2: integer; S: integer;
  i, x, y: integer;
begin
  readln(N);
  ND1:=0; ND2:=0;
  for i:=1 to N do begin
    readln(x,y);
    if (x=y) then begin
      ND1:=ND1+1;
      D1[ND1]:=abs(x);
    end;
    if (x=-y) then begin
      ND2:=ND2+1;
      D2[ND2]:=abs(y);
    end;
  end;
  Max1:=0; Max2:=0;
  for i:=1 to ND1 do begin
    if (D1[i] > Max1) then Max1:= D1[i];
  end;
  for i:=1 to ND2 do begin
    if (D2[i] > Max2) then Max2:= D2[i];
  end;
  S:=Max1*Max2;
  if (S=0) then writeln('Треугольник не существует')
  else writeln(S)
end.
```