

## ПРИМЕР ВСТУПИТЕЛЬНОГО ЭКЗАМЕНА ПО ИНФОРМАТИКЕ

№1. Сколько единиц в двоичной записи шестнадцатеричного числа  $5A7_{16}$ ?

№2. Денис заполнял таблицу истинности для выражения F. Он успел заполнить лишь небольшой фрагмент таблицы:

$x_1$	$x_2$	$x_3$	$x_4$	$x_5$	$x_6$	$x_7$	$x_8$	F
		0			0		1	0
	0			1		0		1
		0			1		0	0

Потом он написал 4 выражения, одно из которых является выражением F.

- 1)  $x_1 \vee \neg x_2 \vee x_3 \vee \neg x_4 \vee \neg x_5 \vee x_6 \vee \neg x_7 \vee \neg x_8$
- 2)  $x_1 \vee x_2 \vee x_3 \vee \neg x_4 \vee \neg x_5 \vee \neg x_6 \vee \neg x_7 \vee \neg x_8$
- 3)  $\neg x_1 \wedge \neg x_2 \wedge \neg x_3 \wedge x_4 \wedge x_5 \wedge x_6 \wedge x_7 \wedge x_8$
- 4)  $x_1 \wedge \neg x_2 \wedge x_3 \wedge \neg x_4 \wedge x_5 \wedge x_6 \wedge \neg x_7 \wedge x_8$

Под каким номером написано выражение F?

№3. На вход алгоритма подаётся двоичное число, в котором ровно 5 значащих разрядов. Алгоритм преобразует его по следующим правилам.

1. В конец числа (справа) дописывается 1, если количество единиц в его записи нечётно. В противном случае, если количество единиц чётно, дописывается 0. Например, число 11100 преобразуется в число 111001.

2. Та же операция применяется к полученному 6-значному двоичному числу.

3. Полученное двоичное число переводится в десятичную систему счисления.

Укажите десятичное число, которое может быть результатом работы алгоритма.

Вот 4 числа:

42, 96, 115, 222

Какое из них может получиться в результате работы алгоритма? Если таких чисел несколько, укажите большее из них.

№4. Ниже представлены две таблицы из базы данных. Каждая строка таблицы 2 содержит информацию о ребёнке и об одном из его родителей. Информация представлена значением поля ID в соответствующей строке таблицы 1. Определите на основании приведённых данных фамилию и инициалы тети Гайко В.В.. В ответе запишите соответствующее значение поля ID из таблицы 1.

*Пояснение: тётей считается сестра отца или матери.*

Таблица 1		
ID	Фамилия_И.О.	Пол
16	Белова И.М.	Ж
25	Гайко А.В.	М
26	Гайко В.А.	М
29	Гайко В.В.	М
33	Драль Т.А.	Ж
37	Драль Б.Г.	Ж
38	Драль Г.Г.	М
42	Нушич А.С.	Ж
47	Нушич В.А.	М
48	Венцель К.Г.	Ж
49	Венцель И.К.	М
55	Старова Н.В.	Ж
61	Брамс Г.В.	Ж
...	...	...

Таблица 2	
ID_Родителя	ID_Ребёнка
25	26
42	26
26	29
61	29
25	33
42	33
33	37
38	37
16	38
33	48
38	48
26	55
61	55
...	...

**№5.** Ольге нужно с помощью электронных таблиц построить таблицу квадратов двузначных чисел от 10 до 49.

Для этого сначала в диапазоне В1:К1 он записал числа от 0 до 9, и в диапазоне А2:А5 он записал числа от 1 до 4. Затем в ячейку В2 записал формулу квадрата двузначного числа (А2 – число десятков; В1 – число единиц), после чего скопировал её во все ячейки диапазона В2:К5. В итоге получил таблицу квадратов двузначных чисел. На рисунке ниже представлен фрагмент этой таблицы.

	А	В	С	Д	Е
1		<b>0</b>	<b>1</b>	<b>2</b>	<b>3</b>
2	<b>1</b>	100	121	144	169
3	<b>2</b>	400	441	484	529
4	<b>3</b>	900	961	1024	1089
5	<b>4</b>	1600	1681	1764	1849

Ниже написаны 4 формулы; среди них есть та, которая была записана в ячейку В2. Укажите номер этой формулы.

- 1)  $=($B$1+10*$A$2)^2$
- 2)  $=($B1+10*$A2)^2$
- 3)  $=($B1+10*A$2)^2$
- 4)  $=(B$1+10*$A2)^2$

*Примечание: знак \$ используется для обозначения абсолютной адресации.*

№6. Для кодирования некоторой последовательности, состоящей из букв А, В, С, D и Е, используется неравномерный двоичный код, позволяющий однозначно декодировать полученную двоичную последовательность. Вот этот код:

А – 00; В – 01; С – 101; D – 111; Е – 110. Требуется сократить для одной из букв длину кодового слова так, чтобы код по-прежнему можно было декодировать однозначно. Коды остальных букв меняться не должны.

Каким из указанных способов это можно сделать? Укажите номер правильного ответа.

- 1) это невозможно
- 2) для буквы D – 11
- 3) для буквы С – 10
- 4) для буквы Е – 11

№7. На числовой прямой даны два отрезка: P = [17; 40] и Q = [20; 57]. Выберите такой отрезок А, чтобы приведенная ниже формула была истинна при любом значении переменной x:

$$\neg(x \in A) \rightarrow ((x \in P) \wedge (x \in Q)) \rightarrow (x \in A)$$

- 1) [35; 70]
- 2) [19; 51]
- 3) [10; 31]
- 4) [0; 17]

Укажите номер выбранного ответа.

№8. При регистрации в компьютерной системе каждому пользователю выдаётся пароль, состоящий из 15 символов и содержащий только символы Ф, И, З, К, А (таким образом, используется 5 различных символов). Каждый такой пароль в компьютерной системе записывается минимально возможным и одинаковым целым количеством байт (при этом используют посимвольное кодирование и все символы кодируются одинаковым и минимально возможным количеством бит).

Определите объём памяти (в байтах), отводимый этой системой для записи 30 паролей.

№9. Исполнитель Чертёжник перемещается на координатной плоскости, оставляя след в виде линии. Чертёжник может выполнять команду **сместиться на (a, b)**, где a, b – целые числа. Эта команда перемещает Чертёжника из точки с координатами (x, y) в точку с координатами (x + a; y + b).

Например, если Чертёжник находится в точке с координатами (4, 2), то команда **сместиться на (2, -3)** переместит Чертёжника в точку (6, -1).

Цикл

ПОВТОРИ *число* РАЗ

*последовательность команд*

КОНЕЦ ПОВТОРИ

означает, что *последовательность команд* будет выполнена указанное *число* раз (число должно быть натуральным).

Чертёжнику был дан для исполнения следующий алгоритм (буквами n, a, b обозначены неизвестные числа):

НАЧАЛО

**сместиться на (1, 5)**

ПОВТОРИ  $n$  РАЗ  
 сместиться на  $(a, b)$   
 сместиться на  $(-3, -2)$   
 КОНЕЦ ПОВТОРИ  
 сместиться на  $(-31, -14)$   
 КОНЕЦ

После выполнения этого алгоритма Чертёжник возвратился в исходную точку. Укажите наибольшее возможное значение числа  $n$ .

№10. Все 4-буквенные слова, составленные из букв Е, И, О, У, записаны в алфавитном порядке и пронумерованы.

Вот начало списка:

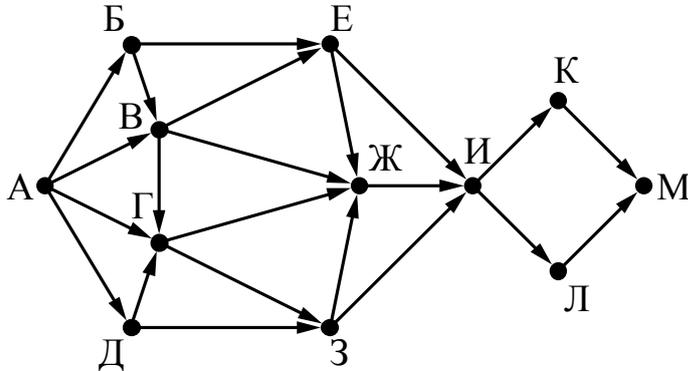
1. ЕЕЕЕ
2. ЕЕЕИ
3. ЕЕЕО
4. ЕЕЕУ

.....

Запишите слово, которое стоит под номером 25.

№11. Сколько единиц содержится в двоичной записи значения выражения:  
 $4^{1812} + 2^{1905} - 31$ ?

№12. На рисунке – схема дорог, связывающих города А, Б, В, Г, Д, Е, Ж, З, И, К, Л, М. По каждой дороге можно двигаться только в одном направлении, указанном стрелкой. Сколько существует различных путей из города А в город М?



№13. В языке запросов поискового сервера для обозначения логической операции «ИЛИ» используется символ «|», а для обозначения логической операции «И» – символ «&».

В таблице приведены запросы и количество найденных по ним страниц некоторого сегмента сети Интернет.

Запрос	Найдено страниц (в сотнях тысяч)
<i>Динамо</i>	35
<i>Спартак</i>	25
<i>Красс</i>	40
<i>Динамо   Спартак   Красс</i>	70
<i>Динамо &amp; Красс</i>	10
<i>Динамо &amp; Спартак</i>	0

Какое количество страниц (в сотнях тысяч) будет найдено по запросу *Спартак & Красс*? Считается, что все запросы выполнялись практически одновременно, так что набор страниц, содержащих все искомые слова, не изменялся за время выполнения запросов.

**№14.** Исполнитель Июль14 преобразует число, записанное на экране. У исполнителя три команды, которым присвоены номера:

1. Прибавь 1
2. Прибавь 2
3. Прибавь 4

Первая из них увеличивает число на экране на 1, вторая увеличивает это число на 2, а третья – на 4. Программа для исполнителя Май4 – это последовательность команд.

Сколько есть программ, которые **число 41** преобразуют в **число 51**?

**№15.** Сколько существует различных наборов значений логических переменных  $x_1, x_2, \dots, x_8, y_1, y_2, \dots, y_8$ , которые удовлетворяют всем перечисленным ниже условиям?

$$(x_1 \vee y_1) \wedge ((\neg x_1 \vee \neg y_1) \rightarrow (\neg x_2 \vee \neg y_2)) = 1$$

$$(x_2 \vee y_2) \wedge ((\neg x_2 \vee \neg y_2) \rightarrow (\neg x_3 \vee \neg y_3)) = 1$$

...

$$(x_7 \vee y_7) \wedge ((\neg x_7 \vee \neg y_7) \rightarrow (\neg x_8 \vee \neg y_8)) = 1$$

$$(x_8 \vee y_8) = 1$$

В ответе **не нужно** перечислять все различные наборы значений переменных  $x_1, x_2, \dots, x_8, y_1, y_2, \dots, y_8$ , при которых выполнена данная система равенств. В качестве ответа Вам нужно указать количество таких решений.

**№16.** Определите, какое число будет напечатано в результате выполнения программы. Для Вашего удобства программа представлена на четырёх языках программирования.

Бейсик	Паскаль
<pre> DIM I, S, N AS INTEGER I = 0 S = 44 N = 1 WHILE S &gt; 0   S = S - 8   N = N * (I + 1)   I = I + 1 WEND PRINT(N) </pre>	<pre> var i, s, n: integer; begin   i := 0;   s := 44;   n := 1;   while s &gt; 0 do     begin       s := s - 8;       n := n * (i + 1);       i := i + 1     end;   writeln(n) end. </pre>

<b>Си</b>	<b>Алгоритмический язык</b>
<pre>#include&lt;stdio.h&gt; void main() {   int i, s, n;   i = 0;   s = 44;   n = 1;   while (s &gt; 0)   {     s = s - 8;     n = n * (i + 1);     i++;   }   printf("%d\n", n); }</pre>	<pre><u>алг</u> <u>нач</u>   <u>цел</u> i, s, n   i := 0   s := 44   n := 1   <u>нц пока</u> s &gt; 0     s := s — 8     n := n * (i + 1)     i := i + 1   <u>кц</u>   <u>вывод</u> n <u>кон</u></pre>

**№17.** В программе используется одномерный целочисленный массив *A* с индексами от 0 до 9. Значения элементов равны 8; 9; 5; 2; 7; 1; 6; 0; 4; 3 соответственно, т.е.  $A[0] = 7$ ;  $A[1] = 9$  и т.д.

Определите значение переменной *c* после выполнения следующего фрагмента программы, записанного ниже на разных языках программирования.

<b>Бейсик</b>	<b>Паскаль</b>
<pre>c = 0 FOR i = 1 TO 9   IF A(i) &lt; A(0) THEN     c = c + 1     t = A(i)     A(i) = A(0)     A(0) = t   ENDIF NEXT i</pre>	<pre>c := 0; for i := 1 to 9 do   if A[i] &lt; A[0] then   begin     c := c + 1;     t := A[i];     A[i] := A[0];     A[0] := t   end;</pre>
<b>Си</b>	<b>Алгоритмический язык</b>
<pre>c = 0; for (i = 1; i &lt;= 9; i++)   if (A[i] &lt; A[0])   {     c++;     t = A[i];     A[i] = A[0];     A[0] = t;   }</pre>	<pre>c := 0 <u>нц для</u> i <u>от</u> 1 <u>до</u> 9   <u>если</u> A[i] &lt; A[0] <u>то</u>     c := c + 1     t := A[i]     A[i] := A[0]     A[0] := t   <u>все</u> <u>кц</u></pre>

№18. Ниже на четырёх языках программирования записан рекурсивный алгоритм F.

Бейсик	Паскаль
<pre>SUB F(n)   PRINT n   IF n &lt; 5 THEN     F(n + 1)     F(n * 2)   END IF END SUB</pre>	<pre>procedure F(n: integer); begin   writeln(n);   if n &lt; 5 then     begin       F(n + 1);       F(n * 2)     end   end end</pre>
Си	Алгоритмический язык
<pre>void F(int n) {   printf("%d\n", n);   if (n &lt; 5)   {     F(n + 1);     F(n * 2);   } }</pre>	<pre>алг F(цел n) нач   <u>ВЫВОД</u> n, нс   <u>если</u> n &lt; 5 <u>то</u>     F(n + 1)     F(n * 2)   <u>все</u> <u>КОН</u></pre>

Чему равна сумма всех чисел, напечатанных на экране при выполнении вызова F(2)?

№19 Ниже на четырёх языках записан алгоритм. Получив на вход число  $x$ , этот алгоритм печатает два числа  $a$  и  $b$ . Укажите наименьшее из таких чисел  $x$ , при вводе которых алгоритм печатает сначала 3, а потом 15.

Бейсик	Паскаль
<pre>DIM X, A, B AS INTEGER INPUT X A=0: B=0 WHILE X &gt; 0   A = A+1   B = B +(X MOD 10)   X = X \ 10 WEND PRINT A PRINT B</pre>	<pre>var x, a, b: integer; begin   readln(x);   a:=0; b:=0;   while x&gt;0 do     begin       a:=a + 1;       b:=b + (x mod 10);       x:=x div 10;     end;   writeln(a); write(b); end.</pre>

Си	Алгоритмический
<pre>#include&lt;stdio.h&gt; void main() {   int x, a, b;   scanf("%d", &amp;x);   a=0; b=0;   while (x&gt;0){     a = a+1;     b = b +(x%10);     x = x/10;   }   printf("%d\n%d", a, b); }</pre>	<pre><u>алг</u> <u>нач</u>   <u>цел</u> x, a, b   <u>ввод</u> x   a:=0; b:=0   <u>нц пока</u> x&gt;0     a:=a+1     b:=b+mod(x,10)     x:=div(x,10)   <u>кц</u>   <u>вывод</u> a, <u>нс</u>, b <u>кон</u></pre>

**№20.** Определите, какое число будет напечатано в результате выполнения следующего алгоритма (для Вашего удобства алгоритм представлен на четырёх языках)

Бейсик	Паскаль
<pre>DIM A, B, T, M, R AS INTEGER A = -20: B = 20 M = A: R = F(A) FOR T = A TO B   IF F(T) &lt; R THEN     M = T     R = F(T)   END IF NEXT T PRINT R  FUNCTION F(x)   F = (29-x)*(29-x)+9 END FUNCTION</pre>	<pre>var a,b,t,M,R :integer; Function F(x: integer):integer; begin   F := (29-x)*(29-x)+9; end; BEGIN a := -20; b := 20; M := a; R := F(a); for t := a to b do begin   if (F(t) &lt; R) then begin     M := t;     R := F(t);   end; end; write(R); END.</pre>

Си	Алгоритмический
<pre>#include&lt;stdio.h&gt; int F(int x) {     return (29-x)*(29-x)+9; } void main() {     int a, b, t, M, R;     a = -20; b = 20;     M = a; R = F(a);     for (t=a; t&lt;=b; t++){         if (F(t) &lt; R) {             M = t; R = F(t);         }     }     printf("%d", R); }</pre>	<pre><u>алг</u> <u>нач</u>     <u>цел</u> a, b, t, R, M     a := -20; b := 20     M := a; R := F(a)     <u>нц для</u> t <u>от</u> a <u>до</u> b         <u>если</u> F(t) &lt; R             <u>то</u>                 M := t; R := F(t)         <u>все</u>     <u>кц</u>     <u>вывод</u> R <u>кон</u> <u>алг цел</u> F(<u>цел</u> x) <u>нач</u>     <u>знач</u> := (29-x)*(29-x)+9 <u>кон</u></pre>

**№21.** На обработку поступает последовательность из четырёх неотрицательных целых чисел (некоторые числа могут быть одинаковыми). Нужно написать программу, которая выводит на экран количество чётных чисел в исходной последовательности и сумму таких чисел. Если чётных чисел нет, требуется на экран вывести «NO». Известно, что вводимые числа не превышают 1000. Ниже эта программа для Вашего удобства приведена на четырёх языках программирования.

Бейсик	Паскаль
<pre>CONST n = 4 count = 0 sum = 0 FOR I = 1 TO n     INPUT x     IF x mod 2 = 0 THEN         count = count + 1         sum = x     END IF NEXT I IF sum &gt; 0 THEN     PRINT count     PRINT sum ELSE     PRINT "NO" ENDIF</pre>	<pre>const n = 4; var i, x: integer; var sum, count: integer; begin     count := 0;     sum := 0;     for i := 1 to n do         begin             read(x);             if x mod 2 = 0 then                 begin                     count := count + 1;                     sum := x                 end             end;         if sum &gt; 0 then             begin</pre>

	<pre>writeln(count); writeln(sum) end else writeln('NO') end.</pre>
<b>Си</b>	<b>Алгоритмический язык</b>
<pre>#include &lt;stdio.h&gt; #define n 4  void main(void) { int i, x; int sum, count; count = 0; sum = 0; for (i = 1; i &lt;= n; i++) { scanf("%d",&amp;x); if (x % 2 == 0) { count++; sum = x; } } if (sum &gt; 0) { printf("%d\n", count); printf("%d\n", sum); } else printf("NO\n"); }</pre>	<pre><u>алг</u> <u>нач</u> цел n = 4 цел i, x цел sum, count count := 0 sum := 0 <u>нц для i от 1 до n</u> <u>ввод x</u> если mod(x, 2) = 0 <u>то</u> count := count + 1 sum := x <u>все</u> <u>кц</u> если sum &gt; 0 <u>то</u> <u>вывод</u> count, <u>нс</u> <u>вывод</u> sum, <u>нс</u> иначе <u>вывод</u> "NO" <u>все</u> <u>кон</u></pre>

Последовательно выполните следующее.

1. Напишите, что выведет эта программа при вводе последовательности:

20, 93, 40, 39

2. Приведите пример такой последовательности, содержащей хотя бы одно чётное число, что, несмотря на ошибки, программа печатает правильный ответ.

3. Найдите все ошибки в этой программе (их может быть одна или несколько). Известно, что каждая ошибка затрагивает только одну строку и может быть исправлена без изменения других строк. Для каждой ошибки:

1) выпишите строку, в которой сделана ошибка;

2) укажите, как исправить ошибку, т.е. приведите правильный вариант строки.

Достаточно указать ошибки и способ их исправления для одного языка программирования.

Обратите внимание, что требуется найти ошибки в имеющейся программе, а не написать свою, возможно, использующую другой алгоритм решения. Исправление ошибки должно затрагивать только строку, в которой находится ошибка.

*Примечание: 0 – чётное число.*

**№22.** Дан целочисленный массив из 20 элементов. Элементы массива могут принимать целые значения от 0 до 10 000 включительно. Опишите на естественном языке или на одном из языков программирования алгоритм, позволяющий найти и вывести количество пар элементов массива, произведение которых нечётно, а сумма не кратна 5. Под парой подразумевается два подряд идущих элемента массива.

Исходные данные объявлены так, как показано ниже на примерах для некоторых языков программирования и естественного языка. Запрещается использовать переменные, не описанные ниже, но разрешается не использовать некоторые из описанных переменных.

<p><b>Бейсик</b></p> <pre>N = 20 DIM A(N) AS LONG DIM I, J, K AS LONG FOR I = 1 TO N   INPUT A(I) NEXT I ... END</pre>	<p><b>Паскаль</b></p> <pre>const N = 20; var a: array [1..N] of longint; i, j, k: longint; begin   for i := 1 to N do     readln(a[i]);   ... end.</pre>
<p><b>Си</b></p> <pre>#include &lt;stdio.h&gt; #define N 20 void main() {   long a[N];   long i, j, k;   for (i = 0; i &lt; N; i++)     scanf("%ld", &amp;a[i]);   ... }</pre>	<p><b>Алгоритмический язык</b></p> <pre>алг нач   цел N = 20   цел таб a[1:N]   цел i, j, k   нц для i от 1 до N     ввод a[i]   кц   ... кон</pre>
<p><b>Естественный язык</b></p> <p>Объявляем массив A из 20 элементов.          Объявляем целочисленные переменные I, J, K.          В цикле от 1 до 20 вводим элементы массива A с 1-го по 20-й.          ...</p>	

**№23.**

Два игрока, Петя и Ваня, играют в следующую игру. Перед игроками лежит куча камней. Игроки ходят по очереди, первый ход делает Петя. За один ход игрок может добавить в кучу **один** или **три** камня или увеличить количество камней в куче **в два раза**. Например,

имея кучу из 15 камней, за один ход можно получить кучу из 16, 18 или 30 камней. У каждого игрока, чтобы делать ходы, есть неограниченное количество камней.

Игра завершается в тот момент, когда количество камней в куче становится не менее 47.

Победителем считается игрок, сделавший последний ход, т.е. первым получивший кучу, в которой будет 47 или больше камней.

В начальный момент в куче было  $S$  камней;  $1 \leq S \leq 46$ .

Будем говорить, что игрок имеет *выигрышную стратегию*, если он может выиграть при любых ходах противника. Описать стратегию игрока – значит описать, какой ход он должен сделать в любой ситуации, которая ему может встретиться при различной игре противника.

Выполните следующие задания. Во всех случаях обосновывайте свой ответ.

#### Задание 1

- а) Укажите все такие значения числа  $S$ , при которых Петя может выиграть в один ход. Обоснуйте, что найдены все нужные значения  $S$ , и укажите выигрышающие ходы.
- б) Укажите такое значение  $S$ , при котором Петя не может выиграть за один ход, но при любом ходе Пети Ваня может выиграть своим первым ходом. Опишите выигрышную стратегию Вани.

#### Задание 2

Укажите два таких значения  $S$ , при которых у Пети есть выигрышная стратегия, причём одновременно выполняются два условия:

- Петя не может выиграть за один ход;
- Петя может выиграть своим вторым ходом независимо от того, как будет ходить Ваня.

Для каждого указанного значения  $S$  опишите выигрышную стратегию Пети.

#### Задание 3

Укажите значение  $S$ , при котором одновременно выполняются два условия:

- у Вани есть выигрышная стратегия, позволяющая ему выиграть первым или вторым ходом при любой игре Пети;
- у Вани нет стратегии, которая позволит ему гарантированно выиграть первым ходом.

Для указанного значения  $S$  опишите выигрышную стратегию Вани.

Постройте дерево всех партий, возможных при этой выигрышной стратегии Вани (в виде рисунка или таблицы). На рисунке на рёбрах дерева указывайте, кто делает ход; в узлах – количество камней в позиции.

**№24.** На плоскости дан набор точек с целочисленными координатами. Необходимо найти четырёхугольник наибольшей площади с вершинами в этих точках, две вершины которого лежат на оси  $Ox$ , а две оставшиеся – по разные стороны от оси  $Oy$ .

Напишите эффективную, в том числе по памяти, программу, которая будет решать эту задачу. Перед текстом программы кратко опишите алгоритм решения задачи и укажите используемый язык программирования и его версию.

#### **Описание входных данных**

В первой строке вводится одно целое положительное число – количество точек  $N$ .

Каждая из следующих  $N$  строк содержит два целых числа: сначала координата  $x$ , затем координата  $y$  очередной точки.

#### **Описание выходных данных**

Программа должна вывести одно число – максимальную площадь четырёхугольника, удовлетворяющего условиям задачи. Если такого четырёхугольника не существует, программа должна вывести ноль.

**Пример входных данных:**

6  
0 0  
2 0  
0 2  
3 -3  
-5 -5  
6 6

**Пример выходных данных для приведённого выше примера входных данных:**

11