

Отметьте **ТИПЫ**, используемые в программе.

```
import java.io.*;
import java.util.*;

public class Sum
{
    public static void main(String[] args)
    {
        Scanner in = new Scanner(System.in);
        PrintWriter out = new PrintWriter(System.out);

        int a = in.nextInt();
        int b = in.nextInt();
        out.println(a + b);

        in.close();
        out.flush();
        out.close();
    }
}
```

Выберите один или несколько ответов:

- out
- String
- a
- println
- java.io
- args
- Scanner
- PrintWriter
- nextInt

1

Допишите класс, вычисляющий сумму двух введенных с консоли чисел.

При этом **НЕ используйте** пробельные символы. **Важны** регистр и все остальные символы (точки и пр.).

```
public class Sum {  
    public static  main(String[] args)  
    {  
        Scanner scanner = new Scanner(System.in);  
  
        int a = scanner.  
        int b = scanner.  
        .println(a + b);  
    }  
}
```

Программа состоит из единственного класса Test, выводящего на консоль "Hello, world!". Вставьте пропущенные символы. При этом **НЕ используйте** пробельные символы. Регистр и все остальные символы (точки, точки с запятой и пр.) важны.

```
public    
{  
    public static void (String [] args)   
        System..println("Hello, world!");  
      
}
```

Минимальное значение, которое можно поместить в переменную типа `byte`, равно...

Выберите один или несколько ответов:

- 255
- 256
- 127
- 128
- 0

Переменной типа `int` можно присваивать значения...

Выберите один или несколько ответов:

- "567"
- 5
- "5"
- 567
- 5

in Scanner nextInt println System.out System.out.println System.out.print System.out.printf System.out.format System.out.println System.out.println System.out.print System.out.printf System.out.format System.out.println System.out.print System.out.printf System.out.format System.out.println

Перетаскивая выше приведенные фрагменты, составьте верную последовательность для считывания с клавиатуры строки в переменную f

```
 f = (new  (.)).();
```

В результате работы фрагмента программы

```
System.out.print(a + b + "\t");
```

(a и b - действительные числа) на экране будет напечатано...

Выберите один ответ:

- строка из двух чисел и табуляции
- строка из одного числа
- строка из двух чисел
- ошибка компиляции
- строка из одного числа и табуляции

Какое значение нужно поставить вместо подчеркивания в коде:

```
int a = ____;  
while (a > 0){  
    a++;  
    a %= a;  
    System.out.println("*");  
}
```

Чтобы программа вывела ровно три звездочки?

Введите числовой ответ или символы

TL - если будет бесконечный цикл в любом случае;

CE - если программа не скомпилируется в любом случае;

ER - если программа при каких-то значениях закончит работу корректно, но выполнить условие невозможно.

Какое число нужно поставить на место подчеркивания в коде:

```
int a = ____;  
while (a > 0){  
    System.out.println("*");  
    a++;  
    a %= 1000;  
}
```

Чтобы программа вывела 1001 звездочку?

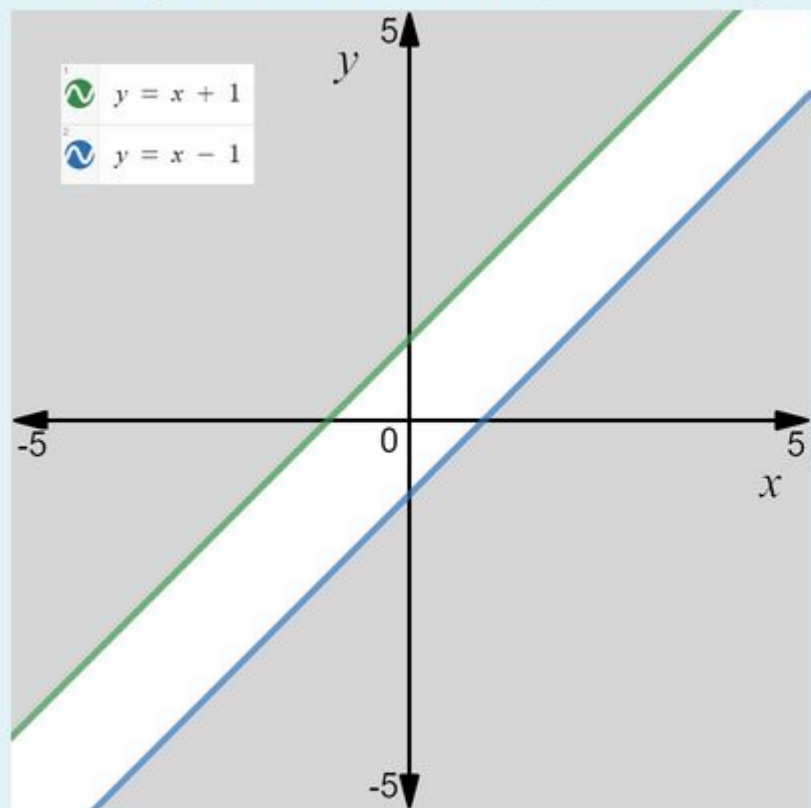
Введите числовой ответ или символы

TL - если будет бесконечный цикл в любом случае;

ER - если программа при каких-то значениях закончит работу корректно, но выполнить условие невозможно.

5

Напишите условие на языке Java, которое соответствует закрашенной серым цветом области (не в диагональной полосе).



Ответом должно быть логическое выражение, включающее переменные x и y , истинное в том и только в том случае, если точка (x, y) принадлежит закрашенной области.

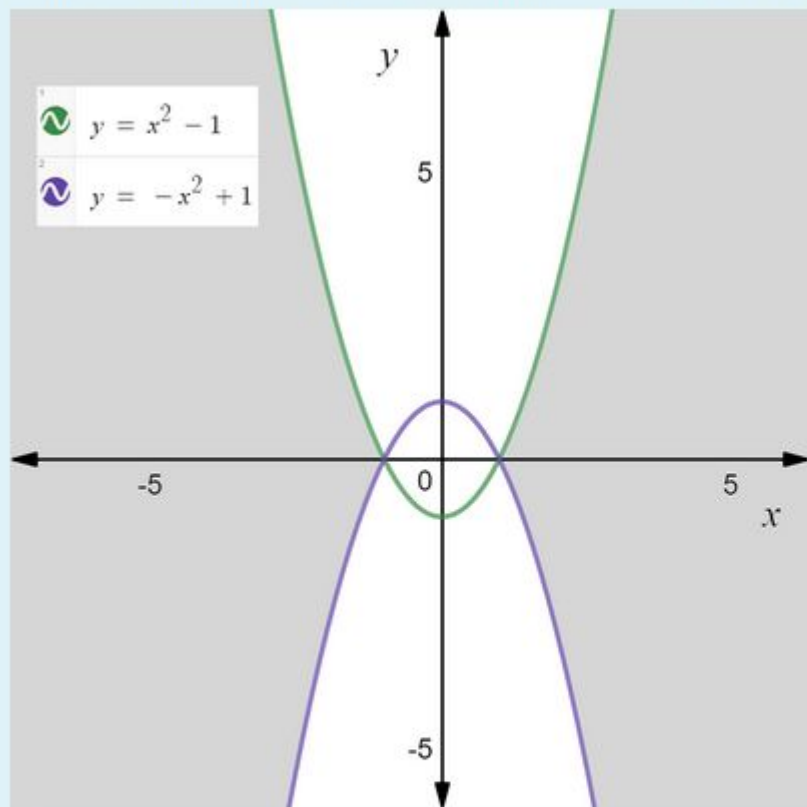
Границы не входят.

Например, корректным по форме, но не по содержанию ответом является

```
x == y && (x * x < 10 || x != y)
```

5

Напишите условие на языке Java, которое соответствует закрашенной серым цветом области (вне парабол).



Ответом должно быть логическое выражение, включающее переменные x и y , истинное в том и только в том случае, если точка (x, y) принадлежит закрашенной области.

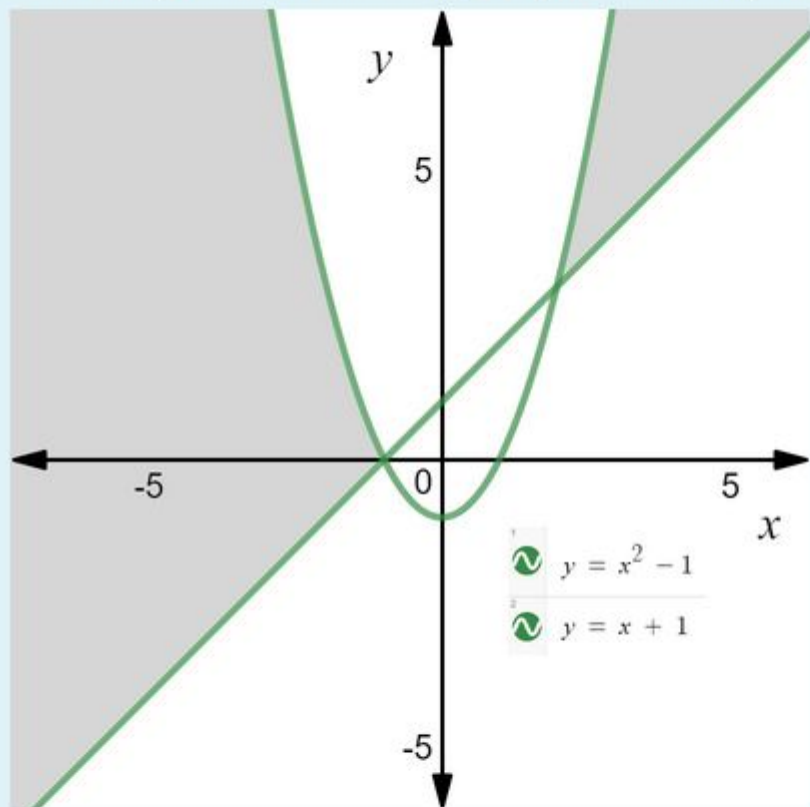
Границы не входят.

Например, корректным по форме, но не по содержанию ответом является

```
x == y && (x * x < 10 || x != y)
```


5

Напишите условие на языке Java, которое соответствует закрашенной серым цветом области.



Ответом должно быть логическое выражение, включающее переменные x и y , истинное в том и только в том случае, если точка (x, y) принадлежит закрашенной области.

Границы не входят.

Например, корректным по форме, но не по содержанию ответом является

```
x == y && (x * x < 10 || x != y)
```

Для вычисления меньшего из двух неравных чисел n и m в Java применима конструкция

Выберите один или несколько ответов:

- `min = m < n ? n : m;`
- `min = n < m ? n : m;`
- `min = n < m ? m : n;`
- `min = m > n ? n : m;`
- `min = m == n ? m : n;`

Аналогом фрагмента кода

```
if (!(a == 2 || b != 3)) {f();}
```

является ..

Выберите один ответ:

- `if (a != 2) if (b == 3) {f();}`
- `if (a + b != 5) {f();}`
- `if (a != 2) {f();}`
`else if (b == 3) {f();}`
- `if (a != 2 || b != 3) {f();}`
- `if (a != 2 && b != 3) {f();}`

Допишите недостающее ключевое слово

```
int i= 3;
    _____ (i) {
    case 0:
        System.out.print(i+10);break;
    case 1:
        System.out.print(i/10);break;
    default:
        System.out.print(i*0.1);break;
    }
```

Какое значение должно содержать i, чтобы вывелось на экран "здравствуй мир"

```
char i= '__';
switch (i) {
    case 'a':
        System.out.print("hello");break;
    case 'b':
        System.out.print("здравствуй");break;
    default:
        System.out.print("not defined");break;
}
System.out.print(" мир");
```

8

Каким значением необходимо инициализировать переменную `i`, чтобы фрагмент кода вывел 4 строки?

```
int i = ;  
while (i > 8){  
    System.out.println(i);  
    i--;  
}
```

Вставьте целое число, при котором данным фрагментом кода будет выведено на экран 5 строк.

```
int i = -5;  
while (i < ){  
    System.out.println(i);  
    i++;  
}
```

n = 1 n-- - n++ n > 0 n = 0 n < 2

Вставьте пропущенные знаки и числа, чтобы for вывел ровно n звездочек (переменная n определена ранее)
Нужно использовать не все элементы

```
for (  ;  ;  ) System.out.println("*");
```

Соберите строку, которая выводит на консоль сто звездочек.

```
int i = 0; for ( i < 200; ) i += 2 {System.out.println("*");}
```

```

for (int k = 0; k < a.length; k++) {
    if (a[k] % 2 == 0) System.out.print(a[k] + " ");
    else System.out.println(a[k]);
}

```

Эквивалентными приведенному фрагменту по результату работы с целочисленным массивом являются конструкции...

Выберите один или несколько ответов:

```

int p = 0;
while (p != a.length) {
    if (a[k] % 2 != 0) System.out.println(a[k]);
    else System.out.print(a[k] + " ");
    k++;
}

```

```

int k = 0;
for (; k < a.length; k++) {
    System.out.print(a[k] + " ");
    if (a[k] % 2 == 0) break;
    System.out.println();
}

```

```

int k = -1;
do {
    ++k;
    System.out.print(a[k] + " ");
    if (a[k] % 2 != 0) continue;
    System.out.println();
} while (k != a.length);

```

```

int p;
while (p < a.length) {
    if (a[k] % 2 != 0) continue;
    System.out.println(a[k]);
    else System.out.print(a[k] + " ");
}

```

```

int k = 0;
for (; k < a.length; k++) {
    System.out.print(a[k] + " ");
    if (a[k] % 2 == 0) continue;
    System.out.println();
}

```

Для фрагмента программы

```
Scanner input = new Scanner(System.in);
double i = input.nextDouble();
m:
for (double k = 0; k < i; k += 2.5) {
    System.out.print(k + ":");
    for (int j = 0; j < i; j++) {
        System.out.print("a");
        if (j > k) break m;
        System.out.print(",");
    }
}
```

определите наименьшее целое значение i , при котором на экране будет выведено

0.0:a,a

11

Чему равен элемент с индексом 5, для массива

```
int a[] = new int[]{1,1,2,3,5,8,16,32,64};
```

Индекс последнего элемента для массива из 25 элементов равен ...

12

В приведенном фрагменте реализован поиск максимума из отрицательных элементов в массиве чисел, которые лежат в промежутке [-10,5].

В массиве гарантированно есть хотя бы одно отрицательное число.

```
int array[] = new int[10];  
int max =  ;  
for (int i: array)  
    if ( i<0 && i>max )  
        max = i;
```

Каким МАКСИМАЛЬНЫМ значением необходимо инициализировать переменную max?

В приведенном фрагменте кода реализован поиск максимального элемента в массиве array.

```
int array[] = new int[10];
    for (int i = 0; i < array.length; i++)
        array[i] = (int) (1 + Math.random() * 11 );
int max = _____;
for (int i: array)
    if (max < i) max = i;
```

Корректные варианты инициализации переменной max ...

Выберите один или несколько ответов:

 10 Integer.MIN_VALUE array[0] 11 Integer.MAX_VALUE 2 1

Функции, определение которых вызовет ошибку компиляции ...

Выберите один или несколько ответов:

- `int f(float a, b){ return a+b;}`
- `String f(){ return "a+b";}`
- `void f(int d){ return d++;}`
- `short f(String a, int b){ return (short)b++;}`
- `float f(int a, double b) { return a+b;}`
- `double f(double a, int b){return a+b+" "};`

`void` `x & int y` `double x` `x + y` `x, int y` `int`

Выберите нужные фрагменты и постройте функцию, вычисляющую сумму двух целых чисел x и y:

`f(int` `)``{return` `};`

Какое значение должна иметь переменная b, чтобы на экран было выведено -160

```
public class Test {  
    public static int a=0,b;  
    public static void main(String[] args) {  
        b=__;  
        if(b==0){  
            int a=-10;  
        } else {  
            int a=10;  
        }  
        for(int i=1;i<5;i++) a+=a;  
        System.out.println(a);  
    }  
}
```

Выберите один ответ:

- любое
- 0
- 1
- ни один из вариантов

Что нужно дописать , чтобы на экран было выведено 0

```
public class Test {  
    public static int a=0,b=0;  
    public static void main(String[] args) {  
        if(b==0){  
            __ a=-10;  
        }  
        System.out.println(a);  
    }  
}
```

Выберите один ответ:

ничего не нужно

любой числовой тип

условие не может быть выполнено

b=

m 1 i + j == 0 i < matrix.length 0 j++ j j < 0 i < m i < j 2 i m - 1 m - j - 1

Перетаскивая приведенные выше элементы, допишите код так, чтобы он заполнял массив "змейкой".

Допускается использование не всех элементов

```
int n = in.nextInt(), m = in.nextInt();
int[][] matrix = new int[n][m];
for (int i = 0; [ ]; i++) {
    if (i % 2 == 0) {
        for (int j = 0; j < matrix[i].length; j++) {
            matrix[[ ]][[ ]] = j + m * i;
        }
    } else {
        for (int j = [ ]; j < matrix[i].length; j++) {
            matrix[i][[ ]] = j + m * i;
        }
    }
}
```

Пример "змейки" при вводе 5 4

```
0 1 2 3
7 6 5 4
8 9 10 11
15 14 13 12
16 17 18 19
```

1 2 maxX x < m x--; x >= 0 maxY++; x++ m m - 1

Перетаскивая приведенные выше элементы, допишите код так, чтобы он заполнял массив "диагоналями".

Допускается использование не всех элементов

```
int x = 0, y = 0, maxX = 0, maxY = 0;
int number = 0;
int n = in.nextInt(), m = in.nextInt();
int[][] a = new int[n][m];
while (maxX <  || maxY < n) {
    x = ;
    y = maxY;
    do {
        a[y][x] = number;
        number++;
        
        y++;
    } while ( && y < n);
    if (maxX == m - 1) {
        
    } else {
        maxX++;
    }
}
}
```

Пример заполнения "диагоналями" при вводе 5 4

```
0 1 3 6
2 4 7 10
5 8 11 14
9 12 15 17
13 16 18 19
```

Напишите программу, которая находит в массиве целых чисел количество элементов, равных максимальному.

Возможно решать задачу по-разному, например, использовать дополнительные массивы или даже не использовать массивы совсем.

Главное - вывести правильный ответ.

*Вы должны ввести **полный** текст программы на Java: необходимые импорты, один `public` класс в пакете по умолчанию (в коде отсутствует подстрока `package`), функцию `main()`.*

Комментарии на кириллице НЕ разрешены.

Ввод осуществляется с консоли, вывод - на консоль.

Программа должна работать точно по протоколу, который показан в примерах.

Например, нельзя выводить ничего лишнего, скажем, фразу "Введите N".

Входные данные

Первая строка содержит размер массива N .

Во второй строке через пробел задаются N чисел – элементы массива.

Гарантируется, что данные корректны и $N < 10001$.

Выходные данные

Программа должна вывести два целых числа, разделив их пробелом: максимальный элемент массива и количество элементов массива, равных максимальному.

Пример:

Входные данные	Выходные данные
6 1 2 3 2 3 3	3 3

Напишите программу, которая находит в массиве целых чисел количество элементов, равных максимальному.

Возможно решать задачу по-разному, например, использовать дополнительные массивы или даже не использовать массивы совсем.

Главное - вывести правильный ответ.

*Вы должны ввести **полный** текст программы на Java: необходимые импорты, один `public` класс в пакете по умолчанию (в коде отсутствует подстрока `package`), функцию `main()`.*

Комментарии на кириллице НЕ разрешены.

Ввод осуществляется с консоли, вывод - на консоль.

Программа должна работать точно по протоколу, который показан в примерах.

Например, нельзя выводить ничего лишнего, скажем, фразу "Введите N".

Входные данные

Первая строка содержит размер массива N .

Во второй строке через пробел задаются N чисел – элементы массива.

Гарантируется, что данные корректны и $N < 10001$.

Выходные данные

Программа должна вывести два целых числа, разделив их пробелом: максимальный элемент массива и количество элементов массива, равных максимальному.

Пример:

Входные данные	Выходные данные
6 1 2 3 2 3 3	3 3

Напишите программу, которая находит в массиве целых чисел номера элементов, равных минимальному.
Нумерация элементов начинается **с единицы**.

Возможно решать задачу по-разному, например, использовать дополнительные массивы или даже не использовать массивы совсем.

Главное - вывести получившуюся последовательность.

*Вы должны ввести **полный** текст программы на Java: необходимые импорты, один public класс в пакете по умолчанию (в коде отсутствует подстрока package), функцию main().*

Комментарии на кириллице НЕ разрешены.

Ввод осуществляется с консоли, вывод - на консоль.

Программа должна работать точно по протоколу, который показан в примерах.

Например, нельзя выводить ничего лишнего, скажем, фразу "Введите N".

Входные данные

Первая строка содержит размер массива N .

Во второй строке через пробел задаются N чисел – элементы массива. Гарантируется, что $N < 10001$.

Гарантируется, что входные данные корректны и $N < 10001$.

Выходные данные

Программа должна вывести в одной строке номера элементов массива, равных минимальному, разделив их пробелами.

Пример:

Входные данные	Выходные данные
5 1 2 1 3 1	1 3 5
10 1 1 1 1 1 1 1 1 1 1	1 2 3 4 5 6 7 8 9 10