Q1

The Pan Am 73 flight from Bombay to New York en route to Karachi and Frankfurt was hijacked by a few Palestinian terrorists at the Karachi International Airport.

The senior flight purser Neerja Banhot had to wither her fear and start evacuating the passengers on board. She pleaded the hijackers to release the oldest and the youngest person in the aircraft. Heeding to her plea the chief of the hijacker agreed to let go the oldest and the youngest. Given the ages of the passengers find the oldest and the youngest.

**Input Format**

The first line of input consists of an integer n, corresponding to the number of passengers in the aircraft.

The next line consists of the age of passengers separated by a space.

**Constraints**

No constrains

**Output Format**

The output prints the youngest and oldest separated by a space.

Print Invalid Input if n or any one of the ages is negative.

**Sample Input 0**

5

18 17 19 12 16

**Sample Output 0**

Youngest=12

Oldest=19

**Sample Input 1**

7

67 23 44 77 24 21 56

**Sample Output 1**

Youngest=21

Oldest=77

**Contest ends in 4 days**

**Submissions:**

[155](https://www.hackerrank.com/contests/array-statement-java/challenges/youngest-oldest-1/leaderboard)

**Max Score:**

10

**Difficulty:**

Medium

**Rate This Challenge:**

More

1

import java.util.\*;

2

public class Main {

3

   public static void main(String[] args) {

4

       Scanner sc = new Scanner(System.in);

5

       try {

6

           int n = sc.nextInt();

7

           if (n <= 0) {

8

               System.out.println("Invalid Input");

9

               return;

10

          }

11

           int[] ages = new int[n];

12

           for (int i = 0; i < n; i++) {

13

               ages[i] = sc.nextInt();

14

               if (ages[i] < 0) {

15

                   System.out.println("Invalid Input");

16

                   return;

17

              }

18

          }

19

           int youngest = Arrays.stream(ages).min().getAsInt();

20

           int oldest = Arrays.stream(ages).max().getAsInt();

21

           System.out.println("Youngest=" + youngest);

22

           System.out.println("Oldest=" + oldest);

23

      } catch (Exception e) {

24

           System.out.println("Invalid Input");

25

      }

26

  }

27

}

Line: 27 Col: 2

Run Code Submit Code

Upload Code as File

Test against custom input

* [**Testcase 0**](https://www.hackerrank.com/#testcase1)
* [**Testcase 1**](https://www.hackerrank.com/#testcase2)

**Congratulations, you passed the sample test case.**

Click the **Submit Code** button to run your code against all the test cases.

**Input (stdin)**

5

18 17 19 12 16

**Your Output (stdout)**

Youngest=12

Oldest=19

**Expected Output**

Youngest=12

Oldest=19

Q2

Karthick decided to go to his friend's home. He has reached his friend's home Street. In that place, lots of buildings are there. Then he started to search for his friend's home. He forgot the door number. But he knows the address very well. Finally, he found his friend's home. He wants to know the door number of his friend's home address. Could you please tell me how did he find the door number of his friend's home?

**Input Format**

Input consists of number of buildings in that street and Door numbers for the each home. Karthick friend's home address.

**Constraints**

No constraints

**Output Format**

If he found the home addrress after that returns the door number. If he not found the home,Return the value -1.

**Sample Input 0**

5

7 9 5 3 2

5

**Sample Output 0**

Door Number is 002-DN

**Sample Input 1**

4

8 9 6 4

0

**Sample Output 1**

-1

**Sample Input 2**

4

5 6 7 8

5

**Sample Output 2**

Door Number is 000-DN

**Contest ends in 4 days**

**Submissions:**

[159](https://www.hackerrank.com/contests/array-statement-java/challenges/array-176/leaderboard)

**Max Score:**

10

**Difficulty:**

Medium

**Rate This Challenge:**

More

1

import java.io.\*;

2

import java.util.\*;

3

public class Solution {

4

   public static void main(String[] args) {

5

       Scanner sc = new Scanner(System.in);

6

       int n = sc.nextInt();

7

       int[] doors = new int[n];

8

       for (int i = 0; i < n; i++) {

9

           doors[i] = sc.nextInt();

10

      }

11

       int search = sc.nextInt();

12

       int index = -1;

13

       for (int i = 0; i < n; i++) {

14

           if (doors[i] == search) {

15

               index = i;

16

               break;

17

          }

18

      }

19

       if (index == -1) {

20

           System.out.println(-1);

21

      } else {

22

           String result = String.format("%03d-DN", index);

23

           System.out.println("Door Number is " + result);

24

      }

25

  }

26

}

Line: 2 Col: 8

Run Code Submit Code

Upload Code as File

Test against custom input

* [**Testcase 0**](https://www.hackerrank.com/#testcase1)
* [**Testcase 1**](https://www.hackerrank.com/#testcase2)
* [**Testcase 2**](https://www.hackerrank.com/#testcase3)

**Congratulations, you passed the sample test case.**

Click the **Submit Code** button to run your code against all the test cases.

**Input (stdin)**

5

7 9 5 3 2

5

**Your Output (stdout)**

Door Number is 002-DN

**Expected Output**

Door Number is 002-DN

Q3

* [**Problem**](https://www.hackerrank.com/contests/array-statement-java/challenges/difference-of-the-character-1)
* [**Submissions**](https://www.hackerrank.com/contests/array-statement-java/challenges/difference-of-the-character-1/submissions)
* [**Leaderboard**](https://www.hackerrank.com/contests/array-statement-java/challenges/difference-of-the-character-1/leaderboard)
* [**Discussions**](https://www.hackerrank.com/contests/array-statement-java/challenges/difference-of-the-character-1/forum)

Given a string S(input consisting) of '' and '#'. The length of the string is variable. The task is to find the minimum number of '' or '#' to make it a valid string. The string is considered valid if the number of '' and '#' are equal. The '' and '#' can be at any position in the string.

Note : The output will be a positive or negative integer based on number of '\*' and '#' in the input string.

(*>#): positive integer (#>*): negative integer (#=\*): 0 Example 1:

Input 1:

*\** -> Value of S

Output :

0 → number of \* and # are equal

**Input Format**

Input consist of one string

**Constraints**

No constraints

**Output Format**

Execute the given output format.check the output data whether it is odd or even.

**Sample Input 0**

#\*\*#\*\*

**Sample Output 0**

The Difference of the character in the given string: 02

**Sample Input 1**

\*\*##\*\*#

**Sample Output 1**

The Difference of the character in the given string: 001

**Sample Input 2**

##\*#\*\*

**Sample Output 2**

0

**Contest ends in 4 days**

**Submissions:**

[108](https://www.hackerrank.com/contests/array-statement-java/challenges/difference-of-the-character-1/leaderboard)

**Max Score:**

10

**Difficulty:**

Medium

**Rate This Challenge:**

More

1

import java.io.\*;

2

import java.util.\*;

3

​

4

public class Solution {

5

   public static void main(String[] args) {

6

       Scanner sc = new Scanner(System.in);

7

       String s = sc.nextLine().trim();

8

       int stars = 0, hashes = 0;

9

       for (int i = 0; i < s.length(); i++) {

10

           char c = s.charAt(i);

11

           if (c == '\*') stars++;

12

           else if (c == '#') hashes++;

13

      }

14

       int diff = Math.abs(stars - hashes);

15

       if (diff == 0) {

16

           System.out.println(0);

17

      } else {

18

           System.out.println("The Difference of the character in the given string: "

19

                              + String.format("%03d", diff));

20

      }

21

  }

22

}

Line: 1 Col: 1

Run Code Submit Code

Upload Code as File

Test against custom input

Q4

sela wants to know how to find the Count of the Positive and Negative Integer Number in the list.could you please help her to find it and implements the program.

**Input Format**

Input Consists of integer list values

**Constraints**

No constraints

**Output Format**

Print the Number of Positive and Negative integer in the list.

**Sample Input 0**

5

3 4 6 -1 -2

**Sample Output 0**

Count of Positive Integer is 3.00

Count of Negative Integer is 2.00

**Sample Input 1**

4

2 7 3 9

**Sample Output 1**

Count of Positive Integer is 4.00

Count of Negative Integer is 0.00

**Contest ends in 4 days**

**Submissions:**

[140](https://www.hackerrank.com/contests/array-statement-java/challenges/count-the-positive-and-negative-integer-number-1/leaderboard)

**Max Score:**

10

**Difficulty:**

Medium

**Rate This Challenge:**

More

1

import java.io.\*;

2

import java.util.\*;

3

​

4

public class Solution {

5

   public static void main(String[] args) {

6

       Scanner sc = new Scanner(System.in);

7

       int n = sc.nextInt();

8

       int posCount = 0, negCount = 0;

9

​

10

       for (int i = 0; i < n; i++) {

11

           int num = sc.nextInt();

12

           if (num > 0) posCount++;

13

           else if (num < 0) negCount++;

14

      }

15

​

16

       System.out.printf("Count of Positive Integer is %.2f%n", (double)posCount);

17

       System.out.printf("Count of Negative Integer is %.2f%n", (double)negCount);

18

  }

19

}

Line: 19 Col: 2

Run Code Submit Code

Upload Code as File

Test against custom input

* [**Testcase 0**](https://www.hackerrank.com/#testcase1)
* [**Testcase 1**](https://www.hackerrank.com/#testcase2)

**Congratulations, you passed the sample test case.**

Click the **Submit Code** button to run your code against all the test cases.

**Input (stdin)**

5

3 4 6 -1 -2

**Your Output (stdout)**

Count of Positive Integer is 3.00

Count of Negative Integer is 2.00

**Expected Output**

Count of Positive Integer is 3.00

Count of Negative Integer is 2.00

Q5

Vihaan wants to find the array mean values using programming.Could you please find the array mean value and implements it.

**Input Format**

Input consists of list values.

**Constraints**

No constraints

**Output Format**

Print the array mean values

**Sample Input 0**

5

1 2 3 4 5

**Sample Output 0**

Array Mean Value is 3.00

**Sample Input 1**

3

6 4 3

**Sample Output 1**

Array Mean Value is 4.33

**Sample Input 2**

1

7

**Sample Output 2**

Array Mean Value is 7.00

**Contest ends in 4 days**

**Submissions:**

[129](https://www.hackerrank.com/contests/array-statement-java/challenges/array-mean-4/leaderboard)

**Max Score:**

10

**Difficulty:**

Medium

**Rate This Challenge:**

More

1

import java.io.\*;

2

import java.util.\*;

3

​

4

public class Solution {

5

   public static void main(String[] args) {

6

       Scanner sc = new Scanner(System.in);

7

       int n = sc.nextInt();

8

       double sum = 0;

9

       for (int i = 0; i < n; i++) {

10

           sum += sc.nextInt();

11

      }

12

       double mean = sum / n;

13

       System.out.printf("Array Mean Value is %.2f", mean);

14

  }

15

}

Line: 15 Col: 2

Run Code Submit Code

Upload Code as File

Test against custom input

* [**Testcase 0**](https://www.hackerrank.com/#testcase1)
* [**Testcase 1**](https://www.hackerrank.com/#testcase2)
* [**Testcase 2**](https://www.hackerrank.com/#testcase3)

**Congratulations, you passed the sample test case.**

Click the **Submit Code** button to run your code against all the test cases.

**Input (stdin)**

5

1 2 3 4 5

**Your Output (stdout)**

Array Mean Value is 3.00

**Expected Output**

Array Mean Value is 3.00

Q6

Write a program to count the number of distinct elements in an array.

**Input Format**

Input consists of 1 integer and 1 array. The first integer corresponds to the size of the array.

**Constraints**

No Constraints

**Output Format**

Print the no of the distinct element

**Sample Input 0**

6

2

6

6

4

5

5

**Sample Output 0**

There are 4 distinct elements in the array.

**Sample Input 1**

4

1

1

1

1

**Sample Output 1**

There are 1 distinct element in the array.

**Contest ends in 4 days**

**Submissions:**

[116](https://www.hackerrank.com/contests/array-statement-java/challenges/count-distinct-elements-8/leaderboard)

**Max Score:**

10

**Difficulty:**

Medium

**Rate This Challenge:**

More

1

import java.io.\*;

2

import java.util.\*;

3

​

4

public class Solution {

5

   public static void main(String[] args) {

6

       Scanner sc = new Scanner(System.in);

7

       int n = sc.nextInt();   // size of the array

8

       Set<Integer> distinctSet = new HashSet<>();

9

       for (int i = 0; i < n; i++) {

10

           int num = sc.nextInt();

11

           distinctSet.add(num);   // add elements to set (duplicates ignored)

12

      }

13

       System.out.println("There are " + distinctSet.size() + " distinct element" + (distinctSet.size() > 1 ? "s" : "") + " in the array.");

14

  }

15

}

Line: 15 Col: 2

Run Code Submit Code

Upload Code as File

Test against custom input

* [**Testcase 0**](https://www.hackerrank.com/#testcase1)
* [**Testcase 1**](https://www.hackerrank.com/#testcase2)

**Congratulations, you passed the sample test case.**

Click the **Submit Code** button to run your code against all the test cases.

**Input (stdin)**

6

2

6

6

4

5

5

**Your Output (stdout)**

There are 4 distinct elements in the array.

**Expected Output**

There are 4 distinct elements in the array.

Q7

Dhiya wants to search for the presence of an element in an array.could you please help her to search it and implements in program.

**Input Format**

Input consists of one integer and list of array elements. First input corresponds to the array size. Second input corresponds to the array elements. Third input corresponds to finding an array element.

**Constraints**

No constraints

**Output Format**

Print the Statement whether the given elements is Presented in the array or not.

**Sample Input 0**

6

1

2

3

4

5

6

7

**Sample Output 0**

7 is not presented in an array.

**Sample Input 1**

5

2

5

3

6

2

3

**Sample Output 1**

3 is presented in an array.

**Contest ends in 4 days**

**Submissions:**

[114](https://www.hackerrank.com/contests/array-statement-java/challenges/search-an-element-7/leaderboard)

**Max Score:**

10

**Difficulty:**

Medium

**Rate This Challenge:**

More

1

import java.io.\*;

2

import java.util.\*;

3

​

4

public class Solution {

5

   public static void main(String[] args) {

6

       Scanner sc = new Scanner(System.in);

7

​

8

       int n = sc.nextInt();          // size of the array

9

       int[] arr = new int[n];

10

​

11

       for (int i = 0; i < n; i++) {

12

           arr[i] = sc.nextInt();     // array elements

13

      }

14

​

15

       int search = sc.nextInt();     // element to find

16

       boolean found = false;

17

​

18

       for (int i = 0; i < n; i++) {

19

           if (arr[i] == search) {

20

               found = true;

21

               break;

22

          }

23

      }

24

​

25

       if (found) {

26

           System.out.println(search + " is presented in an array.");

27

      } else {

28

           System.out.println(search + " is not presented in an array.");

29

      }

30

  }

31

}

Line: 31 Col: 2

Run Code Submit Code

Upload Code as File

Test against custom input

* [**Testcase 0**](https://www.hackerrank.com/#testcase1)
* [**Testcase 1**](https://www.hackerrank.com/#testcase2)

**Congratulations, you passed the sample test case.**

Click the **Submit Code** button to run your code against all the test cases.

**Input (stdin)**

6

1

2

3

4

5

6

7

**Your Output (stdout)**

7 is not presented in an array.

**Expected Output**

7 is not presented in an array.

Q8

Write a Java program to find the k largest elements in a given array. Elements in the array can be in any order.

**Input Format**

First Input corresponds to the array size Second input corresponds to the array elements

**Constraints**

No constraints

**Output Format**

Execute the output in the given format.

**Sample Input 0**

5

7 6 9 3 2

2

**Sample Output 0**

The elements are in the order: 2 3 6 7 9

The Kth value is 2 and Largest elements are 9 7

**Sample Input 1**

6

17 21 12 26 13 31

4

**Sample Output 1**

The elements are in the order: 12 13 17 21 26 31

The Kth value is 4 and Largest elements are 31 26 21 17

**Contest ends in 4 days**

**Submissions:**

[108](https://www.hackerrank.com/contests/array-statement-java/challenges/find-k-largest-elements-1/leaderboard)

**Max Score:**

10

**Difficulty:**

Medium

**Rate This Challenge:**

More

1

import java.io.\*;

2

import java.util.\*;

3

public class Solution{

4

public static void main(String[] args){

5

Scanner sc=new Scanner(System.in);

6

int n=sc.nextInt();

7

int[] arr=new int[n];

8

for(int i=0;i<n;i++)arr[i]=sc.nextInt();

9

int k=sc.nextInt();

10

Arrays.sort(arr);

11

System.out.print("The elements are in the order: ");

12

for(int i=0;i<n;i++){

13

System.out.print(arr[i]+(i==n-1?"":" "));

14

}

15

System.out.println();

16

int kk=Math.min(k,n);

17

System.out.print("The Kth value is "+k+" and Largest elements are ");

18

for(int i=n-1,cnt=0;cnt<kk;i--,cnt++){

19

System.out.print(arr[i]+(cnt==kk-1?"":" "));

20

}

21

System.out.println();

22

}

23

}

24

​

Line: 24 Col: 1

Run Code Submit Code

Upload Code as File

Test against custom input

* [**Testcase 0**](https://www.hackerrank.com/#testcase1)
* [**Testcase 1**](https://www.hackerrank.com/#testcase2)

**Congratulations, you passed the sample test case.**

Click the **Submit Code** button to run your code against all the test cases.

**Input (stdin)**

5

7 6 9 3 2

2

**Your Output (stdout)**

The elements are in the order: 2 3 6 7 9

The Kth value is 2 and Largest elements are 9 7

**Expected Output**

The elements are in the order: 2 3 6 7 9

The Kth value is 2 and Largest elements are 9 7

Q9

Write a program to find the sum of positive square elements in an array.

**Input Format**

Input consists of 1 integer and 1 array. Integer corresponds to the size of the array.

**Constraints**

No Constraints

**Output Format**

Print the sum of positive square elements in an array.

**Sample Input 0**

3

2

4

6

**Sample Output 0**

56

**Sample Input 1**

5

6

4

2

-5

3

**Sample Output 1**

65

**Sample Input 2**

4

-2

-4

-3

-7

**Sample Output 2**

0

**Contest ends in 3 days**

**Submissions:**

[71](https://www.hackerrank.com/contests/array-statement-java/challenges/sum-of-positive-square-1/leaderboard)

**Max Score:**

10

**Difficulty:**

Medium

**Rate This Challenge:**

More

1

import java.io.\*;

2

import java.util.\*;

3

public class Solution{

4

public static void main(String[] args){

5

Scanner sc=new Scanner(System.in);

6

int n=sc.nextInt();

7

int[] arr=new int[n];

8

for(int i=0;i<n;i++)

9

arr[i]=sc.nextInt();

10

int sum=0;

11

for(int i=0;i<n;i++){

12

if(arr[i]>0)

13

sum+=arr[i]\*arr[i];

14

}

15

System.out.println(sum);

16

}

17

}

18

​

Line: 18 Col: 1

Run Code Submit Code

Upload Code as File

Test against custom input

* [**Testcase 0**](https://www.hackerrank.com/#testcase1)
* [**Testcase 1**](https://www.hackerrank.com/#testcase2)
* [**Testcase 2**](https://www.hackerrank.com/#testcase3)

**Congratulations, you passed the sample test case.**

Click the **Submit Code** button to run your code against all the test cases.

**Input (stdin)**

3

2

4

6

**Your Output (stdout)**

56

**Expected Output**

56

Q10

The function accepts an integers arr of size ’length’ as its arguments you are required to return the sum of second largest element from the even positions and second smallest from the odd position of given ‘arr’

Assumption:

All array elements are unique Treat the 0th position as even

**Input Format**

Input

arr:3 2 1 7 5 4

Output

7

**Constraints**

No constraints

**Output Format**

Second largest among even position elements(1 3 5) is 3 Second smallest among odd position element is 4 Thus output is 3+4 = 7

**Sample Input 0**

7

1 8 0 2 3 5 6

**Sample Output 0**

Sum=8

**Contest ends in 3 days**

**Submissions:**

[82](https://www.hackerrank.com/contests/array-statement-java/challenges/second-array-max-and-min-1-1/leaderboard)

**Max Score:**

10

**Difficulty:**

Medium

**Rate This Challenge:**

More

1

import java.io.\*;

2

import java.util.\*;

3

public class Solution{

4

public static void main(String[] args){

5

Scanner sc=new Scanner(System.in);

6

int n=sc.nextInt();

7

int[] arr=new int[n];

8

for(int i=0;i<n;i++)

9

arr[i]=sc.nextInt();

10

ArrayList<Integer> even=new ArrayList<>();

11

ArrayList<Integer> odd=new ArrayList<>();

12

for(int i=0;i<n;i++){

13

if(i%2==0) even.add(arr[i]);

14

else odd.add(arr[i]);

15

}

16

Collections.sort(even);

17

Collections.sort(odd);

18

int secondLargestEven=even.size()>=2?even.get(even.size()-2):even.get(even.size()-1);

19

int secondSmallestOdd=odd.size()>=2?odd.get(1):odd.get(0);

20

int sum=secondLargestEven+secondSmallestOdd;

21

System.out.println("Sum="+sum);

22

}

23

}

Line: 23 Col: 2

Run Code Submit Code

Upload Code as File

Test against custom input

* [**Testcase 0**](https://www.hackerrank.com/#testcase1)

**Congratulations, you passed the sample test case.**

Click the **Submit Code** button to run your code against all the test cases.

**Input (stdin)**

7

1 8 0 2 3 5 6

**Your Output (stdout)**

Sum=8

**Expected Output**

Sum=8

Q1

Write a program that reads an positive integer and count the number of digits the number

**Input Format**

Input consists of one integer.

**Constraints**

1<=num>=10000000

**Output Format**

Execute the count of the digit integer.

**Sample Input 0**

67578

**Sample Output 0**

The count of the given integer is: 5

**Sample Input 1**

9876543445

**Sample Output 1**

Enter a Valid Input

**Contest ends in 3 days**

**Submissions:**

[108](https://www.hackerrank.com/contests/looping-statement-java/challenges/count-digits-in-an-integer-1/leaderboard)

**Max Score:**

10

**Difficulty:**

Medium

**Rate This Challenge:**

More

1

import java.io.\*;

2

import java.util.\*;

3

public class Solution {

4

   public static void main(String[] args) {

5

       Scanner sc = new Scanner(System.in);

6

       long num = sc.nextLong();

7

       if (num < 1 || num > 10000000) {

8

           System.out.println("Enter a Valid Input");

9

      } else {

10

           int count = String.valueOf(num).length();

11

           System.out.println("The count of the given integer is: " + count);

12

      }

13

  }

14

}

Line: 14 Col: 2

Run Code Submit Code

Upload Code as File

Test against custom input

* [**Testcase 0**](https://www.hackerrank.com/#testcase1)
* [**Testcase 1**](https://www.hackerrank.com/#testcase2)

**Congratulations, you passed the sample test case.**

Click the **Submit Code** button to run your code against all the test cases.

**Input (stdin)**

67578

**Your Output (stdout)**

The count of the given integer is: 5

**Expected Output**

The count of the given integer is: 5

Q2

Print the below pattern

**Input Format**

Get the integer value

**Constraints**

n value should be within 0 to 9

**Output Format**

print the patten

**Sample Input 0**

4

**Sample Output 0**

A

B C

D E F

G H I J

**Sample Input 1**

3

**Sample Output 1**

A

B C

D E F

**Sample Input 2**

12

**Sample Output 2**

Invalid Input

**Contest ends in 3 days**

**Submissions:**

[155](https://www.hackerrank.com/contests/looping-statement-java/challenges/print-a-pattern-4/leaderboard)

**Max Score:**

10

**Difficulty:**

Medium

**Rate This Challenge:**

More

1

import java.io.\*;

2

import java.util.\*;

3

public class Solution {

4

   public static void main(String[] args) {

5

       Scanner sc = new Scanner(System.in);

6

       int n = sc.nextInt();

7

       if (n < 0 || n > 9) {

8

           System.out.println("Invalid Input");

9

           return;

10

      }

11

       char ch = 'A';

12

       for (int i = 1; i <= n; i++) {

13

           for (int j = 1; j <= i; j++) {

14

               System.out.print(ch + (j == i ? "" : " "));

15

               ch++;

16

          }

17

           System.out.println();

18

      }

19

  }

20

}

21

​

Line: 21 Col: 1

Run Code Submit Code

Upload Code as File

Test against custom input

* [**Testcase 0**](https://www.hackerrank.com/#testcase1)
* [**Testcase 1**](https://www.hackerrank.com/#testcase2)
* [**Testcase 2**](https://www.hackerrank.com/#testcase3)

**Congratulations, you passed the sample test case.**

Click the **Submit Code** button to run your code against all the test cases.

**Input (stdin)**

4

**Your Output (stdout)**

A

B C

D E F

G H I J

**Expected Output**

A

B C

D E F

G H I J

Q3

Write the program to print the pattern

**Input Format**

Input consists of integer.

**Constraints**

No constraints

**Output Format**

Print the pattern

**Sample Input 0**

5

**Sample Output 0**

A

A B

A B C

A B C D

A B C D E

A B C D

A B C

A B

A

**Contest ends in 3 days**

**Submissions:**

[148](https://www.hackerrank.com/contests/looping-statement-java/challenges/alphabet-diamond-1/leaderboard)

**Max Score:**

10

**Difficulty:**

Medium

**Rate This Challenge:**

More

1

import java.io.\*;

2

import java.util.\*;

3

public class Solution {

4

   public static void main(String[] args) {

5

       Scanner sc = new Scanner(System.in);

6

       int n = sc.nextInt();

7

       for (int i = 1; i <= n; i++) {

8

           for (int s = 0; s < n - i; s++) System.out.print(" ");

9

           for (int j = 0; j < i; j++) {

10

               System.out.print((char)('A' + j));

11

               if (j < i - 1) System.out.print(" ");

12

          }

13

           System.out.println();

14

      }

15

       for (int i = n - 1; i >= 1; i--) {

16

           for (int s = 0; s < n - i; s++) System.out.print(" ");

17

           for (int j = 0; j < i; j++) {

18

               System.out.print((char)('A' + j));

19

               if (j < i - 1) System.out.print(" ");

20

          }

21

           System.out.println();

22

      }

23

  }

24

}

25

​

Line: 25 Col: 1

Run Code Submit Code

Upload Code as File

Test against custom input

* [**Testcase 0**](https://www.hackerrank.com/#testcase1)

**Congratulations, you passed the sample test case.**

Click the **Submit Code** button to run your code against all the test cases.

**Input (stdin)**

5

**Your Output (stdout)**

A

A B

A B C

A B C D

A B C D E

A B C D

A B C

A B

A

**Expected Output**

A

A B

A B C

A B C D

A B C D E

A B C D

A B C

A B

A

Q4

write the program to print the pattern

**Input Format**

Input consists of integer.

**Constraints**

No constraints

**Output Format**

Print the following pattern

**Sample Input 0**

5

**Sample Output 0**

\*\*\*\*\*

\* \*

\* \*

\* \*

\*\*\*\*\*

**Sample Input 1**

3

**Sample Output 1**

\*\*\*

\* \*

\*\*\*

**Contest ends in 3 days**

**Submissions:**

[141](https://www.hackerrank.com/contests/looping-statement-java/challenges/hollow-square-pattern-4-1/leaderboard)

**Max Score:**

10

**Difficulty:**

Medium

**Rate This Challenge:**

More

1

import java.io.\*;

2

import java.util.\*;

3

public class Solution {

4

   public static void main(String[] args) {

5

       Scanner sc = new Scanner(System.in);

6

       int n = sc.nextInt();

7

       for (int i = 0; i < n; i++) {

8

           for (int j = 0; j < n; j++) {

9

               if (i == 0 || i == n - 1 || j == 0 || j == n - 1)

10

                   System.out.print("\*");

11

               else

12

                   System.out.print(" ");

13

          }

14

           System.out.println();

15

      }

16

  }

17

}

18

​

19

​

Line: 19 Col: 1

Run Code Submit Code

Upload Code as File

Test against custom input

* [**Testcase 0**](https://www.hackerrank.com/#testcase1)
* [**Testcase 1**](https://www.hackerrank.com/#testcase2)

**Congratulations, you passed the sample test case.**

Click the **Submit Code** button to run your code against all the test cases.

**Input (stdin)**

5

**Your Output (stdout)**

\*\*\*\*\*

\* \*

\* \*

\* \*

\*\*\*\*\*

**Expected Output**

\*\*\*\*\*

\* \*

\* \*

\* \*

\*\*\*\*\*

Q5

Write the program to print the pattern

**Input Format**

Input consists of integer

**Constraints**

No constraints

**Output Format**

Print the pattern

**Sample Input 0**

5

**Sample Output 0**

A

B C

C D E

D E F G

E F G H I

**Contest ends in 3 days**

**Submissions:**

[132](https://www.hackerrank.com/contests/looping-statement-java/challenges/alphabet-right-triangle/leaderboard)

**Max Score:**

10

**Difficulty:**

Medium

**Rate This Challenge:**

More

1

import java.io.\*;

2

import java.util.\*;

3

public class Solution {

4

   public static void main(String[] args) {

5

       Scanner sc = new Scanner(System.in);

6

       int n = sc.nextInt();

7

       for (int i = 0; i < n; i++) {

8

           char ch = (char)('A' + i);

9

           for (int j = 0; j <= i; j++) {

10

               System.out.print(ch + (j < i ? " " : ""));

11

               ch++;

12

          }

13

           System.out.println();

14

      }

15

  }

16

}

Line: 16 Col: 2

Run Code Submit Code

Upload Code as File

Test against custom input

* [**Testcase 0**](https://www.hackerrank.com/#testcase1)

**Congratulations, you passed the sample test case.**

Click the **Submit Code** button to run your code against all the test cases.

**Input (stdin)**

5

**Your Output (stdout)**

A

B C

C D E

D E F G

E F G H I

**Expected Output**

A

B C

C D E

D E F G

E F G H I

Q1

* [**Problem**](https://www.hackerrank.com/contests/basic-statement-java/challenges/time-24)
* [**Submissions**](https://www.hackerrank.com/contests/basic-statement-java/challenges/time-24/submissions)
* [**Leaderboard**](https://www.hackerrank.com/contests/basic-statement-java/challenges/time-24/leaderboard)
* [**Discussions**](https://www.hackerrank.com/contests/basic-statement-java/challenges/time-24/forum)

On one fine Sunday, Sita and Radha decided to solve puzzles. Sita asked Radha to solve her puzzle. Sita gave the puzzle on time. She gave Radha the number of hours, number of minutes and number of seconds as input. With the given input, Radha should find out the total number of hours, the total number of minutes and the total number of seconds. Help Radha by writing a program for this. For example, the input given by Sita is 1, 70, 5. Now, 70 minutes is equal to 1 hour 10 minutes. So the output will be Total number of hour(s) is 2 Total number of minutes is 10 Total number of seconds is 5

**Input Format**

Input consists of three integers. First input corresponds to the number of hours. Second input corresponds to the number of minutes. Third input corresponds to the number of seconds.

**Constraints**

No constraints

**Output Format**

Execute the Time, Minutes and Second

**Sample Input 0**

1

15

10

**Sample Output 0**

Total Number of hours is 1

Total Number of minutes is 15

Total Number of seconds is 10

**Sample Input 1**

5

239

39

**Sample Output 1**

Total Number of hours is 8

Total Number of minutes is 59

Total Number of seconds is 39

**Contest ends in 3 days**

**Submissions:**

[138](https://www.hackerrank.com/contests/basic-statement-java/challenges/time-24/leaderboard)

**Max Score:**

10

**Difficulty:**

Medium

**Rate This Challenge:**

More

1

import java.io.\*;

2

import java.util.\*;

3

public class Solution {

4

   public static void main(String[] args) {

5

       Scanner sc = new Scanner(System.in);

6

       int hours = sc.nextInt();

7

       int minutes = sc.nextInt();

8

       int seconds = sc.nextInt();

9

​

10

       hours += minutes / 60;

11

       minutes = minutes % 60;

12

​

13

       System.out.println("Total Number of hours is " + hours);

14

       System.out.println("Total Number of minutes is " + minutes);

15

       System.out.println("Total Number of seconds is " + seconds);

16

  }

17

}

18

​

Line: 18 Col: 1

Run Code Submit Code

Upload Code as File

Test against custom input

* [**Testcase 0**](https://www.hackerrank.com/#testcase1)
* [**Testcase 1**](https://www.hackerrank.com/#testcase2)

**Congratulations, you passed the sample test case.**

Click the **Submit Code** button to run your code against all the test cases.

**Input (stdin)**

1

15

10

**Your Output (stdout)**

Total Number of hours is 1

Total Number of minutes is 15

Total Number of seconds is 10

**Expected Output**

Total Number of hours is 1

Total Number of minutes is 15

Total Number of seconds is 10

Q1

* [**Problem**](https://www.hackerrank.com/contests/array-statement-java/challenges/richest-customer-wealth-7-1)
* [**Submissions**](https://www.hackerrank.com/contests/array-statement-java/challenges/richest-customer-wealth-7-1/submissions)
* [**Leaderboard**](https://www.hackerrank.com/contests/array-statement-java/challenges/richest-customer-wealth-7-1/leaderboard)
* [**Discussions**](https://www.hackerrank.com/contests/array-statement-java/challenges/richest-customer-wealth-7-1/forum)

You are given an m x n integer grid accounts where accounts[i][j] is the amount of money the ith customer has in the jth bank. Return the wealth that the richest customer has.

A customer's wealth is the amount of money they have in all their bank accounts. The richest customer is the customer that has the maximum wealth.

**Input Format**

Example 1:

Input: accounts = [[1,2,3],[3,2,1]] Output: 6 Explanation: 1st customer has wealth = 1 + 2 + 3 = 6 2nd customer has wealth = 3 + 2 + 1 = 6 Both customers are considered the richest with a wealth of 6 each, so return 6.

Example 2:

Input: accounts = [[1,5],[7,3],[3,5]] Output: 10 Explanation: 1st customer has wealth = 6 2nd customer has wealth = 10 3rd customer has wealth = 8 The 2nd customer is the richest with a wealth of 10.

Example 3:

Input: accounts = [[2,8,7],[7,1,3],[1,9,5]] Output: 17

**Constraints**

m == accounts.length n == accounts[i].length 1 <= m, n <= 50 1 <= accounts[i][j] <= 100

**Output Format**

--

**Sample Input 0**

3 4

1 3 5 2

4 2 6 2

1 4 3 7

**Sample Output 0**

Richest Customer Wealth is 15

**Contest ends in 3 days**

**Submissions:**

[60](https://www.hackerrank.com/contests/array-statement-java/challenges/richest-customer-wealth-7-1/leaderboard)

**Max Score:**

10

**Difficulty:**

Medium

**Rate This Challenge:**

More

1

import java.io.\*;

2

import java.util.\*;

3

public class Solution {

4

   public static void main(String[] args) {

5

       Scanner sc = new Scanner(System.in);

6

       int m = sc.nextInt();

7

       int n = sc.nextInt();

8

       int[][] accounts = new int[m][n];

9

       for (int i = 0; i < m; i++) {

10

           for (int j = 0; j < n; j++) {

11

               accounts[i][j] = sc.nextInt();

12

          }

13

      }

14

       int maxWealth = 0;

15

       for (int i = 0; i < m; i++) {

16

           int sum = 0;

17

           for (int j = 0; j < n; j++) {

18

               sum += accounts[i][j];

19

          }

20

           if (sum > maxWealth) maxWealth = sum;

21

      }

22

       System.out.println("Richest Customer Wealth is " + maxWealth);

23

  }

24

}

25

​

Line: 25 Col: 1

Run Code Submit Code

Upload Code as File

Test against custom input

* [**Testcase 0**](https://www.hackerrank.com/#testcase1)

**Congratulations, you passed the sample test case.**

Click the **Submit Code** button to run your code against all the test cases.

**Input (stdin)**

3 4

1 3 5 2

4 2 6 2

1 4 3 7

**Your Output (stdout)**

Richest Customer Wealth is 15

**Expected Output**

Richest Customer Wealth is 15

Q2

Write a program to execute the transpose of the given 2D matrix.

**Input Format**

first input consists of array size(row and column) second input consists of array elements.

**Constraints**

No Constraints

**Output Format**

Execute the transpose of the given matrix

**Sample Input 0**

3

1

2

3

4

5

6

7

8

9

**Sample Output 0**

Array elements are:

1 2 3

4 5 6

7 8 9

Transpose matrix is:

1 4 7

2 5 8

3 6 9

**Contest ends in 3 days**

**Submissions:**

[61](https://www.hackerrank.com/contests/array-statement-java/challenges/transpose-matrix-24/leaderboard)

**Max Score:**

10

**Difficulty:**

Medium

**Rate This Challenge:**

More

1

import java.io.\*;

2

import java.util.\*;

3

public class Solution {

4

   public static void main(String[] args) {

5

       Scanner sc = new Scanner(System.in);

6

       int n = sc.nextInt();

7

       int[][] matrix = new int[n][n];

8

       for (int i = 0; i < n; i++)

9

           for (int j = 0; j < n; j++)

10

               matrix[i][j] = sc.nextInt();

11

​

12

       System.out.println("Array elements are:");

13

       for (int i = 0; i < n; i++) {

14

           for (int j = 0; j < n; j++)

15

               System.out.print(matrix[i][j] + (j == n - 1 ? "" : " "));

16

           System.out.println();

17

      }

18

​

19

       System.out.println("Transpose matrix is:");

20

       for (int i = 0; i < n; i++) {

21

           for (int j = 0; j < n; j++)

22

               System.out.print(matrix[j][i] + (j == n - 1 ? "" : " "));

23

           System.out.println();

24

      }

25

  }

26

}

27

​

Line: 27 Col: 1

Run Code Submit Code

Upload Code as File

Test against custom input

* [**Testcase 0**](https://www.hackerrank.com/#testcase1)

**Congratulations, you passed the sample test case.**

Click the **Submit Code** button to run your code against all the test cases.

**Input (stdin)**

3

1

2

3

4

5

6

7

8

9

**Your Output (stdout)**

Array elements are:

1 2 3

4 5 6

7 8 9

Transpose matrix is:

1 4 7

2 5 8

3 6 9

**Expected Output**

Array elements are:

1 2 3

4 5 6

7 8 9

Transpose matrix is:

1 4 7

2 5 8

3 6 9

Q3

Write a program to find the element of maximum value in each column.

**Input Format**

The first input corresponds to the number of rows. The second input corresponds to the number of columns. The third input corresponds to the elements of the 2D array.

**Constraints**

No constraints

**Output Format**

print the maximum value of each column.

**Sample Input 0**

3

3

1 2 9

8 3 1

4 1 1

**Sample Output 0**

Maximum value in column 1 is 8

Maximum value in column 2 is 3

Maximum value in column 3 is 9

**Sample Input 1**

3

3

7 8 9

4 5 6

1 2 3

**Sample Output 1**

Maximum value in column 1 is 7

Maximum value in column 2 is 8

Maximum value in column 3 is 9

**Contest ends in 3 days**

**Submissions:**

[89](https://www.hackerrank.com/contests/array-statement-java/challenges/maximum-of-each-column-1/leaderboard)

**Max Score:**

10

**Difficulty:**

Medium

**Rate This Challenge:**

More

1

import java.io.\*;

2

import java.util.\*;

3

public class Solution {

4

   public static void main(String[] args) {

5

       Scanner sc = new Scanner(System.in);

6

       int rows = sc.nextInt();

7

       int cols = sc.nextInt();

8

       int[][] matrix = new int[rows][cols];

9

       for (int i = 0; i < rows; i++)

10

           for (int j = 0; j < cols; j++)

11

               matrix[i][j] = sc.nextInt();

12

​

13

       for (int j = 0; j < cols; j++) {

14

           int max = matrix[0][j];

15

           for (int i = 1; i < rows; i++)

16

               if (matrix[i][j] > max)

17

                   max = matrix[i][j];

18

           System.out.println("Maximum value in column " + (j + 1) + " is " + max);

19

      }

20

  }

21

}

22

​

Line: 22 Col: 1

Run Code Submit Code

Upload Code as File

Test against custom input

* [**Testcase 0**](https://www.hackerrank.com/#testcase1)
* [**Testcase 1**](https://www.hackerrank.com/#testcase2)

**Congratulations, you passed the sample test case.**

Click the **Submit Code** button to run your code against all the test cases.

**Input (stdin)**

3

3

1 2 9

8 3 1

4 1 1

**Your Output (stdout)**

Maximum value in column 1 is 8

Maximum value in column 2 is 3

Maximum value in column 3 is 9

**Expected Output**

Maximum value in column 1 is 8

Maximum value in column 2 is 3

Maximum value in column 3 is 9