/\*

public class Assignment7Q1 {

public static void main(String[] args) {

// Define an array of integers

int[] numbers = {1, 2, 3, 4, 5};

System.out.println("Elements of the array:");

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

System.out.print(numbers[i]+" ");

}

}

}

\*/

/\*

import java.util.Arrays;

public class Q2ArrayCheck {

public static void main(String[] args) {

int[] array1 = {1, 2, 3, 4, 5}, array2 = {1, 2, 3, 5, 4};

System.out.println(checkEquality(array1, array2) ? "Arrays are equal." : "Arrays are not equal.");

}

public static boolean checkEquality(int[] array1, int[] array2) {

return array1.length == array2.length &&

Arrays.equals(Arrays.stream(array1).sorted().toArray(), Arrays.stream(array2).sorted().toArray());

}

}

\*/

/\*

import java.util.Arrays;

public class Q3FindPairs {

public static void main(String[] args) {

findPairsWithSum(new int[]{1, 2, 3, 4, 5, 6}, 7);

}

public static void findPairsWithSum(int[] array, int targetSum) {

Arrays.sort(array);

for (int left = 0, right = array.length - 1; left < right; ) {

int sum = array[left] + array[right];

if (sum == targetSum) System.out.println(array[left++] + ", " + array[right--]);

else if (sum < targetSum) left++;

else right--;

}

}

}

\*/

/\*

import java.util.Arrays;

public class Q4ReverseArray {

public static void main(String[] args) {

int[] array = {1, 2, 3, 4, 5};

System.out.println("Original Array: " + Arrays.toString(array));

for (int i = 0, j = array.length - 1; i < j; i++, j--) {

int temp = array[i];

array[i] = array[j];

array[j] = temp;

}

System.out.println("Reversed Array: " + Arrays.toString(array));

}

}

\*/

/\*

public class Q5Minmax {

public static void main(String[] args) {

int[] array = {3, 6, 2, 8, 1, 9, 4};

int smallest = array[0], largest = array[0];

for (int num : array) {

smallest = Math.min(smallest, num);

largest = Math.max(largest, num);

}

System.out.println("Smallest number: " + smallest);

System.out.println("Largest number: " + largest);

}

}

\*/

/\*

import java.util.Arrays;

public class Q6ThirdLargestNumber {

public static void main(String[] args) {

int[] array = {24,54,31,16,82,45,67};

System.out.println("Third largest number: " + findThirdLargest(array));

}

public static int findThirdLargest(int[] arr) {

return Arrays.stream(arr)

.distinct()

.sorted()

.skip(Math.max(0, arr.length - 3))

.findFirst()

.orElseThrow();

}

}

\*/

/\*

import java.util.Arrays;

public class Q7MergeArrays {

public static void main(String[] args) {

int[] arr1 = {23,60,94,3,102};

int[] arr2 = {42,16,74};

int[] mergedArray = mergeArrays(arr1, arr2);

System.out.println("Merged Array: " + Arrays.toString(mergedArray));

}

public static int[] mergeArrays(int[] arr1, int[] arr2) {

int[] mergedArray = new int[arr1.length + arr2.length];

int i = 0, j = 0, k = 0;

// Merge arrays until one of them is exhausted

while (i < arr1.length && j < arr2.length) {

if (arr1[i] < arr2[j]) {

mergedArray[k++] = arr1[i++];

} else {

mergedArray[k++] = arr2[j++];

}

}

// Copy remaining elements from the first array, if any

while (i < arr1.length) {

mergedArray[k++] = arr1[i++];

}

// Copy remaining elements from the second array, if any

while (j < arr2.length) {

mergedArray[k++] = arr2[j++];

}

return mergedArray;

}

}

\*/

/\*

public class Q8RunningAverage {

public static void main(String[] args) {

int[] numbers = {5,14,35,89,140}; // Example array

for (int i = 0; i <= numbers.length - 3; i++)

System.out.print((int)(numbers[i] + numbers[i + 1] + numbers[i + 2]) / 3 +" ");

}

}

\*/

/\*

public class Q9SeriesGenerator {

public static void main(String[] args) {

for (int i = 1; i <= 6; i++) {

System.out.print((i % 2 == 0) ? i \* i \* i : i \* i);

if (i < 6) System.out.print(",");

}

}

}

\*/

/\*

ppublic class Q10OrderChecker {

public static void main(String[] args) {

int[] arr = {65,14,129,34,7};

boolean ascending = true;

boolean descending = true;

for (int i = 1; i < arr.length; i++) {

if (arr[i] < arr[i - 1]) ascending = false;

if (arr[i] > arr[i - 1]) descending = false;

}

if (ascending && !descending) System.out.println("Ascending");

else if (descending && !ascending) System.out.println("Descending");

else System.out.println("Random");

}

}

\*/