CDAC-Mumbai

# Subject: Coding Challenge

**Date:18-09-2024 Total Marks:5**

# Time duration: 1:30hrs. Batch: Aug-24

**----------------------------------------------------------------------------------------------------------------**

1. Write a Java program to prove that Euclid’s algorithm computes the greatest common divisor of two positive given integers. **[1 Marks]**

*"The Euclidean algorithm is based on the principle that the greatest common divisor of two numbers does not change if the larger number is replaced by its difference with the smaller number. For example, 21 is the GCD of 252 and 105 (as 252 = 21 × 12 and 105 = 21 × 5), and the same number 21 is also the GCD of 105 and 252 − 105 = 147. Since this replacement reduces the larger of the two numbers, repeating this process gives successively smaller pairs of numbers until the two numbers become equal. When that occurs, they are the GCD of the original two numbers. By reversing the steps, the GCD can be expressed as a sum of the two original numbers each multiplied by a positive or negative integer, e.g., 21 = 5 × 105 + (−2) × 252. The fact that the GCD can always be expressed in this way is known as Bézout's identity."*

**import** java.util.Scanner;

**public** **class** GCD {

**public** **static** **void** main(String[] args) {

**int** gcd = 0;

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Enter the first number");

**int** n1 = sc.nextInt();

**int** arr1[] = **new** **int**[n1+1];

**for**(**int** i=2;i<=n1;i++) {

**if**(n1%i==0) {

arr1[i] = i;

System.***out***.println(arr1[i]);

}

}

System.***out***.println("Enter the second number");

**int** n2 = sc.nextInt();

**int** arr2[] = **new** **int**[n2+1];

**for**(**int** i=2;i<=n2;i++) {

**if**(n2%i==0) {

arr2[i] = i;

System.***out***.println(arr2[i]);

}

}

**if**(n1>n2) {

**for**(**int** i=0;i<n1;i++) {

**for**(**int** j=0;j<n2;j++) {

**if**(arr1[i] == arr2[j] && arr1[i]!=0 && arr2[j]!=0) {

gcd = arr2[j];

}

}

}

}

**else** {

**for**(**int** i=0;i<n2;i++) {

**for**(**int** j=0;j<n1;j++) {

**if**(arr1[j] == arr2[i] && arr1[j]!=0 && arr2[i]!=0) {

gcd = arr2[i];

}

}

}

}

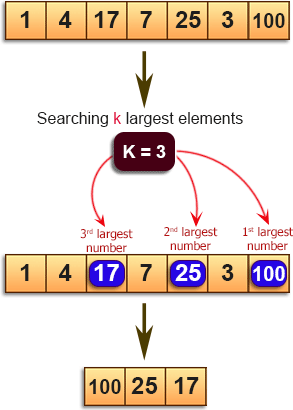
System.***out***.println("GCD of "+n1+" and " + n2 + " is "+ gcd);

sc.close();}

}

1. Write a Java program to find the k largest elements in a given array. Elements in the array can be in any order (Use scanner class to input and Array, below picture is example do not make static array as repsrented in example). **[1 Mark]**

# Pictorial Presentation:



import java.util.Scanner;

import java.util.Arrays;

public class KLarEle {

public static void main(String[] args) {

Scanner sc = new Scanner(System.in);

System.out.println("Enter the size of array");

int n = sc.nextInt();

int arr[] = new int[n];

System.out.println("Enter the elements of array");

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

arr[i] = sc.nextInt();

}

Arrays.sort(arr);

System.out.println("Enter the number of elements to be searched");

int k = sc.nextInt();

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

System.out.println(arr[i]);

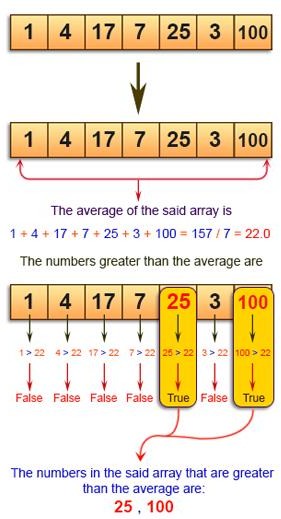
}

sc.close();}

}

1. Write a Java program to find the numbers greater than the average of the numbers of a given array (Use scanner class to input and Array, below picture is example do not make static array as repsrented in example). **[1 mark]**

# Pictorial Presentation:



**import** java.util.Scanner;

**public** **class** GrThAv {

**public** **static** **void** main(String[] args) {

**int** sum=0;

**float** avg=0.0f;

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Enter the size of array");

**int** n = sc.nextInt();

**int** arr[] = **new** **int**[n];

System.***out***.println("Enter the elements of array");

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

arr[i] = sc.nextInt();

}

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

sum+=arr[i];

}

avg = sum/n;

System.***out***.print("The numbers in the said array that are greater than the average are: ");

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

**if**(arr[i]>avg) {

System.***out***.print(arr[i]+",");

}

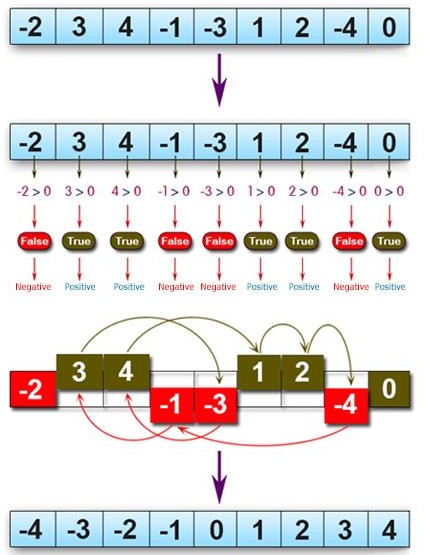
}

sc.close();}

}

Q4. Write a Java program to move every positive number to the right and every negative number to the left of a given array of integers (Use scanner class to input and Array, below picture is example do not make static array as repsrented in example). **[1 Mark]**

# Pictorial Presentation:



import java.util.Scanner;

public class Arr\_Int {

public static void main(String[] args) {

int temp;

Scanner sc = new Scanner(System.in);

System.out.println("Enter the size of array");

int n = sc.nextInt();

int arr[] = new int[n];

System.out.println("Enter the elements of array");

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

arr[i] = sc.nextInt();

}

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

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

if(arr[i]>arr[j]) {

temp=arr[i];

arr[i]=arr[j];

arr[j]=temp;

}

}

}

sc.close();

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

System.out.println(arr[i]);

}

}

}

5. Write a Java program to find the median of the number inside the window (size k) at each moving in a given array of integers with duplicate numbers. Move the window from the start of the array. **[1 Mark]**

Example:

{|1, 2, 3|, 4, 5, 6, 7, 8, 8} -> Return median 2

{1, |2, 3, 4|, 5, 6, 7, 8, 8} -> Return median 3

{1, 2, |3, 4, 5|, 6, 7, 8, 8} -> Return median 4

{1, 2, 3, |4, 5, 6|, 7, 8, 8} -> Return median 5

{1, 2, 3, 4, |5, 6, 7|, 8, 8} -> Return median 6

{1, 2, 3, 4, 5, |6, 7, 8|, 8} -> Return median 7

{1, 2, 3, 4, 5, 6, |7, 8, 8|} -> Return median 8

Result array {2, 3, 4, 5, 6, 7, 8}

Note: Above representation is just example, use scanner class for input the array and for getting size of window to slide.

**import** java.util.Scanner;

**import** java.lang.Math;

**import** java.util.Arrays;

**public** **class** Median {

**public** **static** **void** main(String[] args) {

// **TODO** Auto-generated method stub

Scanner sc = **new** Scanner(System.***in***);

System.***out***.println("Enter the size of array");

**int** n = sc.nextInt();

**int** arr[] = **new** **int**[n];

System.***out***.println("Enter the elements of array");

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

arr[i] = sc.nextInt();

}

Arrays.*sort*(arr);

System.***out***.println("Enter the window size of array");

**int** k = sc.nextInt();

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

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

**if**(k+i<=n) {

**int** a = arr[i];

System.***out***.println("arr["+i+"] ="+ arr[i]);

**int** b = arr[k+i-1];

System.***out***.println("arr["+(k+i-1)+"] ="+ arr[k+i-1]);

**double** c = Math.*ceil*(((**double**)a+(**double**)b)/2.0);

System.***out***.println((**int**)c);

**break**;

}

}

}

sc.close();}

}