NAME : WIJAYAWARDHANA W.A.H.A.

REGISTRATION NO. : 2019/E/166

SEMESTER : SEMESTER 04

DATE ASSIGNED : 15 MARCH 2022

LINEAR ABSTRACT DATA TYPES – LAB 04

EC 4070

DATA STRUCTURES AND ALGORITHMS

01.

Code:-

public class PriorityQueue {

int n;

int k;

int zeroIndex;

int maximumSum;

int[] arrayA = new int[n];

public PriorityQueue() // Running default constructor.

{

maximumSum =0;

n = 0;

k = 0;

}

public void setElement(int[] arrayA , int n) // Set element method.

{

this.arrayA = arrayA; // Setting array.

this.n = n; // Setting n value.

}

public void findModifyingTime() // Find the number of modifying time.

{

int i = 0;

while ((arrayA[i] >= 0))

{

i++;

}

k = arrayA[i] \* -2;

findZeroElement();

}

public void modifyArray(int modifyingIndex) // Modify the array elements.

{

while ((k > 0)&&(modifyingIndex<n)) // While k>0 and until end of the array both condition true loop will run.

{

if(arrayA[modifyingIndex] < 0) // If element found less than 0 that will modify and added to maximum sum.

{

maximumSum = maximumSum + (arrayA[modifyingIndex] \* -1); // Modify element and added to the sum.

k--;

modifyArray(modifyingIndex+1); // Recursive the method.

}

modifyingIndex++; // Increment the array index.

}

while ((k > 0)) // After modifying done for 0 less values zero will have the change.

{

k--;

arrayA[zeroIndex] = arrayA[zeroIndex] \* -1;

}

}

public void SortingArray() // PrintElement method is for sorting and printing the array.

{

for(int i = 0; i<n; i++) // Sorting the array.

{

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

{

if(arrayA[i] > arrayA[j])

{

int temp = arrayA[i];

arrayA[i] = arrayA[j];

arrayA[j] = temp;

}

}

}

for(int k = 0; k<n; k++) // Printing the array.

{

System.out.println(arrayA[k]);

}

}

public void findZeroElement() // Find the zero element.

{

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

{

if(arrayA[i] == 0)

{

zeroIndex = i;

}

}

}

public void getMaximumSum() // Calculate the sum of array.

{

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

{

if(arrayA[j] > 0) // 0 less value will not consider already added to this.

{

maximumSum = maximumSum+arrayA[j]; // Adding element values.

}

}

System.out.println("Maximum sum : " + maximumSum); // Print the maximum value.

}

public static void main(String[] args) {

int[] array = new int[] {-2,0,5,-1,2,-5,8}; // Define an array.

PriorityQueue newObject = new PriorityQueue(); // Creating object of the class.

newObject.setElement(array, array.length); // Calling set element method.

newObject.SortingArray(); // Calling sortingArray method.

newObject.findModifyingTime(); // Calling findModifyingTime method.

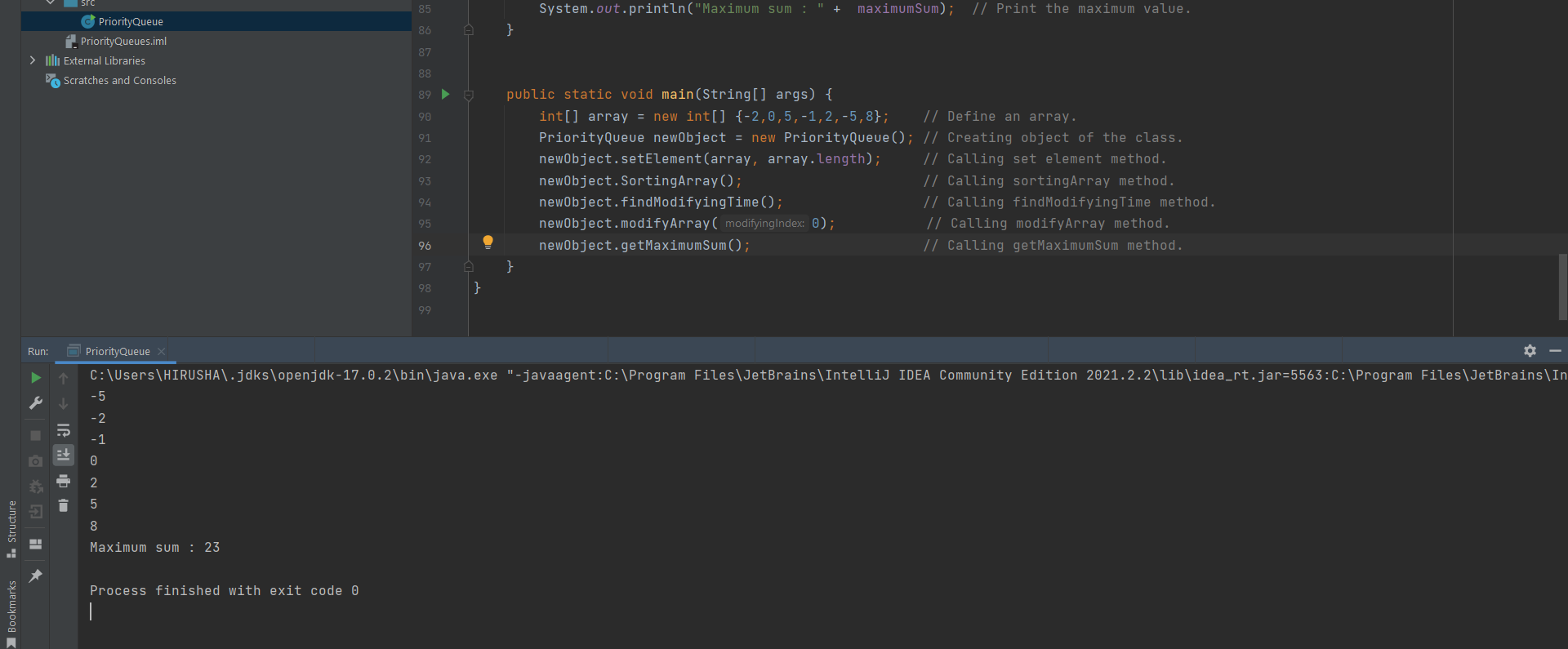
newObject.modifyArray(0); // Calling modifyArray method.

newObject.getMaximumSum(); // Calling getMaximumSum method.

}

}

Outputs:-



02.

Code:-

import java.util.Scanner;

public class CircularLinkedList {

Scanner scanner = new Scanner(System.in);

int numberOfPeople;

int numberCountingOff;

int n = 0;

int[] arrayElement = new int[numberOfPeople];

int[] temporaryArray = new int[n];

public CircularLinkedList()

{

numberOfPeople = 0;

numberCountingOff = 0;

}

public void setElement()

{

System.out.println("Enter the number of people in the circle (n) : ");

numberOfPeople = scanner.nextInt();

System.out.println("Enter the number used for counting off (m) :");

numberCountingOff = scanner.nextInt();

int[] buildArray = new int[numberOfPeople];

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

{

buildArray[i] = i+1;

}

arrayElement = buildArray;

n = numberOfPeople;

temporaryArray = buildArray;

}

public void committedSuicide()

{

for(int i = 0; i <numberOfPeople; i+=numberCountingOff) // i <numberOfPeople-1

{

if((i+numberCountingOff-1) < numberOfPeople)

{

System.out.print(temporaryArray[i+numberCountingOff-1] + " ");

//arrayElement[i+numberCountingOff-1] = 0;

arrayElement[temporaryArray[i+numberCountingOff-2]] = 0;

n++;

}

}

System.out.println();

int k = 0;

for(int j = 0; j < (numberOfPeople-1); j++)

{

if(arrayElement[j] != 0)

{

System.out.print(arrayElement[j] + " ");

temporaryArray[k] = arrayElement[j];

k++;

}

}

System.out.println();

if(temporaryArray.length != 1)

{

committedSuicide();

}

else

{

System.out.println(temporaryArray[0]);

}

}

public void countOffAroundCircle(int startingIndex)

{

for(int i =0; i<numberOfPeople-1;i++)

{

}

}

public static void main(String[] args) {

CircularLinkedList newObject = new CircularLinkedList();

newObject.setElement();

newObject.committedSuicide();

}

}