# **Homework**

# **Grade 11 Review 3 – Arrays**

1. Write a program to perform each task (in separate loops):
   1. Ask user to enter an integer *n*, then declare an array of integer of size *n*
   2. Initialize all elements of the array to one
   3. Ask user to enter *n* integers, and fill the arrays with these integer in order
   4. Switch the values at either end of the array
   5. Change any negative values to positive values (of the same magnitude)
   6. Set the variable sampleSum to the sum of the values of all the elements
   7. Print all the even numbers in the array

| /\*  \* Program name: Gr11Review\_Array\_1\_a.java  \*  \* By: Lucas Chow (Last edited: 2022-09-19)  \*  \* ICS4U1 - 03\_Gr11Review  \*  \* This program starts by prompting the user for an integer n.  \* (if n is negative, it turns it into a positive)  \* > the program then initializes an integer array length n  \* > The program prompts for n integers for the user to enter  \* > The integer elements are ordered from least to greatest  \* > elements at either ends of the arrays are switched  \* > All negative elements switched to positive numbers of the same magnitude  \* > Prints sampleSum, the sum of all elements in the array  \* > Prints all even numbers in the array  \*  \*/  //importing java.io for the bufferedReader scanner  import java.io.\*;  public class Gr11Review\_Arrays\_1 {      /\*  \* void bubbleSort(int inputArr[])  \*  \* inputArr[] -> the inputted array, which the bubble sort is returned:  \*  \* void; since arrays are pass-by objects, returning the array would be redundant  \*  \* this method uses the bubblesort algorithm; checking two adjacent consecutive  \* elements, and comparing them. If the earlier element is larger than the following  \* element, the integers swap.  \*  \*  \*/  public static void bubbleSort(int inputArr[])  {  int n =inputArr.length;  for (int i = 0; i < n - 1; i++)  for (int j = 0; j < n - i - 1; j++)  if (inputArr[j] >inputArr[j + 1]) {  int temp =inputArr[j];  inputArr[j] =inputArr[j + 1];  inputArr[j + 1] = temp;  }  }  /\*  \* boolean isInt(String input)  \*  \* String input -> the input as a string for inclusivity.  \*  \* This method checks if the string input can be parsed to an integer,  \* otherwise catching an error. If it CAN be parsed to an integer,  \* implying it is an integer, returns the boolean isInt, which will be true;  \* Otherwise, returns false  \*  \*  \*/  public static boolean isInt(String input)  {  boolean isInt = false;  try  {  Integer.parseInt(input);  isInt = true;  }  catch (NumberFormatException e)  {}    return isInt;    }    //start of the main method  public static void main(String[] args)  {    try  {  //creating bufferedReader  BufferedReader sc = new BufferedReader(new InputStreamReader(System.in));  String input;  System.out.print("Enter an integer: ");  int sampleSum;  int lengthN;  int[] intArray;  boolean isInt;  int temp;    //declaring lengthN to 0  lengthN = 0;  //declaring sampleSum to 0  sampleSum = 0;      //this do-while loop ensures the user's prompts are a positive integer, otherwise prompting indefinitely  do {  isInt = false;  input = sc.readLine();  if (!isInt(input))  {  System.out.println("Make sure input is an integer");  }  else  {  if (Integer.parseInt(input) <= 0)  {  System.out.print("Please make sure the input is a positive integer: ");  }  else  {  lengthN = Integer.parseInt(input);  isInt = true;  }  }  } while (!isInt);  intArray = new int[lengthN];    //initializing all elements in the array to 1  for (int i = 0; i < intArray.length; i++)  {  intArray[i] = 1;  }  System.out.println("Enter "+lengthN+ " integer: ");  for (int i = 0; i < lengthN; i++)  {  System.out.print("Integer #"+(i+1)+": ");  do  {  input = sc.readLine();  if (!isInt(input))  {  System.out.println("Please make sure the input is an integer");  }  else  {  intArray[i] = Integer.parseInt(input);  }  } while(!isInt(input));  }      //bubble sort method, see above. Uses the bubble sort algorithm  bubbleSort(intArray);      //swapping the first and last elements in the array  temp = intArray[0];  intArray[0] = intArray[intArray.length-1];  intArray[intArray.length-1] = temp;      //ensuring all values in the array are positive, changing negative numbers to positive numbers  for (int i = 0; i < lengthN; i++)  {  if (intArray[i] < 0)  {  intArray[i] = -intArray[i];  }  sampleSum += intArray[i];  }      //printing sampleSum  System.out.println("The sample sum, or the sum of all elements in the array is: "+sampleSum);      //printing all even numbers in the array  System.out.println("Printing all the even numbers in the array");  for (int i = 0; i < lengthN; i++)  {  if (intArray[i]%2 == 0)  {  System.out.println("Element #" + (i+1) +": "+intArray[i]);  }  }    //printing out values in the array  System.out.println("Values in the Array: ");  for(int i = 0; i < lengthN; i++)  {  System.out.print(intArray[i]+" ");  }    //closing bufferedreader  sc.close();  }    //catching IOExceptions when using Buffered Reader  catch (IOException e)  {  //printing out the error  System.out.println("Error "+e);  }  }  } |
| --- |
|  |

1. Write a program that initialized an array of size 14 with user input and then reverses the order of the values in the array using the specified methods:
   1. The first version uses two arrays. The original array is not changed. The second array gets the elements of the first array in reversed order. The content of the second array is then printed in order.
   2. In the second version, there is only one array and its values are reversed and printed.

a)

| /\*  \* Program name: Gr11Review\_Array\_2\_a.java  \*  \* By: Lucas Chow (Last edited: 2022-09-21)  \*  \* ICS4U1 - 03\_Gr11Review  \*  \* This program prompts the user to fill in an array length 14,  \* then reverses the array using a second array  \*  \*  \*/  //importing scanner  import java.util.\*;  public class Gr11Review\_Array\_2\_a {  public static void main(String[] args)  {  //creating scanner and variables  Scanner sc = new Scanner(System.in);  final int ARRAY\_LENGTH = 14;  int[] intArray1 = new int[ARRAY\_LENGTH];  int[] intArray2 = new int[ARRAY\_LENGTH];      //looping through the array and prompting the user to enter the values  for (int i = 0; i < ARRAY\_LENGTH; i++)  {  System.out.print("Enter value #"+(i+1)+": ");  intArray1[i] = Integer.parseInt(sc.nextLine());  }    //reversing the array using another array    for (int i = 0; i < ARRAY\_LENGTH; i++)  {    //copying the elements into intArray2 in the opposite order  intArray2[ARRAY\_LENGTH-1-i] = intArray1[i];  }    //printing the first and second array  System.out.println("Element - Arr1 - Arr2");  for (int i = 0; i < ARRAY\_LENGTH; i++)  {  System.out.printf("#%-3d 1:%-3d 2:%-3d \n",(i+1),(intArray1[i]), (intArray2[i]));  }    //closing scanner  sc.close();  }  } |
| --- |
|  |

b)

| /\*  \* Program name: Gr11Review\_Array\_2\_b.java  \*  \* By: Lucas Chow (Last edited: 2022-09-21)  \*  \* ICS4U1 - 03\_Gr11Review  \*  \* This program prompts the user to fill in an array length 14,  \* then reverses the array using the same array  \*  \*  \*/  //importing scanner  import java.util.\*;  public class Gr11Review\_Array\_2\_b {  public static void main(String[] args)  {  //creating scanner  Scanner sc = new Scanner(System.in);  final int ARRAY\_LENGTH = 14;  int[] intArray = new int[ARRAY\_LENGTH];  int temp;    for (int i = 0; i < ARRAY\_LENGTH; i++)  {  System.out.print("Enter value #"+(i+1)+": ");  intArray[i] = Integer.parseInt(sc.nextLine());  }    //reversing the array in the same array by swapping opposite values    for (int i = 0; i < ARRAY\_LENGTH; i++)  {  temp = intArray[i];  intArray[i] = intArray[ARRAY\_LENGTH-1-i];  intArray[ARRAY\_LENGTH-1-i] = temp;  }    //printing the array  System.out.println("Element - Arr1");  for (int i = 0; i < ARRAY\_LENGTH; i++)  {  System.out.printf("#%-3d 1:%-3d\n",(i+1),(intArray[i]));  }    //closing scanner  sc.close();  }  } |
| --- |
|  |

1. Assume the array data has been initialized as followed:

int[][] data = { {3, 2, 5},

{1, 4, 4, 8, 13},

{9, 1, 0, 2},

{0, 2, 6, 4, -1, -8}};

Write a program that computes the sum of all the elements of the array. Your program should work even if the dimensions of the rows and columns are changed. So use length rather than hard-coded numbers.

| /\*  \* Program name: SumOfElementsInArray.java  \*  \* By: Lucas Chow (Last edited: 2022-09-21)  \*  \* ICS4U1 - 03\_Gr11Review  \*  \* This program calculates the sum of all the values in the given 2d array  \*  \*  \*/  public class SumOfElementsInArray {    //start of the main method  public static void main(String[] args)  {  //declaring variables  int sum;      //data input  int[][] data = {{3, 2, 5},  {1, 4, 4, 8, 13},  {9, 1, 0, 2},  {0, 2, 6, 4, -1, -8}};    //printing out the data input to the console  System.out.println("\n"  + "int[][] data = {{3, 2, 5},\r\n"  + " {1, 4, 4, 8, 13},\r\n"  + " {9, 1, 0, 2},\r\n"  + " {0, 2, 6, 4, -1, -8}};");    //initializing sum = 0  sum = 0;    //calculating the sum of the data  for (int i = 0; i < data.length; i++)  {  for (int a = 0; a < data[i].length; a++)  {  sum += data[i][a];  }  }    //printing out the sum  System.out.println("\nThe sum of all the elements of the 2d array is "+sum);    }  } |
| --- |
|  |

1. Write a program that uses a 2D array to represents the marks of a set of tests for a class of students. Your program should perform the following tasks
   1. Ask user to enter the number of students and the number of tests
   2. Ask user to enter each of the test marks for all students
   3. Calculate and output the average of each student
   4. Output the student (represented by the number) with the highest average
   5. List the students whose average is higher than the class average

| /\*  \* Program name: ClassMarks.java  \*  \* By: Lucas Chow (Last edited: 2022-09-21)  \*  \* ICS4U1 - 03\_Gr11Review  \*  \* This program prompts the user for number of test and students, prompting them to  \* enter all the marks. The program then calculates the average for each individual student,  \* the student with the highest average, and the student(s) with averages higher than the  \* class average  \*  \*  \*/  //importing scanner  import java.util.\*;  public class ClassMarks {  public static void main(String[] args)  {  //initializing variables  Scanner sc = new Scanner(System.in);  int numberOfStudents;  int numberOfTest;  int studentAverage;  int classAverage;  int highestMark;  int indexOfHighestMark;  int[][] classMarks;  int[] studentsAverage;      //prompting for number of students and test  System.out.print("Please enter the number of students: ");  numberOfStudents=sc.nextInt();  sc.nextLine();    System.out.print("Please enter the number of test: ");  numberOfTest=sc.nextInt();  sc.nextLine();    //initializing arrays  classMarks = new int[numberOfStudents][numberOfTest];  studentsAverage = new int[numberOfStudents];      //prompting for the marks  for (int i = 0; i < numberOfStudents; i++)  {  for (int a = 0; a < numberOfTest; a++)  {  System.out.printf("What is the mark for student #%d test #%d:",(i+1),(a+1));  classMarks[i][a] = sc.nextInt();  sc.nextLine();  }  }      //outputing the averages of each student  for (int i = 0; i < numberOfStudents; i++)  {  studentAverage = 0;  for (int a = 0; a < numberOfTest; a++)  {  studentAverage += classMarks[i][a];  }  studentAverage /= numberOfTest;    System.out.printf("The average for student #%d is %d\n",(i+1),(studentAverage));  studentsAverage[i] = studentAverage;  }    //finding the highest average of students  indexOfHighestMark = 0;  highestMark = 0;  for (int i = 0; i < numberOfStudents; i++)  {  if (studentsAverage[i] > highestMark)  {  highestMark = studentsAverage[i];  indexOfHighestMark = i;  }  }    System.out.println("The highest average is " + studentsAverage[indexOfHighestMark]+" by student "+(indexOfHighestMark+1));    //calculating and outputting students with averages higher than the class average  classAverage = 0;  for (int i = 0; i < numberOfStudents; i++)  {  classAverage += studentsAverage[i];  }    classAverage /= numberOfStudents;    for (int i = 0; i < numberOfStudents; i++)  {  if (studentsAverage[i] > classAverage)  {  System.out.println("Student #"+(i+1)+" has an average higher than the class average");  }  }      //closing scanner  sc.close();    }  } |
| --- |
|  |