## East West University

**Department of Computer Science and Engineering**

## A/2, Jahurul Islam Avenue, Jahurul Islam City, Aftabnagar, Dhaka

**Lab Manual:** 03

**Lab Topic:** Loops and Basic Array Manipulations

**Course Code:** CSE110 (Object Oriented Programming)

**Course Instructor:** Tanni Mittra, Senior Lecturer, CSE

**Lab Objective**

1. **Apply** concepts of loops and arrays
2. **Write** and **execute** programs using these concepts in Java.

**Lab Activities**

## ****Arrays in Java****

* Arrays are declared in the following manner in Java.

int[] numbers = new int[10];

* The above-mentioned line created an integer array numbers of size 10.
* The array elements can be accessed in the same way as in C.
* Examine the following program.

import java.lang.\*;

import java.util.\*;

class SampleArray{

public static void main(String[] args){

// initializing an integer array numbers with size 10

int[] numbers = new int [10];

Scanner input = new Scanner (System.in);

// assigning values randomly

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

numbers[i] = input.nextInt();

// numbers[i] = (int)(Math.random()\*100);// Read random numbers

}

// displaying the values

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

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

}

}

}

* **numbers.length** returns the size of the array.
* For Java String, you can use **chatAt(i)** method to access the character at ithposition of the string. Suppose,

String str = “Hello World”;

System.out.print(str.charAt(0)); // returns the character ‘H’

## ****Enhanced For loop****

## Java also includes another version of for loop

## Enhanced for loop provides a simpler way to iterate through the elements of a collection or array

## It is read only loop where you can’t update the values as opposite to other loops

|  |
| --- |
| public class enhancedforloop  {      public static void main(String args[])      {          int array[] = {10, 20, 30)            //enhanced for loop          for (int x:array)          {              System.out.println(x);          }            /\* for loop for same function          for (int i = 0; i < array.length; i++)          {              System.out.println(array[i]);          }          \*/      }  } |

**Lab Problems**

**01:** Write a Java program that reads student scores, gets the best score, and then assigns grades based on the following scheme:

Grade is A if score is ≥ best - 10

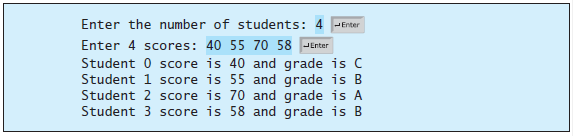
Grade is B if score is ≥ best - 20;

Grade is C if score is ≥ best - 30;

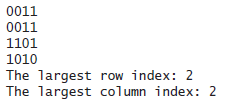
Grade is D if score is ≥ best - 40;

Grade is F otherwise.

The program prompts the user to enter the total number of students, then prompts the user to enter all of the scores and concludes by displaying the grades. Here is a sample run:



**02:** Write a program that randomly fills in 0s and 1s into a 4-by-4 matrix, prints the matrix, and finds the first row and column with the most 1s. Here is a sample run of the program:



**03:** A *Palindrome* is a number or a string that reads the same from either way (forward or backward). As an example, RADAR is a palindrome, but ROVER is not. Write a Java program that prompts the user to enter a string and displays whether the string is a palindrome or not. You may not use any built-in Java methods to accomplish that.

**04:** Write a Java program that reads in ten numbers and displays the number of distinct numbers and the distinct numbers separated by exactly one space (i.e., if a number appears multiple times, it is displayed only once). (*Hint*: Read a number and store it to an array if it is new. If the number is already in the array, ignore it.) After the input, the array contains the distinct numbers. Write a method Isdistinct() to check whether a number exists multiple times or not.

**05:** Write a program to sort a two-dimensional array according to the values in any given column

Input: If our 2D array is given as (Order 4X4)

39 27 11 42

10 93 91 90

54 78 56 89

24 64 20 65

Sorting it by values in column 3

Output: 39 27 11 42

24 64 20 65

54 78 56 89

10 93 91 90