Programming Paradigms Laboratory

B.Tech.

Name : KAUSHAL VASHISTH

Roll Number : 18ETCS002147

Department : Computer Science and Engineering

Faculty of Engineering & Technology
Ramaiah University of Applied Sciences

Faculty	Engineering & Technology
Programme	B. Tech. in Computer Science and Engineering
Year/Semester	2nd Year / 4th Semester
Name of the Laboratory	Programming Paradigms Laboratory
Laboratory Code	19CSL217A

Laboratory 3

Title of the Laboratory Exercise: One dimensional arrays

1. Questions

    a. Develop a Java program to delete the duplicate elements from an array.
    b. Develop a GradeBook class with an instance variables string course name and array of grades that instructors can use to maintain students' grades on an exam and display a grade report that includes the grades, class average, lowest grade and highest grade.

2. Calculations/Computations/Algorithms

Part-

```java
package duplicate_element;
public class Duplicate_element {

public static void remove_dupli(int a[]){
        System.out.println("initial array: ");

        for(int i = 0; i < a.length; i++){
            System.out.println(a[i] + "\t");
        }

        int n = a.length;
        for(int i = 0; i < n; i++){
            for(int j = i+1; j < n; j++){
                if(a[i] == a[j]){
                    a[j] = a[n - 1];
                    n--;
                    j--;
                }
            }
        }

        System.out.println();
        System.out.println("Array with deleted duplicate values: ");

        for(int i = 0; i < n; i++){
            System.out.println(a[i] + "\t");
        }
    }

    public static void main(String[] args){
        remove_dupli(new int[]{1,2,6,8,6,2,8,9,9});
    }

}
```

Part-b:-

```java
package lab3_2;
class gradebook {

    String coursename;
    int m_array[] = {};
    gradebook(String cn, int a[]) {
        this.coursename = cn;
        this.m_array = a;
    }

    char grade[] = new char[20];
    int [] y = new int[20];

    public int gethighest() {
        int high = m_array[0];
        for (int i = 0; i < m_array.length; i++) {
            if (m_array[i] > high) {
                high = m_array[i];
            }
        }
        return high;
    }

    public int getlowest() {
        int low = m_array[0];
        for (int i = 0; i < m_array.length; i++) {
            if (m_array[i] < low) {
                low = m_array[i];
            }
        }
        return low;
    }

    public int getaverage() {
        int sum = 0;
        for (int i = 0; i < m_array.length; i++) {
            sum += m_array[i];
        }
        return sum / m_array.length;
    }
```

```java
public void setgrade() {
    for (int i = 0; i < m_array.length; i++) {
        if (m_array[i] >= 0 && m_array[i] <= 9){
            grade[i] = 'F';y[0]+=1;
        }
        if (m_array[i] >= 10 && m_array[i] <= 19){
            grade[i] = 'F';y[1]+=1;
        }
        if (m_array[i] >= 20 && m_array[i] <= 29){
            grade[i] = 'F';y[2]+=1;
        }
        if (m_array[i] >= 30 && m_array[i] <= 39){
            grade[i] = 'F';y[3]+=1;
        }
        if (m_array[i] >= 40 && m_array[i] <= 49){
            grade[i] = 'F';y[4]+=1;
        }
        if (m_array[i] >= 50 && m_array[i] <= 59){
            grade[i] = 'E';y[5]+=1;
        }
        if (m_array[i] >= 60 && m_array[i] <= 69){
            grade[i] = 'D';y[6]+=1;
        }
        if (m_array[i] >= 70 && m_array[i] <= 79){
            grade[i] = 'C';y[7]+=1;
        }
        if (m_array[i] >= 80 && m_array[i] <= 89){
            grade[i] = 'B';y[8]+=1;
        }
        if (m_array[i] >= 90 && m_array[i] <= 99){
            grade[i] = 'A';y[9]+=1;
        }
        if (m_array[i] == 100){
            grade[i] = 'O';y[10]+=1;
        }
    }
}
void display2(){
    System.out.println("marks array is:-");
    for (int i = 0; i < m_array.length; i++) {
        System.out.printf("%d ",m_array[i]);
    }
    System.out.println();
    System.out.println("highest marks = "+ gethighest());
    System.out.println("lowest marks = "+ getlowest());
    System.out.println("average marks = "+ getaverage());
}
void display() {
    System.out.println();
    System.out.printf("Report for %s is :- \n", coursename);
    System.out.println("---------------------------");
    for (int i = 0; i < m_array.length; i++) {
        System.out.printf("student[%d]:- marks=%d grade=%c \n"
, i, m_array[i],grade[i]);
    }
}
```

```java
    void freq_display(){
        System.out.println("\n
star represents no. of students in given range:-\n");
        System.out.printf("marks\t\t\tfrequecy \n");
        for(int i=0;i<11;i++){
            int r=i*10,r2=r+9;
            System.out.printf("btw %d & %d\t\t",r,r2);
            for(int j=0;j<y[i];j++)System.out.printf("*");
            System.out.println();
        }
    }

}

public class Lab3_2 {

    public static void main(String[] args) {
        String coursename = "cse";
        //array of marks
        int a[] = {18, 78, 79, 90, 56, 58, 90, 99, 76, 100, 87};
        gradebook gb = new gradebook(coursename, a);
        gb.display2();
        gb.setgrade();
        gb.display();
        gb.freq_display();
    }
}
```

3. Presentation of Results

Answer Part-a:-

```
run:
initial array:
1
2
6
8
6
2
8
9
9

Array with deleted duplicate values:
1
2
6
8
9
BUILD SUCCESSFUL (total time: 0 seconds)
```

Answer Part b:-

```
run:
marks array is:-
18 78 79 90 56 58 90 99 76 100 87
highest marks = 100
lowest marks = 18
average marks = 75

Report for cse is :-
----------------------------
student[0]:- marks=18 grade=F
student[1]:- marks=78 grade=C
student[2]:- marks=79 grade=C
student[3]:- marks=90 grade=A
student[4]:- marks=56 grade=E
student[5]:- marks=58 grade=E
student[6]:- marks=90 grade=A
student[7]:- marks=99 grade=A
student[8]:- marks=76 grade=C
student[9]:- marks=100 grade=O
student[10]:- marks=87 grade=B

star represents no. of students in given range:-

marks           frequecy
btw 0 & 9
btw 10 & 19     *
btw 20 & 29
btw 30 & 39
btw 40 & 49
btw 50 & 59     **
btw 60 & 69
btw 70 & 79     ***
btw 80 & 89     *
btw 90 & 99     ***
btw 100 & 109      *
BUILD SUCCESSFUL (total time: 1 second)
```

4. Conclusions :-

Successfully executed programs about 1-D arrays in java.

5. Limitations of Experiments and Results:-

<u>Drawbacks of arrays in java:-</u>

Deleting or inserting − You cannot insert a new element at the middle of the array. In the same way you cannot delete elements from the middle of the array. You can only insert/delete from the end of the array.

Increasing size − You cannot increase the size of the arrays in Java, if you want to add new elements you need to create new array with extended size and assign to the array reference. This leaves the original object for garbage collection and thus wastage of memory occurs.

Storing Objects − You can store objects in an array but you cannot store objects of different types.

Processing Elements − Except some operations provided by the Array class, you cannot process the contents of an array.

Modifying elements − To delete or, change the elements of an array you need to traverse throughout the array which increases the time complexity.

To overcome these disadvantages, you can use collections instead of arrays.