

## DATA STRUCTURE

Type      Structure & Arrays [2D]

LAB PERFORMANCE

Deadline    **IN CLASS**

Weighting   TBA

2

### OBJECTIVES

This assessment item is designed to test your skills on Structure and multi-dimensional arrays.

### OUTCOMES

After this Lab, you should have clear concept on the following topics:

- **Structure**
  - i. When to use.
  - ii. Understanding the mechanisms behind Structure.
  - iii. Data Storing in and Extracting using Structure.
  - iv. Usage of Array of Structure.
  - v. Perform different operations both Structure.
- **Two dimensional/ 2D Array**
  - i. When to use.
  - ii. Understanding the index of any elements in 2D array.
  - iii. Data Storing in and Extracting from 2D array.
  - iv. Passing an entire 2D array to a function.
  - v. Finding minimum and maximum values.
  - vi. Searching a particular element from the array.
  - vii. Perform different operations both on data and indices.

### PRELIMINARIES

- A code file titled “*PlayerStructure.cpp.txt*” has been uploaded into your VUES account to help you out. You may see the code for your understanding.
- Declaring a Student structure for storing/manipulating student information.

<pre>struct {     int rollno;     char name[35];     int marks; }student;</pre>	<pre>cin&gt;&gt; student.rollno; gets(student.name); cin&gt;&gt; student.marks;  cout&lt;&lt;stud.rollno; cout&lt;&lt;stud.name; cout&lt;&lt;stud.marks;</pre>
---	--

- Passing an entire 2D array to a function.

```
int array[10][10];
```

```
void passFunc(int a[][10])
{
    // ...
}
passFunc(array);
```

## ATTENTIONS


- Use meaningful variable's name.
- Maintain indentation properly.

```
int main ()
{
    // an int array with 5 elements.
    int balance[5] = {1000, 2, 3, 17, 50};
    double avg;

    // pass pointer to the array as an argument.
    avg = getAverage( balance, 5 ) ;

    // output the returned value
    cout << "Average value is: " << avg << endl;

    return 0;
}
```



- Don't copy-paste.

## SCENARIO

As you recall from the last lab that, you are a **Jr. Software Engineer** at ABC solutions, one of the leading software firm in the country. At present, they are working in a software project whose objective is to develop a **Virtual University Management System**. As you are a fresh graduate, so they have given you the task to develop **the Student Module** as the part of the whole system. The requirements of the **Student Module** are given below:

1. **Information are vital. Management systems mainly depend on Information. They process a huge volume of Information to ease our life. Virtual University Management System is not different from others. Here you have to process the information of students in a large scale as they are the main actor. Now, in order to process the students' information, you have to store them first. And before storing them, you have to have an efficient and effective data structure. Let's assume, you have to store the information of students like studentName, studentId, CGPA, marks of 3 quizzes. Which data structure would you use?**

- Implement the program with your proposed data structure with 2/3 student records. Display/print the content of your data structure.
- **Don't take input through the keyboard. Initialize each record directly**

*Input 1:* Ahmed Jalal 123 3.65 10 19 20

*Input 2:* Ahmed Kamal 124 3.95 20 19 20

*Output 1:* **Name:** Ahmed Jalal; **ID:** 123; **CGPA:** 3.65; **Marks of the 3 quizzes:** 10 19 20

*Output 2:* **Name:** Ahmed Kamal; **ID:** 124 ; **CGPA:** 3.95 **Marks of the 3 quizzes:** 20 19 20

.  
. .  
.

2. **Searching is a very important feature in any kind of Management System. Searching enables the users to find out the required information at the fingertip in shortest possible time. Your next goal is to add the Searching mechanism in your existing program.**

- Implement a function to Search a record based on student id and display the information content. If the specified id is not present in the list an error message should be displayed.

Sample Input 1: Enter the id of student for information: 123

Sample output 1: **Name:** Ahmed Jalal; **ID:** 123; **CGPA:** 3.65; **Marks of the 3 quizzes:** 10 19 20

Sample Input 2: Enter the id of student for information: 126

Sample output 2: **Error! Student Record Not found.**

3. **So far, you have implemented the storing and retrieving of students' information. Now the next step is information processing. Previously, you have stored and displayed the marks of three quizzes for each student. Now you have to compute the marks of the best quiz and average of the 3 quizzes and add these as the attributes of the existing student structure.**

- Let's modify the structure for the students first. Then add two functions to calculate the best and average marks of the three quizzes for each student.

Sample Input 1: **Name:** Ahmed Jalal; **ID:** 123; **CGPA:** 3.65; **Marks of the 3 quizzes:** 10 19 20

Sample Output 1: **Name:** Ahmed Jalal; **ID:** 123; **CGPA:** 3.65; **Marks of the 3 quizzes:** 10 19 20; **Best Quiz:** 20; **Average Marks:** 16.33

## **Bonus Problems**

4. Add a search functionality to search by CGPA range. For example, if I want to see the all students' information whose cgpa is less than 2.5 then I have to pass two parameter to the function like displayStudentList (0.00, 2.50) which means displayStudentList( float lowerLimit, float upperLimit).

5. Write a program to search a particular element from a 2D-array and print its position.

Sample Input:

Enter size of the array:

row-3, column-2

Enter elements of this array:

4 5

9 10

8 0

Value to be searched: 10

Sample Output:

Position:     Row-1, column-1

6. Find the highest and lowest value from a 2D array and display their location.
7. Define a struct Date variable called MyBirthDate and initialize it to September 21, 1967, in the dd/mm/yy format. Write a program to display the MyBirthDate. Take your own birth date from key board and display it too.

Sample Case:

Enter your Birth date: 22 June 1993

My Birth Date is in: 22 June 1993