

NIT-J CSED

Computer Programming Lab

Part 1: Array, Pointer and Functions with Array

1. Consider a class having 'n' students. You need to read all the student marks for a particular subject and store only marks in a one dimensional array named as *student_array* and then have to do the following things one-by-one using *recursion*. Marks should be between 0 to 100.

- i) display all the marks on the screen and find the total and average marks scored in that subject.
- ii) find the minimum and maximum among all the marks.
- iii) counting the number of marks scored by the students which are below the average marks, total number of marks which are similar, and frequency of each mark stored in the array.

[Hint: The main() function need to call three sub-functions by passing the array and the number of

students using the following prototype.

void display(int *n) // call by reference

min_max(int n) // call by value

count(int *n)// call by reference

In order to extend the accessibility of *student_array* by all the programs of this assignment, declared *student_array* as '*extern int student_array*' in a header file. Subsequently, include this header file in all the programs. This is how we can share a common variable or function in different files. A sample example is shown at the end which will help you to use the *extern* variable.]

2. Extend the program by separating the odd and even marks into two separate arrays named as *odd_marks* and *even_marks* respectively and finally copied these two arrays into a third arrays named as *odd_even_marks* so that all the odds marks are stored before all the marks which are even.

3. Extend the above program such that if any new student has taken late admission then your code should able to accommodate such new student mark in the *student_array*. To do this, your code should ask the position (i.e., index) where the new mark will be entered in the *student_array*. If any student is leaving the class then your program should handle this by deleting the corresponding mark from the *student_array*.

[Hint: Call two functions *insert_mark* (int position) and *delete_mark* (int mark) from main() function.

4. Extend the program such that it can return the nth highest mark from the *student_array*. To do this first sort all the marks of *student_array*.

5. Extend the program using function which will take the mark of a student as the input and search in the *student_array*. It will display appropriate message '*Found*' or '*Not Found*'. In the case of '*Found*', return the index of the *student_array* where the mark was found.

Part 2 : Multi-dimensional Arrays, pointer and Functions

1. WAP to add, subtract & perform scalar multiplication on 2D matrices.
2. WAP to find the multiplication of matrices A and B into matrix C where the initialization of matrices A and B should be done by a function `initMat()`, printing of matrices should be done by `printMat()` and multiplication should be done by `mulMat()`. You need to pass the matrices into functions according to need.
3. Assume that you now planned to stored subject wise student marks in the same *student_array* as discussed in Assignment-VIII by modifying it to a two dimensional array as shown below for three students and six subject marks.

<i>Rollno/Subject</i>	1	2	3	4	5	6	<i>Total</i>
1	30	83	45	92	60	76	386
2	45	43	67	52	89	56	352
3	28	10	29	68	73	62	270
<i>Total</i>	103	136	141	212	222	194	1008 (Grand Total)

Assuming 'n' students and 'm' subjects, design a menu driven program using function which will do the followings where all the students' rollno and subject name are integers as shown in the above table.

- i) Read those student marks and display them on the screen along with their rollno using pointer.
- ii) Display the subject-wise and student-wise total marks obtained and the grand total.
- iii) Display the student details who obtained the highest mark.

Home Assignment

1. WAP to perform following operation on matrix (3*3) using function and pointer
 1. transpose of matrix
 2. check whether symmetric or not.
2. Write a function which takes two sorted arrays as parameters and merge them into another sorted array. Write a program to implement above function.