Birla Institute of Technology & Science, Pilani
Computer Programming (CS F111)
Second Semester 2015-2016
Lab Sheet # 9

Objectives

1. Arrays and Functions

By now you know how to break your program in to modules and how to pass arguments to functions. Try to implement the following questions. Break the problems into proper functions.

1) Consider two arrays: x-coor[] and y-coor[]. The ith location of the two arrays stores the x and y coordinates of a point. For example, x-coor[4] and y-coor[4] stores the x and y coordinates of forth point.

a) Pass these two arrays to a function which prints the quadrant in which each point lies. Emphasis is on passing array to a function.

b) Pass these two arrays to a function. The function stores the quadrant in which each point lies in another array (which is also passed to the function). It then prints the quadrant in main function. Emphasis is on returning array from a function.

2) Write a function to delete duplicate elements from an array of 20 integers. Test it by calling it from main()

3) A common problem in statistics is that of generating frequency distribution of the given data. Assuming that the data consists of 50 positive integers in the range of 1 to 5, write a function that prints the number of stars (*) each integer occur in the data.

4) Write a program to calculate class average for the following 10 students and 5 courses. M1 to M5 are marks in course1 to course5 respectively. Marks are out of 50. Marks of the students are shown in the table below. Total is initialized to zero (It is compute and populated in the array later.)

Id No	M1	M2	M3	M4	M5	Total
1	32.5	43	23	12	44	0
2	23	23.5	45.5	25	19	0
3	34	13.5	32.5	21.5	26	0
4	41	21	31	14	12	0
5	22	42	45	12	14	0
6	33	44	21	11	29	0
7	39	47	28	17	48	0
8	38	27	36	19	39	0
9	49.5	28.5	46	20.5	50	0
10	29	54	33.5	22.5	10	0

Divide the problem into sub problems. Write function for each sub-problem and call all the functions in main function. Write the following functions:

(i) **populateMarksArray**(): Write a function to read values in two dimensional array of marks from the above table. Populate id no in first column, marks in other columns and initialize last column with zeros. Identify input arguments and return type of the function. [Note: Ignore first row from above table].
(ii) **computeTotal()**: Write a function to compute total marks of each student and update the marks array.
(iii) **computeClassAverage()**: Write a function to calculate average of all the students in the class and return it to main function.
(iv) **computeMaxTotal()**: Write a function to find a student with maximum total marks. Function returns id no of the student with maximum total.

5) Write a program to sort the list of students id numbers for five different courses. Assume number of students registered in all five courses are same. The following table shows id numbers of students registered in each course row wise. Declare a two dimensional array to populate the data in the table.

C1	89	65	90	45	23	10	43	23	18	12
C2	85	60	95	40	21	18	40	27	13	17
C3	80	62	91	42	22	12	41	25	10	71
C4	81	61	94	49	29	13	46	21	12	88
C5	83	66	96	47	27	14	48	28	19	55

Accomplish the task by writing following functions:
(i) **populateArray()**: Write a function to read the id numbers in the table shown above. Populate course numbers in first column and id numbers in rest of the columns. Use 1 for C1, 2 for C2 and so on. It does not return anything.
(ii) **sortRows()**: Write a function which sorts the id numbers of the students for each course separately. Here sorting should be done row-wise.
Call both the functions in main. Print the final array in main to see the list of sorted id numbers for each course.