CS19101 PDS laboratory
Assignment7

Write programs for problems 1,2 and 3 in three different files named A7_1_<machine number>_<Roll no.>.c, A7_2_<machine number>_<Roll no.>.c and A7_3_<machine number>_<Roll no.>.c respectively (without the '<' and '>'). Put these three files into a compressed directory named A7_<machine number>_<Roll no.>.zip and submit it.

Example: If your roll number is 19DEP99999 and your machine number is 99, then the names of your files should be A7_1_99_19DEP99999.c, A7_2_99_19DEP99999.c and A7_3_99_19DEP99999.c.

---

1. Declare a global structure **AcadRecord** for storing academic record of a student. It holds the following three pieces of data:

   (a) Name of the student. Use a character array/string of size 100. May contain spaces.

   (b) An alphanumeric roll number. Use a character array of size 10.

   (c) A score out of 100. Use an integer variable. It's value is an integer between 0 and 100 (both inclusive).

   Write a function named **FindTop** with the following prototype:

   void FindTop(struct AcadRecord[]);

   FindTop takes in as argument an array of structures of type **AcadRecord** of size 5. **FindTop** finds in the array the record with the maximum score, and prints the name and roll number of the corresponding student. If there are more than one record with maximum score, it prints the name and roll number of the student whose name appears earliest in the alphabetical order. You can use the string.h library function **strcmp()**.

   In main() define an array of this structure of size 5. Fill them all by inputs through the keyboard in the main(). Then call **FindTop** on this array.

Sample input/output:

Enter name of student 1: Akash
Enter roll of student 1: ho87nu8
Enter score of student 1: 99
Enter name of student 2: Akash Roy
Enter roll of student 2: 76t76n
Enter score of student 2: 99
Enter name of student 3: Sonali Sinha
Enter roll of student 3: uh8h7t8n
Enter score of student 3: 97
Enter name of student 4: Shilpa Saini
Enter roll of student 4: nkuh8oho7
Enter score of student 4: 99
Enter name of student 5: Siddhesh
Enter roll of student 5: kj98j
Enter score of student 5: 96
Name: Akash
Roll: ho87nu8

[20 marks]

2. Make the following global structure declaration:

struct matrix{
int M[4][4];
}

In main() define three structure variables of this type. Name them A, B, C. Fill up the arrays in A and B by inputs through the keyboard. Write a function with the following prototype:

struct matrix MatMul(struct matrix *p, struct matrix *q);

MatMul multiplies the matrices in structures pointed to by p and q, and returns a structure variable of type **matrix** that contains the product matrix.

In main(), pass pointers to A and B to **MatMul** to multiply matrices in A and B. Assign to C what **MatMul** returns. Finally print on the screen the matrix in C (see sample input/output).

Sample input/output:

Matrix 1:
Enter element (1,1): 1
Enter element (1,2): 2
Enter element (1,3): -1
Enter element (1,4): 3

```
Enter element (2,1): 4
Enter element (2,2): 3
Enter element (2,3): -2
Enter element (2,4): -1
Enter element (3,1): 0
Enter element (3,2): 0
Enter element (3,3): 1
Enter element (3,4): 1
Enter element (4,1): 4
Enter element (4,2): 1
Enter element (4,3): 4
Enter element (4,4): 2
Matrix 2:
Enter element (1,1): 0
Enter element (1,2): 0
Enter element (1,3): 0
Enter element (1,4): 2
Enter element (2,1): 3
Enter element (2,2): 1
Enter element (2,3): -4
Enter element (2,4): -2
Enter element (3,1): 5
Enter element (3,2): -3
Enter element (3,3): -4
Enter element (3,4): 2
Enter element (4,1): 1
Enter element (4,2): -5
Enter element (4,3): 2
Enter element (4,4): 4
4 -10 2 8
-2 14 -6 -6
6 -8 -2 6
25 -21 -16 22
```

[20 marks]