

# National Institute of Technology Calicut

## Department of Computer Science and Engineering

B. Tech. (CSE) – Third Semester

### CS2092D: Programming Laboratory

### Extra Questions (Assign\_02\_part-A)

#### General Instructions

- Programs should be written in C language and compiled using C compiler in Linux platform.
- Invalid input should be detected and suitable error messages should be generated.
- Sample inputs are just indicative.
- The extra questions do not come under the purview of evaluation scheme.

#### Part-A: Functions, Parameter Passing, and Recursion

1. Write a program to perform multiplication of two matrices A and B using recursion.

<b>Input</b>	Enter rows and columns for Matrix A	: 2 2
	Enter rows and columns for Matrix B	: 2 2
	Enter elements in Matrix A	: 2 2 3 4
	Enter elements in Matrix B	: 1 2 3 4
<b>Output</b>	Resultant Matrix is	: 8 12 15 22

**Input**

Enter rows and columns for Matrix A	: 2 3
Enter rows and columns for Matrix B	: 2 2
Enter elements in Matrix A	: 12 56 10 45 78 15
Enter elements in Matrix B	: 2 6 5 8

**Output** Matrix multiplication is not possible.

- Write a program to implement the 'Tower of Hanoi' using recursion. Tower of Hanoi is a mathematical puzzle that contains three rods A, B, and C. Rod A contains 'n' disks arranged in decreasing order of their size (from bottom to top of the rod as shown in Figure 1). The objective of the puzzle is to move the 'n' disks from rod A to rod C, obeying the following simple rules:
  - Only one disk can be moved at a time
  - Each move consists of moving the upper disk from one of the rods and placing it on top of another rod i.e., a disk can only be moved if it is the uppermost disk on a rod.
  - No disk may be placed on top of a smaller disk.

Input is the number of disks, 'n'. Output should be the pattern of movements of the 'n' disks to reach rod C from rod A.

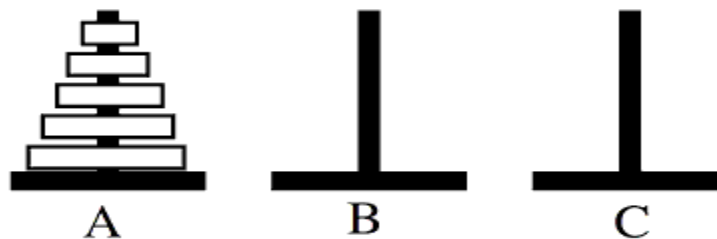


Figure 1: Tower of Hanoi

**Input** : 2  
**Output** : Disk 1 moved from A to B  
 Disk 2 moved from A to C  
 Disk 1 moved from B to C

**Input** : 3  
**Output** : Disk 1 moved from A to C  
 Disk 2 moved from A to B  
 Disk 1 moved from C to B  
 Disk 3 moved from A to C  
 Disk 1 moved from B to A  
 Disk 2 moved from B to C  
 Disk 1 moved from A to C

3. Write a program to find whether two matrices A and B are equal or not. The program must include the following functions:
- read\_matrix (X,m,n): reads the elements of matrix X with ‘m’ rows and ‘n’ columns.
  - check\_matrix (X,Y): checks whether the matrices X and Y are equal .

**Input** Enter the order of the matrix A : 22  
 Enter the order of the matrix B : 22  
 Enter the elements of matrix A : 23 56  
 45 80  
 Enter the elements of matrix B : 50 26  
 39 78

**Output** Two matrices are not equal

**Input** Enter the order of the matrix A : 2 2  
 Enter the order of the matrix B : 2 2  
 Enter the elements of matrix A : 10 50  
 15 30  
 Enter the elements of matrix B : 10 50  
 15 30

**Output:** Two matrices are equal

4. Given the date of birth, write a program to calculate the age of a person using function. Function name should be ***dobToAge( DOB )*** which takes the date of birth as an argument and return the age of a person.

<b>Input</b>	Enter your date of birth (DD/MM/YYYY) :	18/10/1999
	Current Date :	12/03/2018

**Output**      Age is: 20 years 4 months and 26 days

5. Write a program to perform binary addition and subtraction of two given positive integers using functions. The program should read two positive integers, convert them to binary equivalent, perform binary addition and subtraction, and display the corresponding decimal numbers.

Your program should have following functions.

- (a) **int\* decToBinary(int x) :** This function takes an integer as an argument and return a pointer to an integer array.

- (b) **int\* binaryAddition(int\*, int\*)** : This function takes 2 pointers pointing to an integer array and return a pointer to an integer array.

- (c) **int\* binarySubtraction(int\*, int\*)** : This function takes 2 pointers pointing to an integer array and return a pointer to an integer array.

<b>Input</b>	Enter first integer value :	20
	Enter second integer value:	10

<b>Output</b>	Binary equivalent of 10 is:	1010
	Binary equivalent of 20 is:	10100
	Binary Sum is :	11110
	Decimal Sum is :	30
	Binary difference is:	01010
	Decimal difference is:	10

\*\*\*\*\*