

Practice Problems on String, Multidimensional Array and Recursion

- Write a program to perform matrix multiplication. Your program will take as input dimension (number of rows followed by number of columns) and elements of two matrices in row major order. The output will be the dimension and elements of the resultant matrix after multiplication. You can assume no dimension will be greater than 5.

Sample Input Dimension followed by elements	Sample Output
2 3 1 2 3 5 0 4 3 1 1 5 6	2 1 29 29
3 3 10 20 10 4 5 6 2 3 5 3 3 3 2 4 3 3 9 4 4 2	3 3 130 120 240 51 47 73 35 33 45
2 1 10 20 2 2 10 20 30 40	Multiplication not possible

- Write a function to rotate a square matrix clockwise by 90 degree. Your program will take as input dimension of the square matrix and elements of the matrix in row major order. Since this is a square matrix, a single integer n will be given as dimension to indicate $n \times n$ square matrix. Note that, you cannot declare any additional multidimensional array for this problem. You can assume no dimension will be greater than 5.

Hint: Transpose the matrix at first and then reverse each row.

Sample Input	Sample Output
3 3 2 4 3 3 9 4 4 2	4 3 3 4 3 2 2 9 4

- Write a function which takes a string as parameter and removes vowels from it. You cannot declare any additional character array. String length will not exceed 100.

Sample Input	Sample Output
--------------	---------------

ABCdeFghij	BCdFgHj
fjnesdkfjei	Fjnsdkfj
Ab13 ewyLivs9; o]d	b13 wyLvs9;]d

4. Write a function which takes a string and a delimiter character as parameter and splits the string using that delimiter, stores the substrings in a 2D character array and print the delimited substrings. String length will not exceed 100 and no. of delimited substrings will be no more than 30. You cannot use any library function from string.h.

Additional Task: remove any leading and trailing space from the delimited substrings.

H.W. -> Assume the delimiter is a string instead of a character

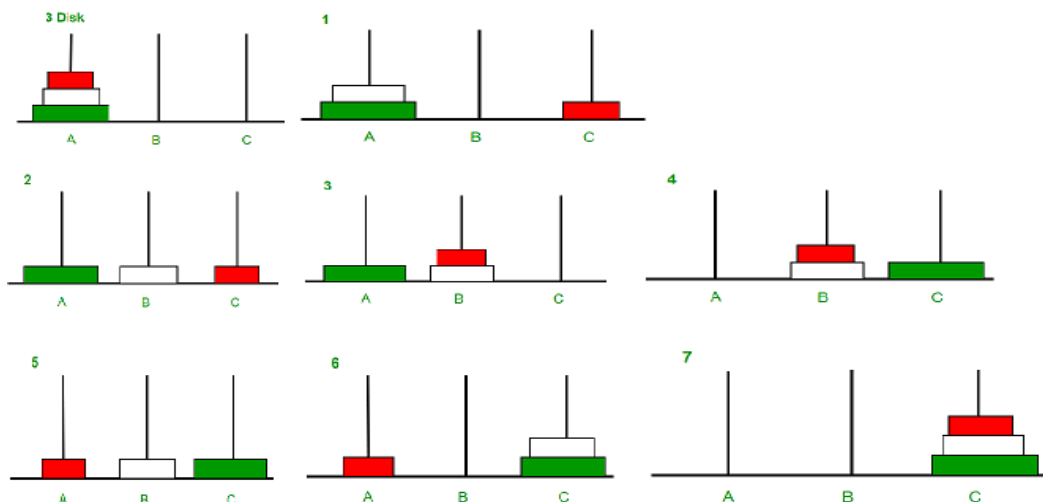
Sample Input	Sample Output
Abc,Def,ghi,jkl ,	Abc Def ghi jkl
cse 101; cse 102; eee 108 ; math 212 ;	cse 101 cse 102 eee 108 math 212

5. Solve the TOH problem with recursion.

Tower of Hanoi (TOH) is a mathematical puzzle where we have three pegs (rods) and n disks. The objective of the puzzle is to move the entire stack from one peg to another, obeying the following simple rules:

- 1) Only one disk can be moved at a time.
- 2) Each move consists of taking the upper disk from one of the stacks and placing it on top of another stack i.e., a disk can only be moved if it is the uppermost disk on a stack.
- 3) No disk may be placed on top of a smaller disk.

Assume that 'A' is the source peg and 'C' is the destination.



Hint for recursive solution:

Problem: Move n disks from source peg to destination peg using auxiliary peg

Base case: If one disk remains, move it from source to destination

Recursion:

- Move $n-1$ disks from source to auxiliary using destination
- Move n^{th} disk from source to destination
- Move $n-1$ disks from auxiliary to destination using source

Sample Input (No. of disks)	Sample Output
2	Disk 1 moved from A to B Disk 2 moved from A to C Disk 1 moved from B to C
3	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

6. Implement Power function (for integers) to minimize number of multiplications

Hint:

$2^8 = 2 * 2 * 2 * 2 * 2 * 2 * 2 * 2$ (7 Multiplications)

But it can be done using 3 multiplications in the following way.

$2^2 = 2^1 * 2^1$ (1st Multiplication)

$2^4 = 2^2 * 2^2$ (2nd Multiplication)

$2^8 = 2^4 * 2^4$ (3rd Multiplication)

Sample Input (base, power)	Sample Output
2 8	256
3 11	177147