CS2710: Programming and Data Structures Lab
Home Assignment #2: Problem Solving using Recursion

Due at 11:55 pm, August 21, 2016

August 16, 2016

Problem 1

Calculate the determinant of a given square matrix of integers.

Input Description: First line of the input will contain a single integer N, followed by N lines,the rows of the matrix, each containing N integers. **Note:** You can assume that $1 \leq \mathrm{N} \leq 10$ and elements of the matrix will fit into 32-bit signed integer whereas determinants will fit into 64-bit signed integer.

Expected Output: A single line containing an integer, i.e. the **determinant** of the matrix

example:
Input:

$$\begin{array}{cccc} 4 & & & \\ 1 & 2 & 3 & 4 \\ 3 & 4 & 5 & -1 \\ 6 & 0 & -2 & 4 \\ -9 & 1 & 7 & 3 \end{array}$$

Output:
212

Problem 2

Solve the N-Queens problem. You are expected to place N queens on an $N \times N$ chess board such that none of the queens can attack the others i.e. they do not share the same row, column or diagonal.
Assume the chess board to be a grid with rows and columns both going from 0 to $N-1$. The position of each cell is thus represented by (i, j) where $i, j \in [0, N-1]$.

To make things clear, suppose the size of the chess-board/grid is 5×5, then possible attacking and non-attacking position from cell (1,2) are shown below, where positions marked by asterisk(*) can be attacked by queen at (1,2) and those with 0 can't be:

0	∗	∗	∗	0
∗	∗	1	∗	∗
0	∗	∗	∗	0
∗	0	∗	0	∗
0	0	∗	0	0

Input Description: A single line containing an integer, N, the size of the grid and the number of queens. You can assume that $1 \leq N \leq 12$
Expected output: As many lines as the number of possible non-attacking configurations with each line containing N integers where i^{th} integer represents the non-attacking cell of the i^{th} row followed by a line containing single integer representing the number of such configurations.

Ex: for N=4, there are only two possible configurations to place queens in non-attacking positions, which are following-

0	1	0	0		0	0	1	0
0	0	0	1		1	0	0	0
1	0	0	0		0	0	0	1
0	0	1	0		0	1	0	0

Cells containing 1s correspond to the positions of the queens
Note: Refer to this example to understand the output format, as described above, for the sample input assuming 0-based indexing.

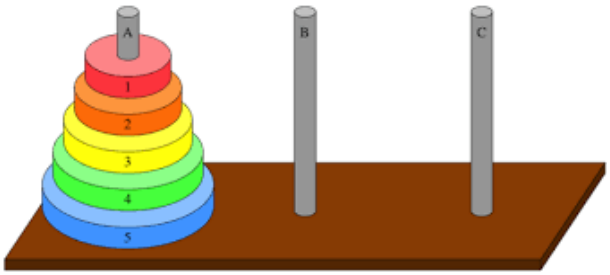
Input:
4
Output:
1 3 0 2
2 0 3 1
2

Problem 3

Solve the Towers of Hanoi puzzle. There are three pegs, say A, B and C. The peg A has n disks (each of different size) attached to it. The disks are arranged in decreasing order of size, with the smallest on top. The objective is to move these disks to the peg C, while following the given rules:

- Only one disk can be moved at a time
- Only the topmost disk of a peg can be moved
- A moved disk can be placed only on the top of the disks on any peg
- No disk can be placed above a disk smaller than itself

Problem 3 continued on next page...



Input Description: A singleline containing an integer,n, the number of disks at A before the game starts.

Expected output As many number of lines as the number of possible movements of disks between all pairs of pegs until all of them are moved to C with each line containing output in the following format, followed by a line containing the number of movements
"source_peg -> dest_peg disks_at_A disks_at_B disks_at_C"

Ex:
Input:
3
Output:
A -> C 2 0 1
A -> B 1 1 1
C -> B 1 2 0
A -> C 0 2 1
B -> A 1 1 1
B -> C 1 0 2
A -> C 0 0 3
7

Note: As a matter of curiosity, you can also attempt this problem with more pegs, and think about how to approach this with an arbitrary number of pegs. This is not for evaluation, at least not within this deadline.