ECS-202: Data Structures and Algorithms

Assignment-3

Deadline: 18.03.2020(23:59) **Total Marks:20**

1 Answer all Questions. $2 \times 5 = 10$

1. Given a positive integer N, construct all possible structurally unique binary search trees with node values $1, 2 \ldots n$. Note that, there are no missing values for a given N, i.e. the trees you look for have all the nodes with values $1, 2 \ldots n$. You should display the 'pre-order' traversals of each of the possible trees and the total count of the possible number of such trees.
   Input: The program should take a single integer as input(N).
   Output: The output should be a single non-negative integer which is the total number of possible trees for the given N followed by 'pre-order' traversals of each possible binary search tree.

2. Given a pre-order traversal of a binary search tree(BST) and a range, say `[x,y]`, write a program that constructs a binary tree with the given traversal and then removes all the nodes for which the values are in the given range maintaining the BST nature of the tree with necessary changes/modifications. You may assume that the 'pre-order' traversal given initially as input is a valid one and need not verify that.
   Input: The input should be a 'pre-order' traversal of a binary search tree and two values which are the non-negative bounds of the range.
   Output: The output should be a the 'pre-order' traversal of the modified binary tree with no nodes in that range.

2 Answer the following $1 \times 10 = 10$

1. Write a program that is capable of creating a binary tree starting from the root node using structures and dynamic memory allocation. Ensure that the tree constructed is binary. Once the user creates the binary tree at run-time, your program should also check if the binary tree is a binary-search tree or not. If not, it should output the node at the minimum depth from the root which violates the condition for a binary search tree.
   Input: The code initially asks for the root node and then dynamically creates a binary tree with non-negative integer values until the user is done.
   Output: The output should be a single string displaying 'YES' or 'NO'

accordingly if the tree entered is binary search tree or not. If not, it should display: `x node at i depth violates the BST condition`
Example: One way to do is to ask for the root node to begin with, then keep asking for a left/right child; followed by a prompt which shows a current root value and asks for its subsequent children and so on. Two special characters, say `!` and `*` can be used to denote that the user is not interested in child node values for the current node and end of the tree respectively. In this way, you create the binary tree, after which you should code to check if it is a binary search tree or not and display the output.(You are free to implement in any way you please)

3 Mode of submission and evaluation

Mode of submission: You have to submit all your codes in the C-Programming language. A google form link given below and also on the Google Classroom page of the course is where you are required to upload the code (extension .c file) and a 'readme'(extension .txt or .pdf) *for each of the problems separately.* The readme should contain the necessary information(if any) for the execution of the code.
Additionally, if your code has bugs that you failed to resolve before the deadline, please upload the faulty code and mention the algorithm that you implemented and/or your routemap for the solution and some analysis of the bugs(if you managed any) in the 'readme' file.

Evaluation: For problems of **Section-1**, you will be awarded marks only for successful execution of your code. For the problem in **Section-2**, partial marks will be awarded in case you write a correct algorithm in the readme(even with a faulty code).
The links for submission will expire sharp at the deadline and no other form of submission will be evaluated.[1]For programmes which look similar or we get a sense of copying or other malpractices, zero mark will be awarded to all such instances.

Submission link: `https://forms.gle/jy3xA1E6KmCZrATSA`

End

[1]P.S. Prefer writing proper indented and commented codes. It is considered a good practice.