National Institute of Technology Calicut
Department of Computer Science and Engineering
Second Semester B. Tech.(CSE)
CS1092E Program Design Laboratory
Assignment #4
Submission deadline (on or before): 26/03/2024, 08:00 PM

Naming Conventions for Submission

- Submit a single ZIP (.zip) file (do not submit in any other archived formats like .rar, .tar, .gz). The name of this file must be

  `<FIRST-NAME>_<ROLLNO>.zip`

  (Example: *LAXMAN_BxxyyyyCS.zip*). DO NOT add any other files (like temporary files, input files, etc.) except your source code, into the zip archive.

- The source codes must be named as

  `<FIRST-NAME>_<ROLLNO>_<PROGRAM-NUMBER>.c`

  (For example: *LAXMAN_BxxyyyyCS_1.c*). If you do not conform to the above naming conventions, your submission might not be recognized by our automated tools, and hence will lead to a score of 0 marks for the submission. So, make sure that you follow the naming conventions.

Standard of Conduct

- Violation of academic integrity will be severely penalized. Each student is expected to adhere to high standards of ethical conduct, especially those related to cheating and plagiarism. Any submitted work **MUST BE an individual effort**. Any academic dishonesty will result in zero marks in the corresponding exam or evaluation and will be reported to the department council for record keeping and for permission to assign F grade in the course. The department policy on academic integrity can be found at: http://cse.nitc.ac.in/sites/default/files/Academic-Integrity_new.pdf.

General Instructions

- Programs should be written in C language.
- Check your programs with sufficiently large values of inputs with in the range as specified in the question.
- Global and/or static variables should not be used in your program.

Q1 Write a menu driven C program for a binary search tree (BST) of integers which does inorder traversal to print its elements.

Your program must contain the following functions:

- Main() - This function repeatedly reads a character from the menu list through the terminal and calls the sub-functions appropriately, as mentioned in Input format, until character ‘e’ is entered. The list of menu options are [‘i’, ‘p’, ’d’, ’s’ ,’t’, ’b’ or ‘e’].
- InsertKey(Root,num) – This function inserts number num in the correct position of the BST whose root is pointed by Root.
- SearchKey(Root, num) – This function checks whether the number num is present or not in the BST whose root is pointed by Root. Print ‘PRESENT’ or ‘NOT PRESENT’ accordingly.
- DeleteKey(Root, num) – This function deletes the number num from the BST whose root is pointed by Root. Print ‘NOT PRESENT in the BST’, if the num is not present in the BST.
- Inorder(Root) - This function prints the inorder traversal of the BST whose root is pointed by Root.
- Preorder(Root) - This function prints the preorder traversal of the BST whose root is pointed by Root.
- Postorder(Root) - This function prints the postorder traversal of the BST whose root is pointed by Root.

Input/Output format

- Your input can contain two parts separated by space; the first part is the input character from the menu list. The second part feeds the parameters according to the menu option.
- Each line in the input starts with a character from {i,p,d,s,t,b,e}.
- Input ‘i’ followed by integer num performs the function InsertKey(Root,num).
- Input ‘s’ followed by integer num performs the function SearchKey(Root,num).
- Input ‘d’ followed by integer num performs the function DeleteKey(Root,num).
- Input ‘p’ performs the function Inorder(Root). Print NULL if the BST is empty. The entire output should be printed on a separate line.
- Input ‘t’ performs the function Preorder(Root). Print NULL if the BST is empty. The entire output should be printed on a separate line.
- Input ‘b’ performs the function Postorder(Root). Print NULL if the BST is empty. The entire output should be printed on a separate line.
- Input ‘e’ exits from the program.

NOTE: Your program **should not** contain any arrays. Assume that the BST at every stage always has unique key values.

Your functions should perform the following time complexity for a BST of height h.

- InsertKey(Root,num) – O(h)
- SearchKey(Root, num) – O(h)
- DeleteKey(Root, num) – O(h)
- Inorder(Root) – O(n)
- Preorder(Root) – O(n)
- Postorder(Root) – O(n)

Sample Input

i 5
i 3
i 10
i 15
p
s 10
d 10
p
i 1
i 2
p
t
b
e

Sample output

3 5 10 15
PRESENT
3 5 15
1 2 3 5 15
5 3 1 2 15
2 1 3 15 5

Q2 Design a C program featuring a menu-driven interface for a binary search tree (BST) consisting of integers and can output the k-th smallest integer in the tree.

Your program must contain the following functions:

- Main() - This function repeatedly reads a character from the menu list through the terminal and calls the sub-functions appropriately, as mentioned in Input format, until character ‘e’ is entered. The list of menu options are [‘i’, ‘p’, ’d’, ’s’ ‘m’, ‘n’ or ‘e’].
- InsertKey(Root,num) – This function inserts number num in the correct position of the BST whose root is pointed by Root.
- SearchKey(Root, num) – This function checks whether the number num is present or not in the BST whose root is pointed by Root. Print ‘PRESENT’ or ‘NOT PRESENT’ accordingly.
- DeleteKey(Root, num) – This function deletes the number num from the BST whose root is pointed by Root. Print ‘NOT PRESENT in the BST’, if the num is not present in the BST.
- KthSmallest(Root, k) - For an integer k, this function prints the k-th smallest integer of the BST whose root is pointed by Root.
- Maximum(Root) - This function prints the maximum valued element of the BST whose root is pointed by Root.
- Successor(Root,num) - This function given number num in the BST (whose root is pointed by Root) prints the next node in Inorder Traversal of the BST.

NOTE: When you define the structure for a node of the BST, it **should not** contain the parent pointer. Your program **should not** contain any arrays. Assume that the BST at every stage always has unique key values.

Your functions should perform the following time complexity for a BST of height h.

- InsertKey(Root,num) – O(h)
- SearchKey(Root, num) – O(h)
- DeleteKey(Root, num) – O(h)
- KthSmallest(Root, k) - O(n)
- Maximum(Root) - O(h)
- Successor(Root,num) - O(h)

Input/Output format

- Your input can contain two parts separated by space; the first part is the input character from the menu list. The second part feeds the parameters according to the menu option.
- Each line in the input starts with a character from {i,d,s,k,e}.
- Input ‘i’ followed by integer num performs the function InsertKey(Root,num).

- Input ‘s’ followed by integer num performs the function SearchKey(Root,num).
- Input ‘d’ followed by integer num performs the function DeleteKey(Root,num).
- Input ‘k’ followed by integer k performs the function KthSmallest(Root, k).
- Input ‘m’ performs the function Maximum(Root).
- Input ‘n’ followed by integer num performs the function Successor(Root,num).
- Input ‘e’ exits from the program.

Sample Input

i 5

i 3

i 10

i 15

k 3

d 10

k 3

i 10

k 2

s 10

m

n 5

e

Sample output

10

15

5

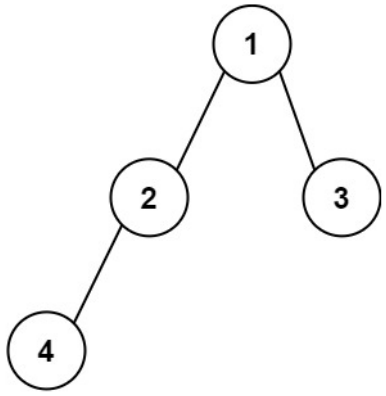
PRESENT

15

10

Q3- A Parenthesis Representation of a **binary tree** is a string consisting of parentheses and integers (separated by spaces) representing a binary tree.
The representation should be based on a preorder traversal of the binary tree and must adhere to the following guidelines:

- Each node in the tree should be represented by its integer value.
- Parentheses for Children: For every node (either left or right), its children should be represented inside parentheses. Specifically:
    - If a node has a left child, the value of the left child should be enclosed in parentheses immediately following the node's value.
    - If a node has a right child, the value of the right child should also be enclosed in parentheses. The parentheses for the right child should follow those of the left child.



For example, for the above binary tree, its parentheses representation is (1 (2 (4 ( ) ( )) ( )) (3 ( ) ( ))). Note that empty parentheses are in the string when a node has a left and/or right child that is empty.

Given the Parenthesis Representation of a binary tree T with n nodes (where n ∈ [1, 10^3]), a node with key value x and a positive integer k<n, print all the nodes at a depth of k in the subtree of T rooted at x. Assume unique key values.

Input format:
• First line of the input contains a space separated Parenthesis Representation of the tree T with key values ∈ [1, 10^6].
• Second line contains an integer x.
• Third line contains an integer k ∈ [1, 10^3].

Output format:
• The output consists of a single line with space-separated key values**.**
Sample Input 1:
(2 (7 (10 ( ) ( )) (6 (5 ( ) ( )) (11 ( ) ( )))) (9 ( ) (3 ( ) ( ))))
7
2
Sample Output 1:
5 11