# CS69201: Computing Lab-1
# Assignment 3 - Binary Search Tree
# August 13, 2024

## ==================== Instructions=====================

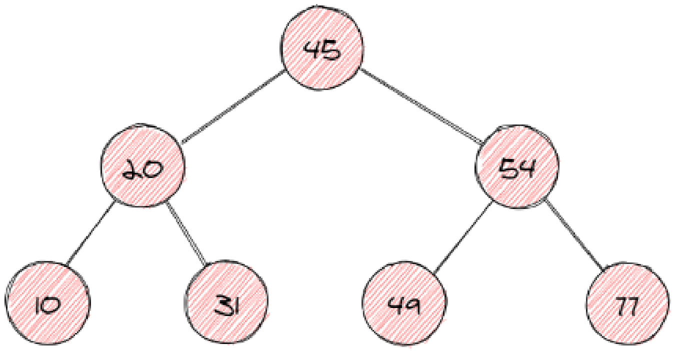
1. In the case of user input assume only valid values will be passed as input.
2. You can use C or C++ as the programming language. **However, you are not allowed to use any STL libraries in C++**
3. Regarding Submission: Create a separate C ( or cpp) file. -> <rollno>_Q1.c. Create a zip in the name <rollno>_BST.zip and submit it to Moodle. For example, if your roll number is 24CS60R15, then your file name will be 24CS60R15_Q1.c and your zip file name will be 24CS60R15_BST.zip.
4. **Inputs should be taken from a file and outputs should be printed to a file named output.txt. The input file name will be passed as a command line argument. Not using the proper files for i/o will incur a penalty**
5. You have been provided a boilerplate code. You may use it according to your requirements.

==========================================

**Question 1 :**

Implement the following functionality of BST in a **single program** with menu menu-driven format.

- Insert new node
- Search a node
- Delete a node
- Find_height of the tree
- Print the Preorder of the tree
- Print the Postorder of the tree
- Print the inorder of the tree
- Level order
- zigzag order



You can use the following numbers for each option in the menu:
1 = insert
2 = search
3 = delete
4 = height
5 = Preorder
6 = Postorder
7 = Inorder
8 = Level order
9 = Zigzag order
Note: 10 in the input indicates end of the input

**Sample Input**

```
1 45
1 54
1 20
1 9
1 31
3 9
1 10
1 49
1 77
2 9
4
5
6
7
8
```

```
9
10
```

**Sample Output:**

```
Not Found
The height of the tree is 2
Preorder: 45 20 10 31 54 49 77
Postorder: 10 31 20 49 77 54 45
Inorder: 10 20 31 45 49 54 77
Level order: 45 20 54 10 31 49 77
Zig zag order: 45 54 20 10 31 49 77
```