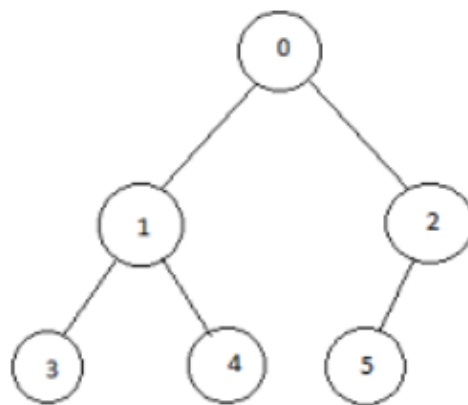


Topic	Binary Tree implementation [Total 25 points]
-------	--

Goal of this programming assignment: The key objective of this assignment is for students to learn how to implement a range of data structures. This assignment will include the concepts of linked lists, structs, standard binary trees and threaded binary trees.

Problem description: In computer data structures, a binary tree is used for searching and sorting data in a hierarchical way. A binary tree can either be an empty binary tree or it can either be a complete or full binary tree. The standard operations in a binary tree are insertion, deletion and traversal. In this assignment students are required to implement these 3 operations in the form of functions and then call these functions when there is a need for the operation.

Programming activity 1: You are required to implement code that creates the binary tree below.



Program requirements: Your program should have the following functions.

Function Name	Description	Points
insert_Root()	If tree is empty insert the node as root. If root exist give error, "This is not an empty binary tree."	5 points
insert_Left()	Inserts a left child of the parent	5 points
insert_Right()	Inserts a right child of parent	5 points
inorder_Succ()	Finds the inorder successor of node X.	5 points
inorder_Trav()	Find the inorder traversal of the tree.	5 points

Required console output:

Your program should provide the following console:

Binary tree functions: Select the function you like to proceed with,

1. Insert root
2. Insert left
3. Insert right
4. Inorder Successor
5. Inorder Traversal

Selection choice: 1

Enter the root node information: 0

The root node is 0.

Binary tree functions: Select the function you like to proceed with,

1. Insert root
2. Insert left
3. Insert right
4. Inorder Successor
5. Inorder Traversal

Selection choice: 1

Error: This is not an empty binary tree.

Binary tree functions: Select the function you like to proceed with,

1. Insert root
2. Insert left
3. Insert right
4. Inorder Successor
5. Inorder Traversal

Selection choice: 2

Enter the parent information: 0

Enter the left child information: 1

The left child is 1.

Binary tree functions: Select the function you like to proceed with,

1. Insert root
2. Insert left
3. Insert right
4. Inorder Successor
5. Inorder Traversal

Selection choice: 3

Enter the parent information: 0

Enter the right child information: 2

The right child is 2.

Binary tree functions: Select the function you like to proceed with,

1. Insert root
2. Insert left
3. Insert right
4. Inorder Successor
5. Inorder Traversal

Selection choice: 4

Enter the node which you want to find the successor: 0

The inorder successor of 0 is 5

Binary tree functions: Select the function you like to proceed with,

1. Insert root
2. Insert left
3. Insert right
4. Inorder Successor
5. Inorder Traversal

Selection choice: 5

The inorder traversal of the tree is 3 1 4 0 5 2

Files to upload:

1. Screenshot of the above-mentioned console of the program.
2. C source code file. Name the file tree.c

Point deduction policy:

Late submission penalty: minus 5 points.

Program does not compile: minus 50% of total points.