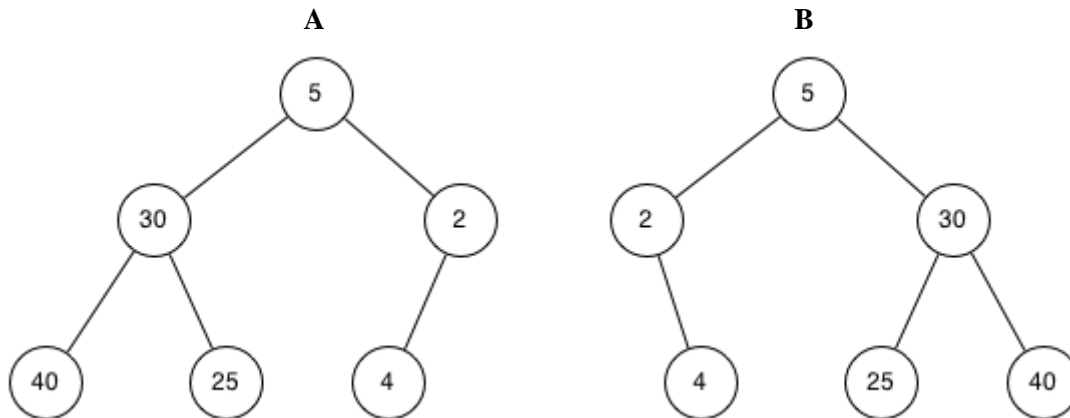


## TK1143 Program Design TREE

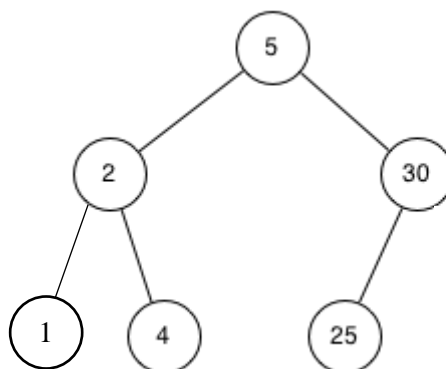
### Section A:

1. Given tree A and B as follows.



Which of the tree shows the binary search tree (BST) after inserting keys in the following order: 5, 30, 2, 40, 25, 4. Justify your answer.

2. Given the following tree.



- a) Identify root, parent, child and leaf node.

- b) Illustrates the process for inserting a new node with key 19 into a binary search tree. Justify your answer.

- c) Illustrates the process for deleting node with key 4 followed by key 5 into a current binary search tree in (2b). Justify your answer.

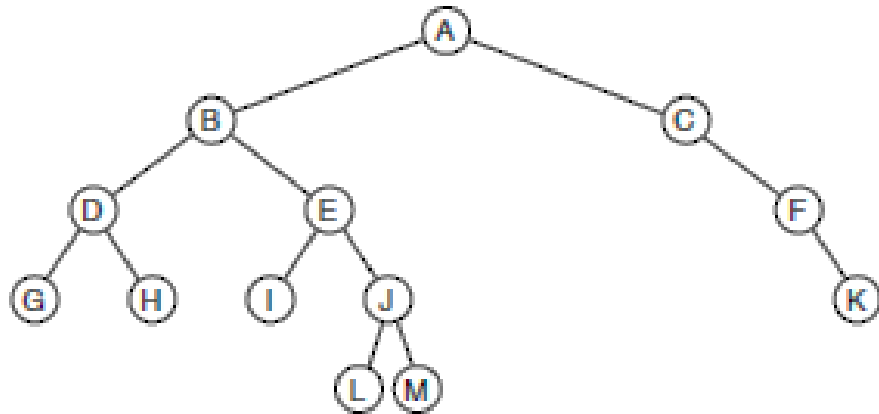
Delete 4

Delete 5

- d) Illustrates the process for searching node with key 40 and 4 into a current binary search tree (2c). Justify your answer.

Search 40	Search 4

3. Answer all the following questions based on the following tree.



a. Is the tree above fulfill the requirement of BST? Justify your answer.

**Section B:**

4. Consider the following code.

```

1 public class BinarySearchTree {
2     /* Class containing left and right child of current node and key value*/
3
4
5
6
7
8
9
10
11
12     // Root of BST
13     Node root;
14     // Constructor
15     BinarySearchTree() {
16         root = null;
17     }
18     // This method mainly calls insertRec()
19     void insert(int key) {
20         root = insertRec(root, key);
21     }
22     // A recursive function to insert a new key in BST
23     Node insertRec(Node root, int key) {
24         if (root == null) {
25             root = new Node(key);
26             return root;
27         }
28         if (key < root.key)
29             root.left = insertRec(root.left, key);
30         else if (key > root.key)
31             root.right = insertRec(root.right, key);
32         return root;
33     }
34     // This method mainly calls deleteRec()
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58     int minValue(Node root)
59     {
60         int minv = root.key;
61         while (root.left != null)
62         {
63             minv = root.left.key;
64             root = root.left;
65         }
66         return minv;
67     }

```

```

68 // This method mainly calls InorderRec()
69 void inorder() {
70     inorderRec(root);
71 }
72 //A utility function to do inorder traversal of BST
73 void inorderRec(Node root) {
74     if (root != null) {
75         inorderRec(root.left);
76         System.out.print(root.key + " ");
77         inorderRec(root.right);
78     }
79 }
80
81 // Driver Program to test above functions
82 public static void main(String[] args) {
83
84
85
86
87
88
89
90
91 // print inorder traversal of the BST
92 System.out.println("Inorder traversal of the given tree");
93
94
95 }
96 }

```

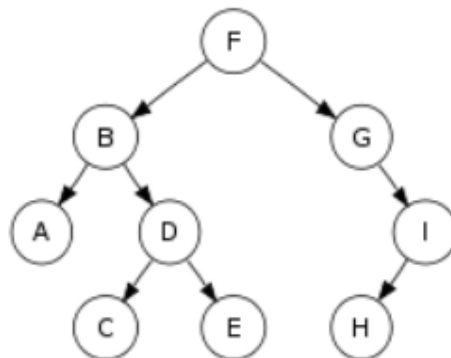
- Write the code to define the class Node (Line 3-11) that containing left and right child of current node and key value of type integer.
- Line 19-33 is a method to \_\_\_\_\_.
- Write the code of delete node method for BST in Line 35-57.
- Line 69-79 is a method to \_\_\_\_\_.
- Write the code to instantiate an object tree in Line 84.
- Complete the code for inserting node with key 15, 8, 6, 12, 21 and 25 in the given space in Line 85-90 and draw the related Binary Search Tree (BST).

- Write the code (Line 93) to delete node with key 12.
- Write the code (Line 94) to traverse object tree using inorder form.
- Run the complete code and state the output.

5. Write the algorithm and code for method inorder, preorder and postorder traversal.  
Refer <https://www.geeksforgeeks.org/tree-traversals-inorder-preorder-and-postorder/>

Traversal	Algorithm	Code
Inorder		<pre>void printInorder(Node node) {  } </pre>
Preorder		<pre>void printPreorder(Node node) {  } </pre>
Postorder		<pre>void printPostorder(Node node) {  } </pre>

6. Determine the result for inorder, preorder and postorder traversal. [Hints: Use the algorithm steps for each traversal]



## Section C:

**Instruction:** Answer all the following questions. You must use any related **Binary Search Tree (BST) methods** for the given problem.

1. Your task is to create an application that can read words in the tree by user input, as well as able to display the output of inorder, preorder and postorder traversal such as the following sample input output.

Input	Output
i go to school by bus	Inorder: bus by go i school to Preorder: i go by bus to school Postorder: bus by go school to i

2. Your task is to create an application that can read 10 integers in the tree and search one integer by user input. If the integer existed in the tree, the output will be display either Found → true, otherwise Not Found → false such as the following sample input output.

Input	Output
7 3 9 2 4 8 10 1 5 6 Search: 2	2 Found --> true
7 3 9 2 4 8 10 1 5 6 Search: 12	12 Not Found --> false

3. Your task is to create an application that can read words in the tree and delete one word by user input. Display the output of inorder traversal form before and after deleting the specific word such as the following sample input output.

Input	Output
please stay at home covid19 Delete: covid19	Inorder: at covid19 home please stay Inorder after deleting covid19: at home please stay