

## EE2410 Data Structure Hw #4 (Chapter 5 Trees)

due date 5/19/2024, 23:59

**Format:** Use MS Word to edit this file by directly typing your student number and name in above blanks and your answer to each homework problem right in the **Sol: blanks** as shown below. Then save your file as **Hw4-SNo.pdf**, where **SNo** is your student number. Submit the **Hw4-SNo.pdf** file via eLearn. The grading will be based on the correctness of your answers to the problems, and the **format requirement**. Fail to comply with the aforementioned format (file name, header, problem, answer, problem, answer,...), will certainly degrade your score. If you have any questions, please feel free to ask. Submit your homework before the deadline (midnight of 5/19 Sun.). Fail to comply (**late homework**) will have **ZERO score**. **Copy** homework will have **ZERO score (both parties)** and **SERIOUS consequences**.

### Trees:

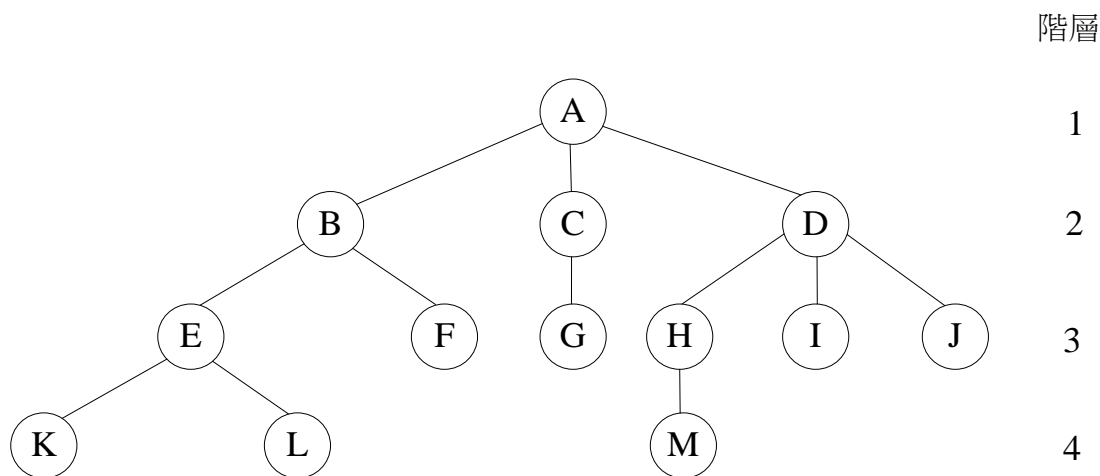
1. (4%) What is the maximum number of nodes in a k-ary tree of height h? Prove your answer.

**Sol:**

$$\max(\text{k\_ary tree node}) = 1 + k + k^2 + \dots + k^{h-1} = \sum_{i=0}^{h-1} k^i = \frac{k^h - 1}{k - 1}$$

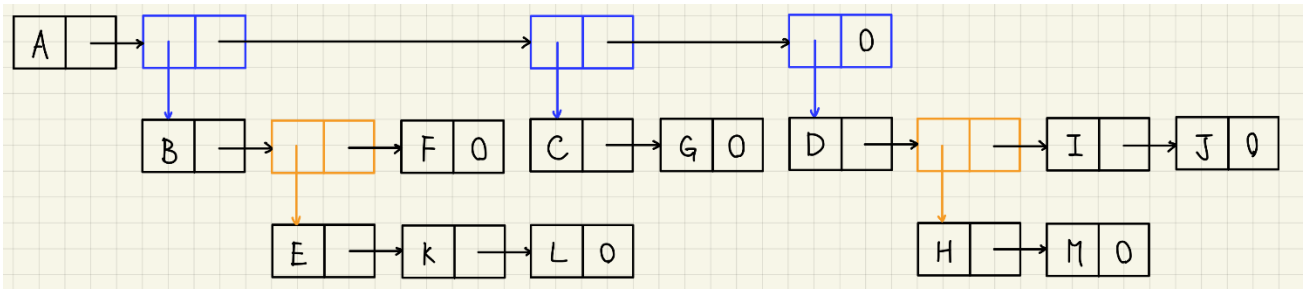
2. (16%) For a simple tree shown below,

- Draw a list representation of this tree using a node structure with three fields: tag, data/down, and next.
- Write down a generalized list expression form for this tree.
- Convert the tree into a left-child and right-sibling tree representation
- Draw a corresponding binary tree for this tree based on (c).
- What is the depth of node L? What is the height of node B? What is the height of the tree?
- Write out the preorder traversal of this tree.
- Write out the postorder traversal of this tree.
- Write out the level order traversal of this tree.



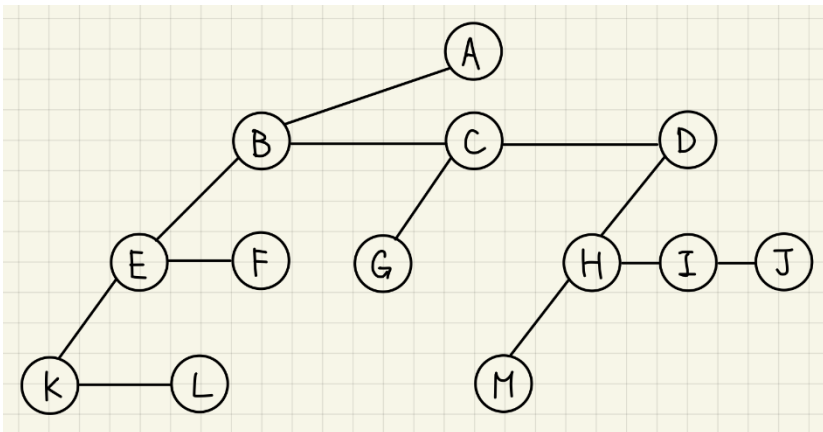
**Sol:**

- (a)

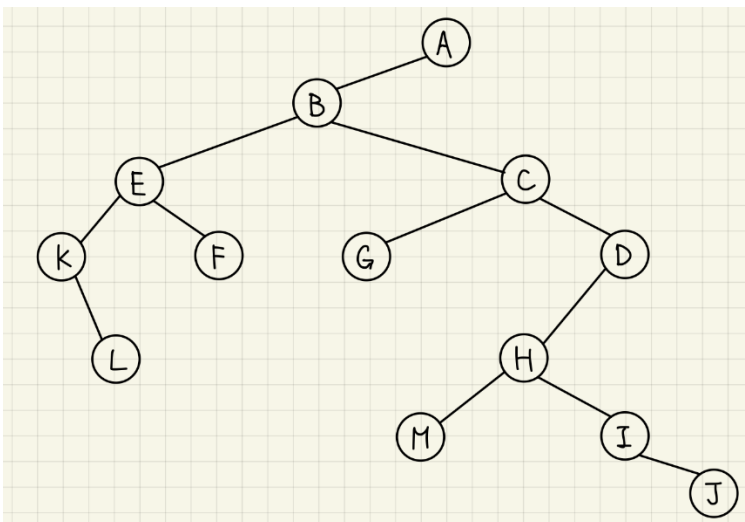


(b)  $(A(B(E(K,L),F),C(G),D(H(M),I,J)))$

(c)



(d)



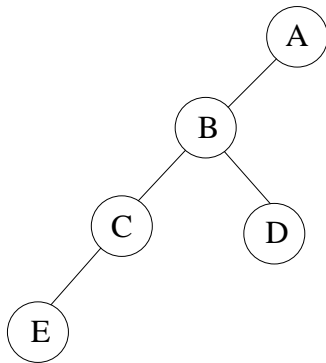
(e) depth of node L = 4; height of node B = 3; height of the tree = 4;

(f) preorder traversal: A B E K L F C G D H M I J

(g) postorder traversal: K L E F B G C M H I J D A

(h) level order traversal: A B C D E F G H I J K L M

3. (10%) Draw the internal memory representation of the binary tree below using (a) sequential and (b) linked representations.

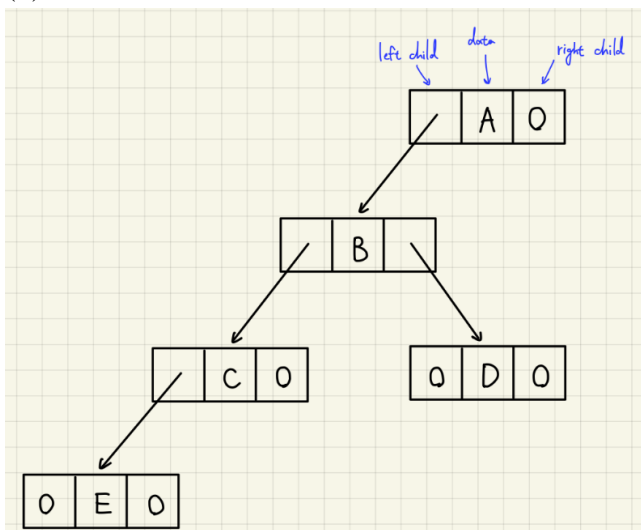


**Sol:**

(a)

Array	a[0]	a[1]	a[2]	a[3]	a[4]	a[5]	a[6]	a[7]	a[8]	a[9] ~ a[15]
node	x	A	B	x	C	D	x	x	E	x

(b)

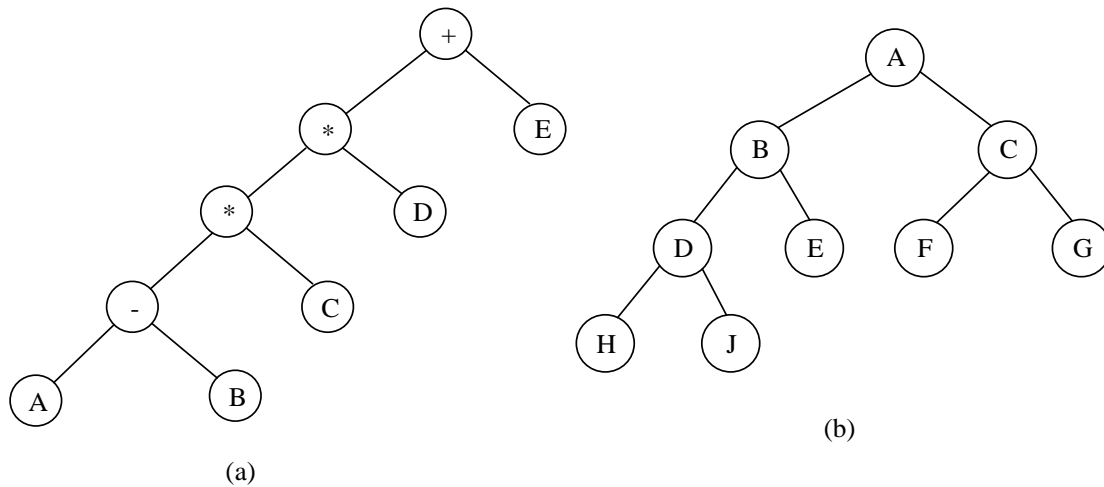


4. (4%) Extend the array representation of a complete binary tree to the case of complete trees whose degree is  $d$ ,  $d > 1$ . Develop formulas for the parent and children of the node stored in position  $i$  of the array.

**Sol:**

For a node stored in position  $i$ , its children are in the position  $\{id - (d - 2), id - (d - 1), \dots, id, id + 1\}$ ; and its parent is in the position  $\left\lfloor \frac{i+d-2}{d} \right\rfloor$ .

5. (16%) Write out the inorder, preorder, postorder, and levelorder traversals for the following binary trees.



**Sol:**

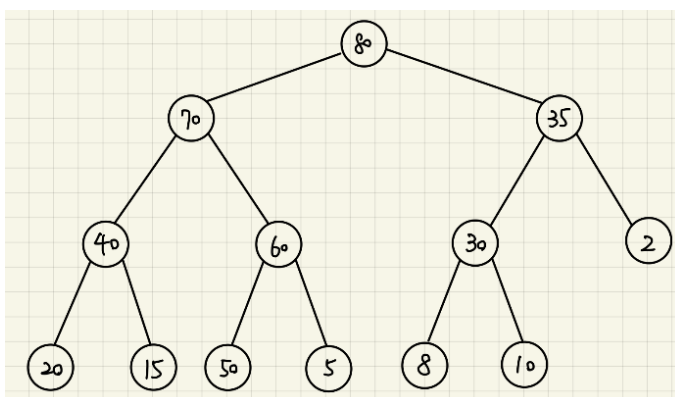
	Inorder	Preorder	Postorder	Level order
(a)	$A - B * C * D + E$	$+ * * - A B C D E$	$A B - C * D * E +$	$+ * E * D - C A B$
(b)	H D J B E A F C G	A B D H J E C F G	H J D E B F G C A	A B C D E F G H J

6. (10%) Given a sequence of 13 integer number: 50, 5, 30, 40, 80, 35, 2, 20, 15, 60, 70, 8, 10.

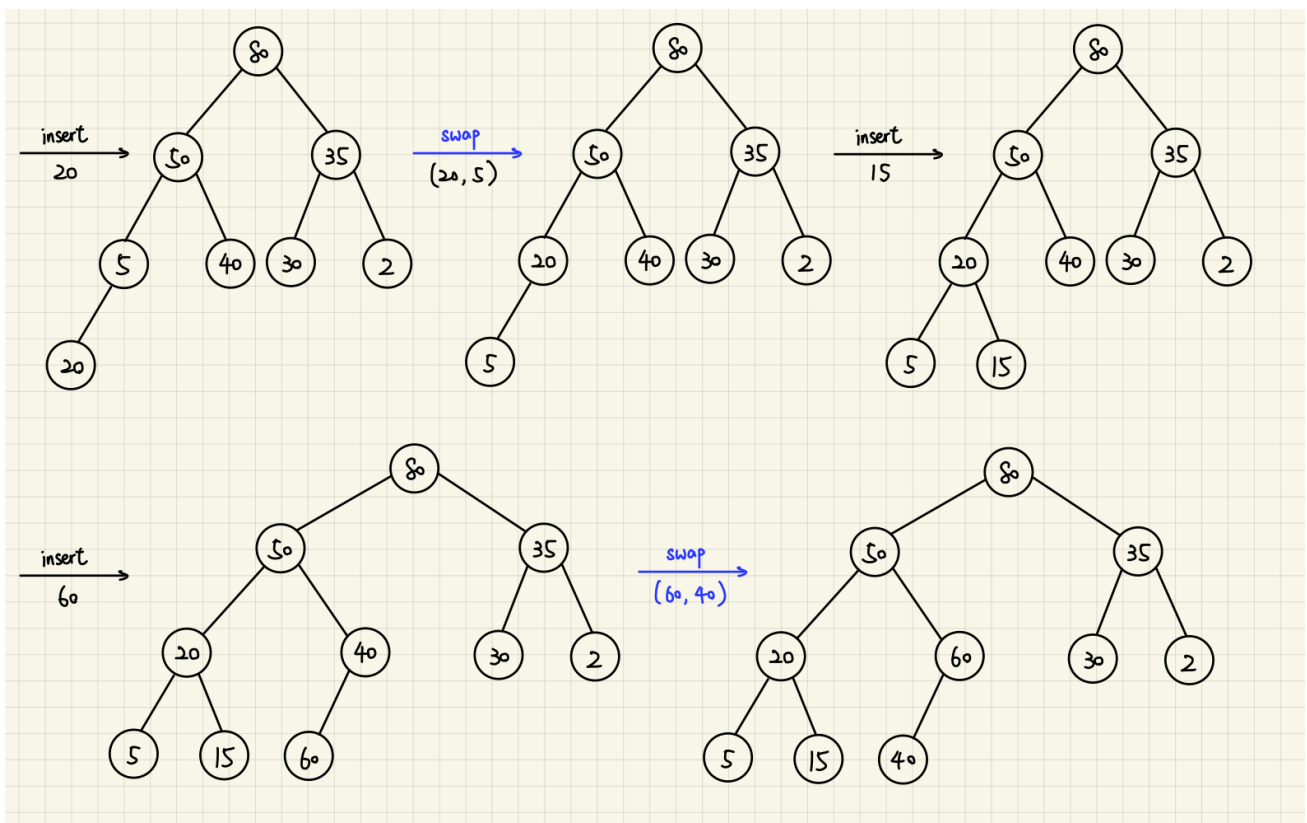
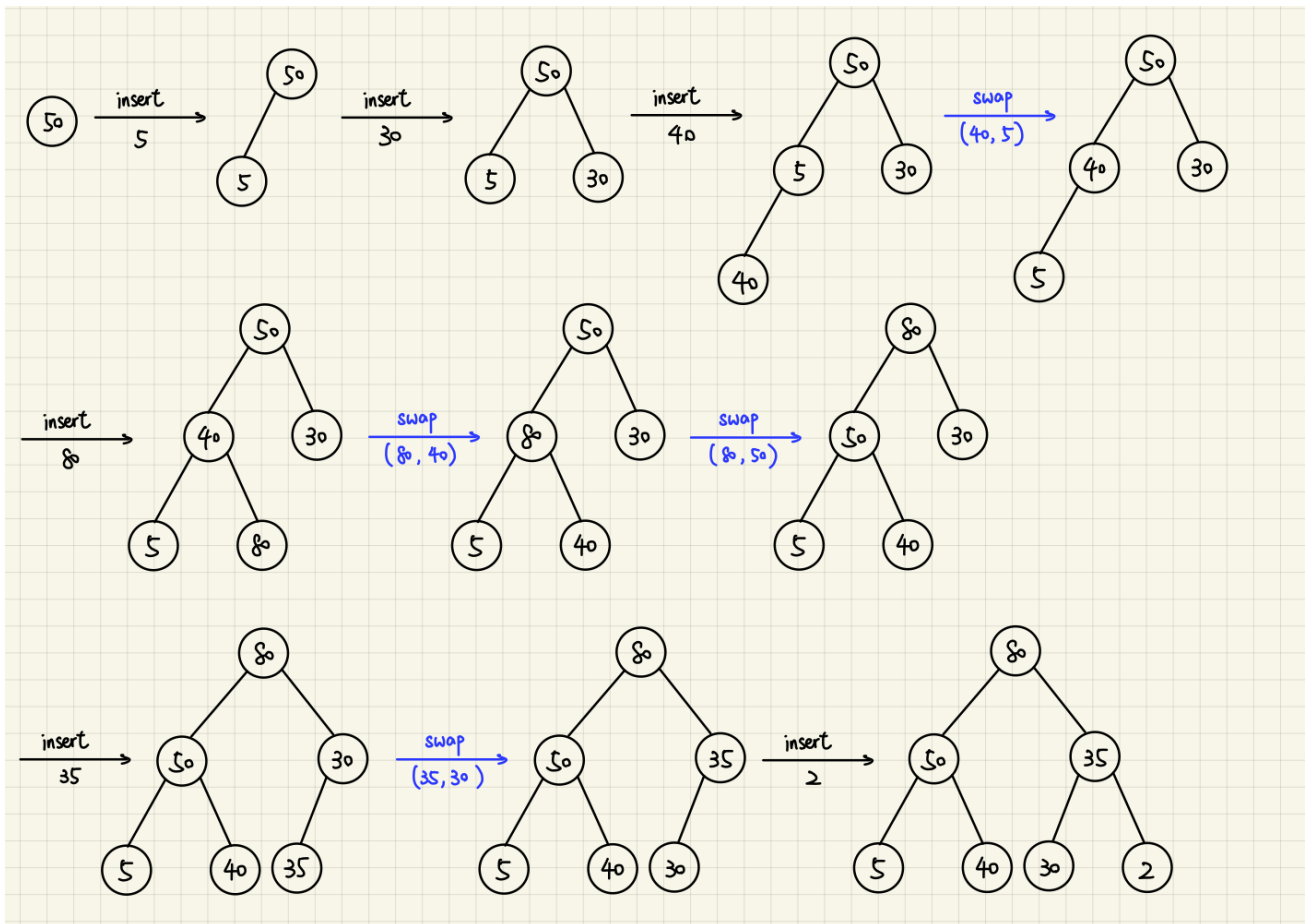
- (a) Assume a **max heap** tree is **initialize** with these 13 numbers placed into nodes of the tree according to node numbering of complete binary tree by using the **bottom up heap construction initialization** process. Please draw the final Max heap tree after initialization process.
- (b) Construct a max heap by **inserting** the given 13 numbers one by one according to the sequence order into an initially empty max heap tree, instead of bottom up heap construction.
- (c) Based on (a) what is the result heap after two removeMax operations?

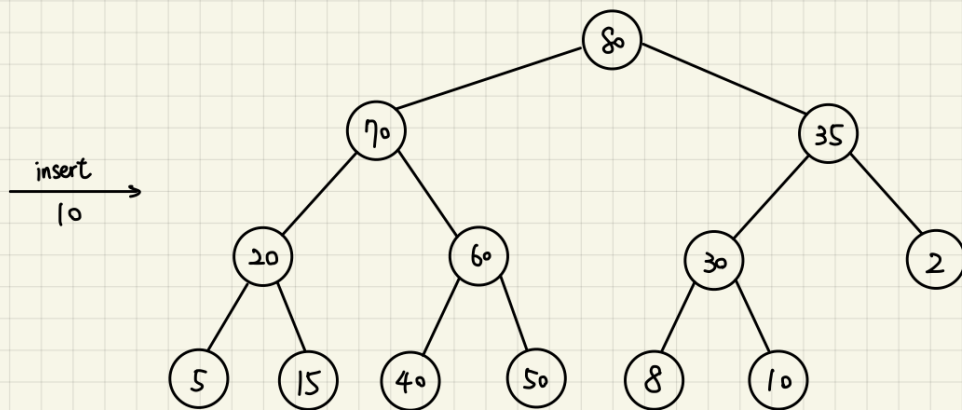
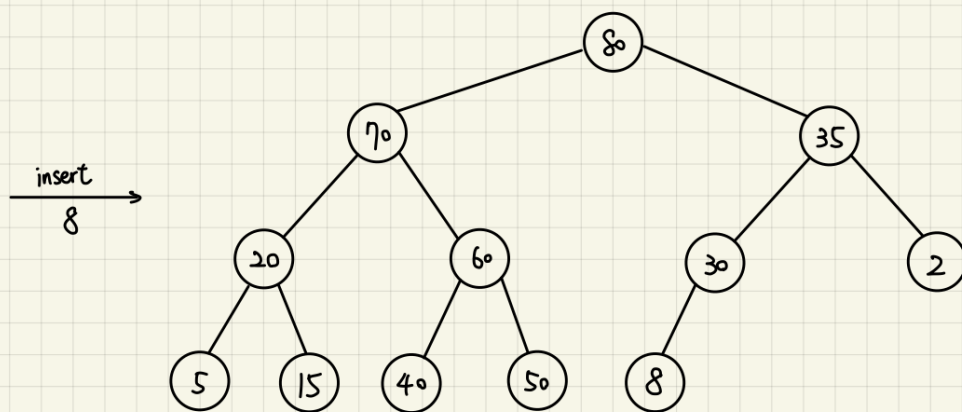
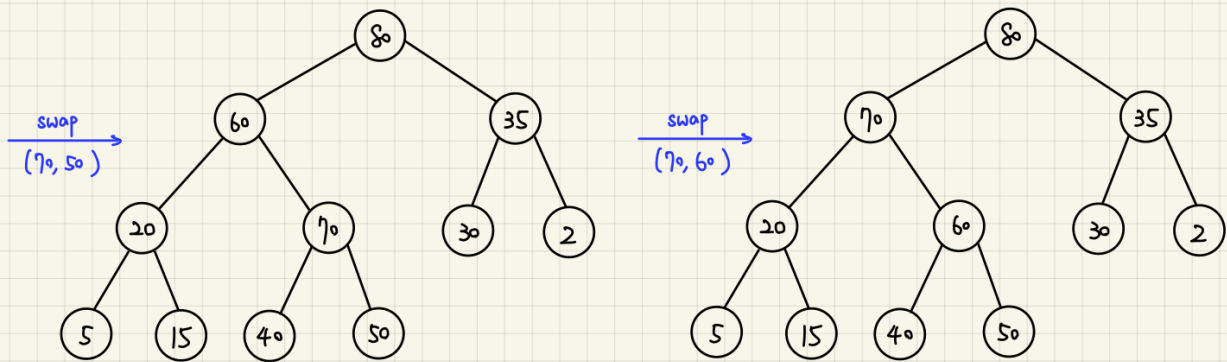
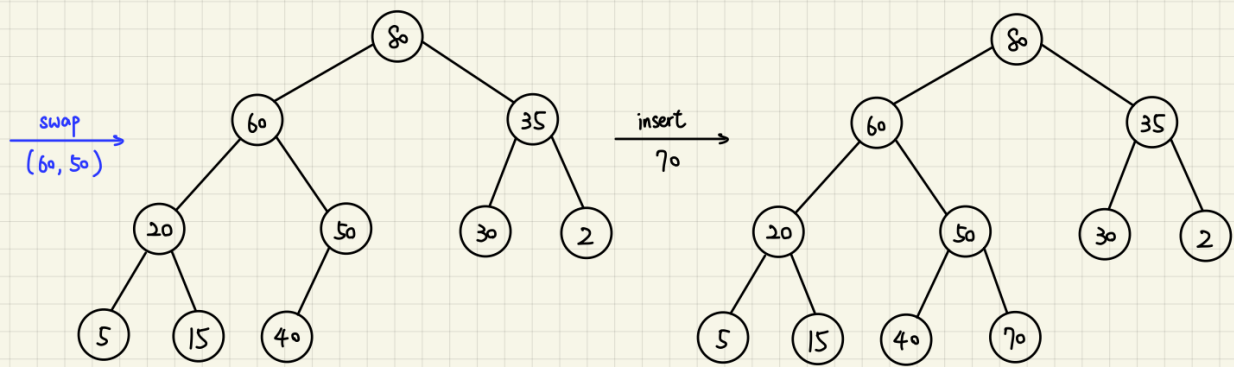
**Sol:**

(a)

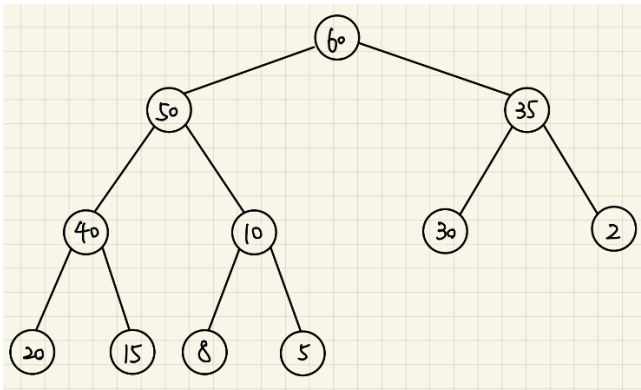


(b)





(c)



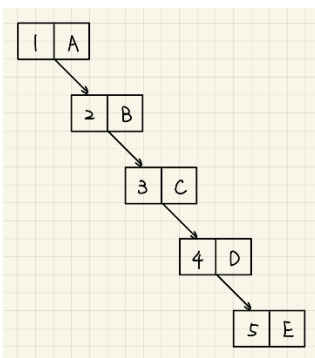
## 7. (10%) Binary Search Tree

- How many different binary search trees can store the keys {1,2,3}?
- If we insert the entries (1,A), (2,B), (3,C), (4,D), and (5,E), where the number denotes the key value of the node, in this order, into an initially empty binary search tree, what will it look like? Please draw this BST.
- John claims that the order in which a fixed set of entries is inserted into a binary search tree does not matter—the same tree results every time. Give a small example that proves he is wrong.
- Given a sequence of 13 integer number: 50, 5, 30, 40, 80, 35, 2, 20, 15, 60, 70, 8, 10, use the BST Insert function (manually) to insert the 13 numbers sequentially to construct a binary search tree. Draw the final 13-node BST.
- A binary search tree produces the following preorder traversal, where “null” indicates an empty subtree (i.e. the left/right child is the null pointer).  
9,5,3,1,null,null,4,null,null,8,6,null,null,null,20,12,10,null,11,null,null,null,30,21,null,null,31,null,null  
Draw the tree that produced this preorder traversal.

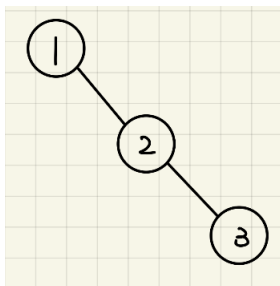
**Sol:**

(a) number of different binary tree  $= \frac{1}{n+1} C_n^{2n} |_{n=3} = \frac{1}{3+1} C_3^6 = 5$

(b)

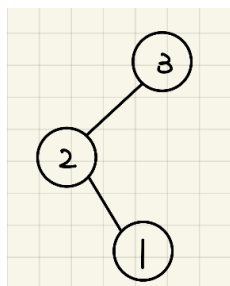


(c)



Insertion order:

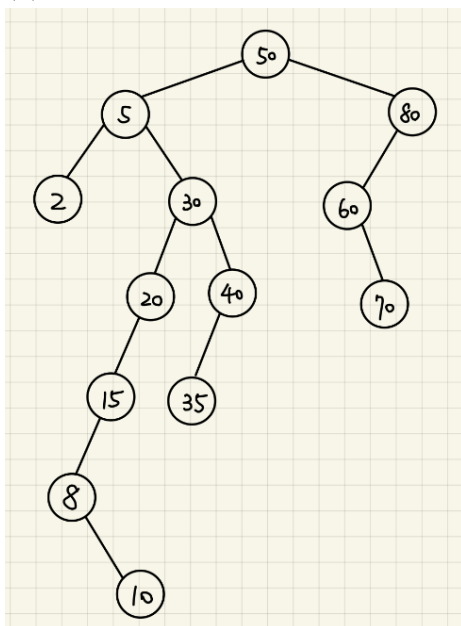
1→2→3



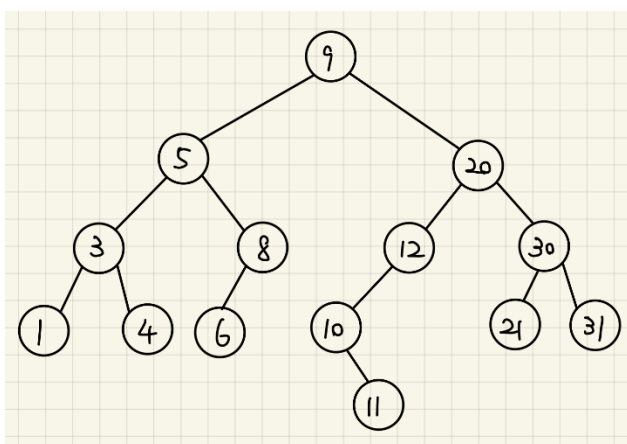
Insertion order:

3→2→1

(d)

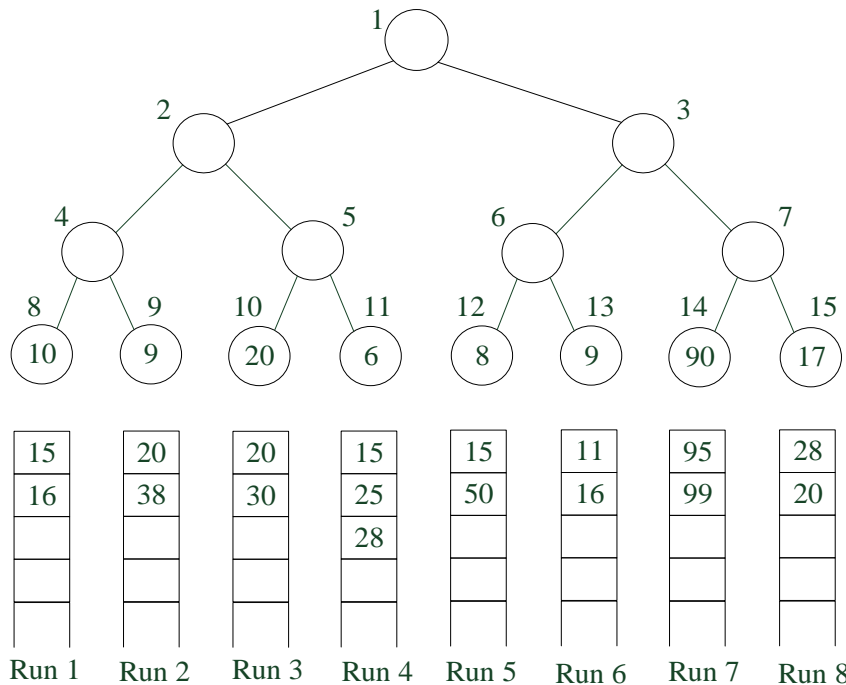


(e)





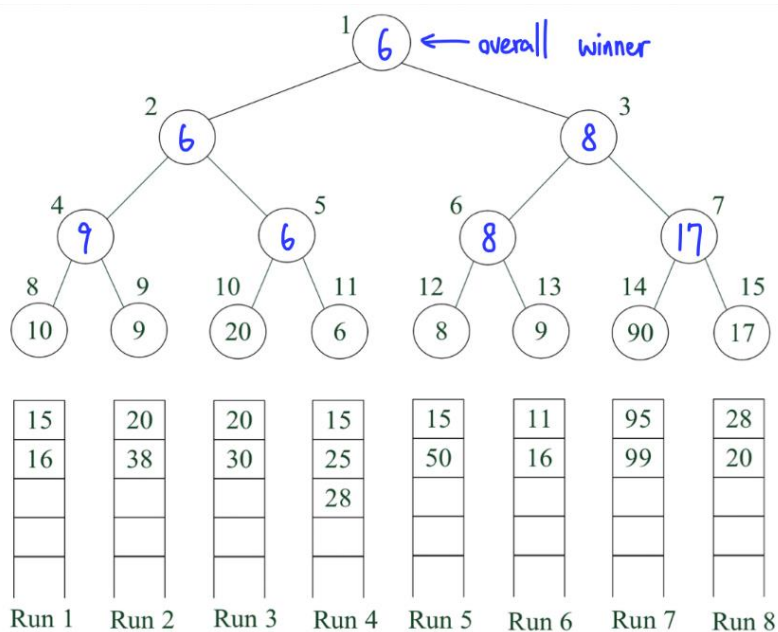
8. (10%) An 8-run with total of 25 numbers are to be merged using Winner tree and Loser tree, respectively. The numbers of the 8 runs are shown below. The first numbers from each of the 8 runs have been placed in the leaf nodes of the tree as shown. Then these eight numbers enter the tournament to get the overall winner.



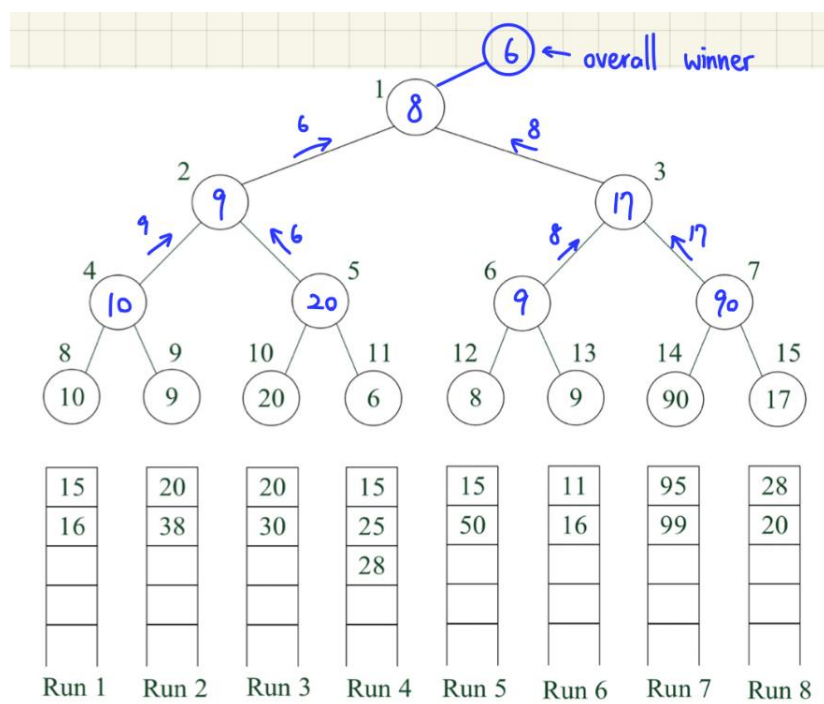
- (a) Draw the winner tree and indicate the overall winner of **this tournament**.  
 (b) Draw the loser tree and indicate (draw) the overall winner of **this tournament**.

**Sol:**

(a)



(b)



9. (5%)

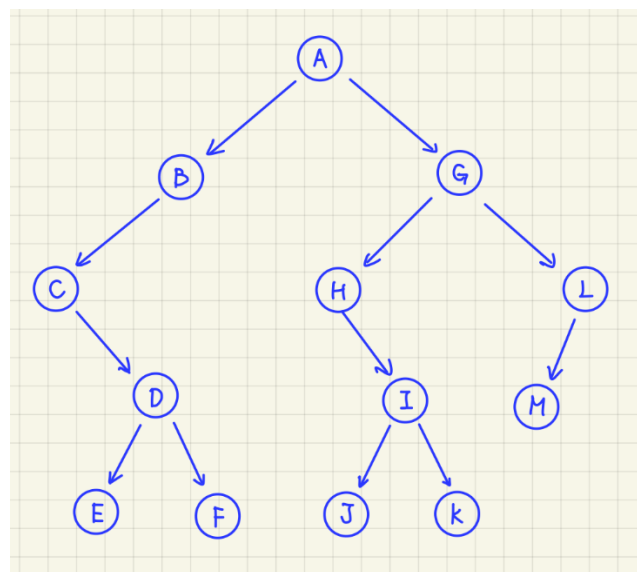
The nodes in a binary tree in preorder and inorder sequences are as follows:

preorder: ABCDEFGHIJKLM

inorder: CEDFBAHJIKGML

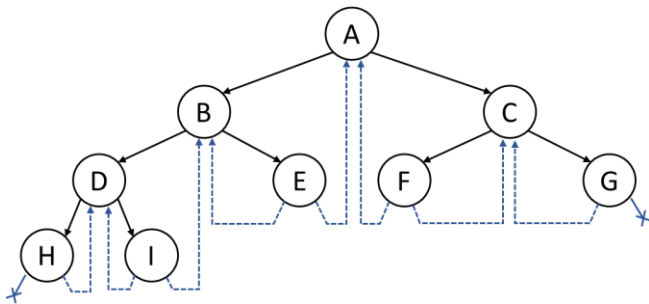
Draw the binary tree.

**Sol:**



10. (15%)

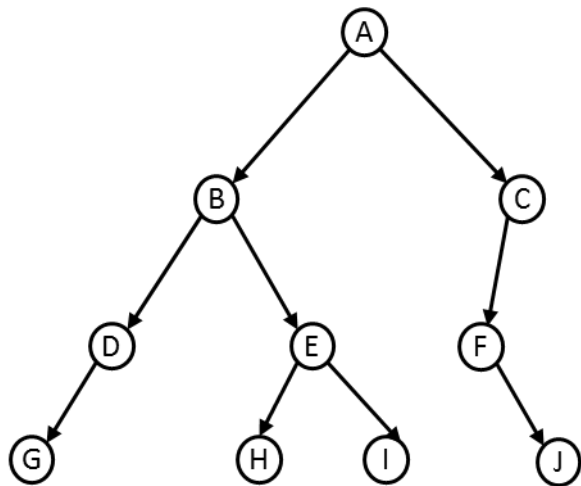
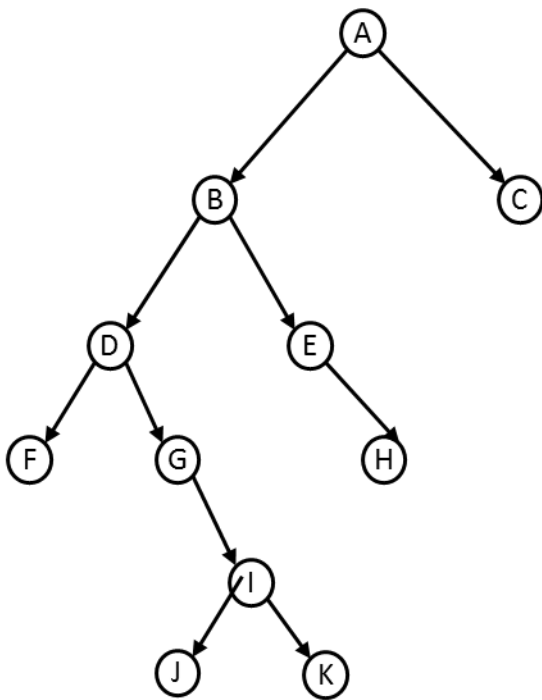
The tree with 9 nodes shown below includes threads linking predecessors and successors according to the inorder traversal.



(a) Redraw the tree with threads linking predecessors and successors according to the postorder traversal.

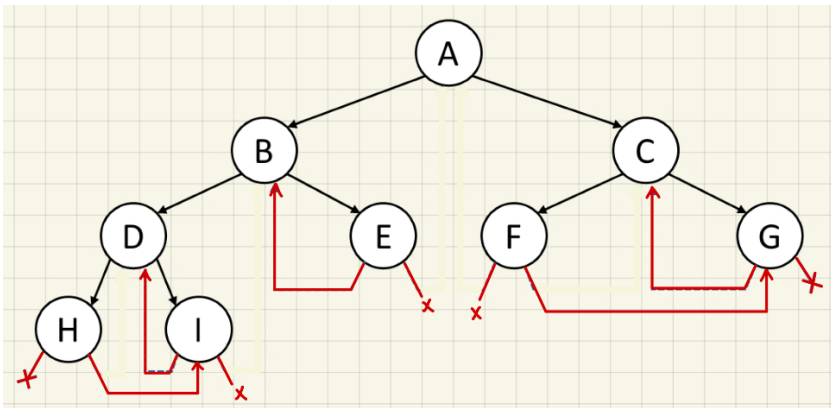
(b) Are these postorder threads adequate to perform threaded preorder, inorder, and postorder traversals? Please explain your answers.

(c) Redo (a) and (b) for the two trees below.



**Sol:**

(a)



(b)

None of the threads is adequate to perform threaded preorder, inorder, and postorder traversals since there's no proper way for some nodes to move back to their parent or ancestors.

For example:

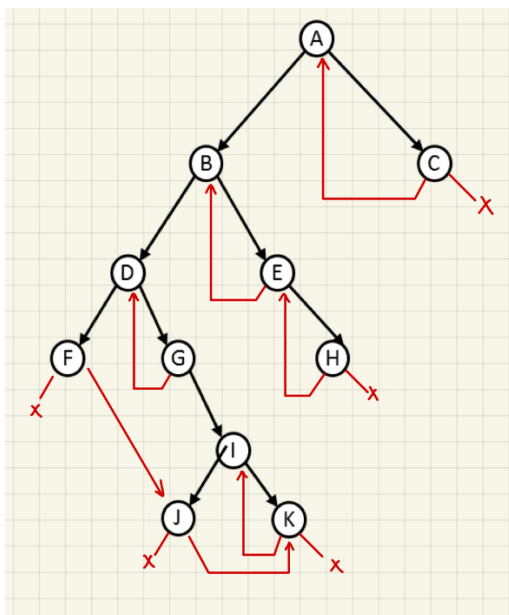
To a threaded preorder traversal, after traveling through ABDHI, we cannot find a proper link field to move up to B, where we can get access to E, the next preorder node.

To a threaded inorder traversal, when starting from H, we cannot find a proper link field to travel to D, which is the parent of H and the next inorder node.

To a threaded postorder traversal, after traveling through HID, we cannot find a proper link field to move up to B, where we can get access to E, the next postorder node.

(c)

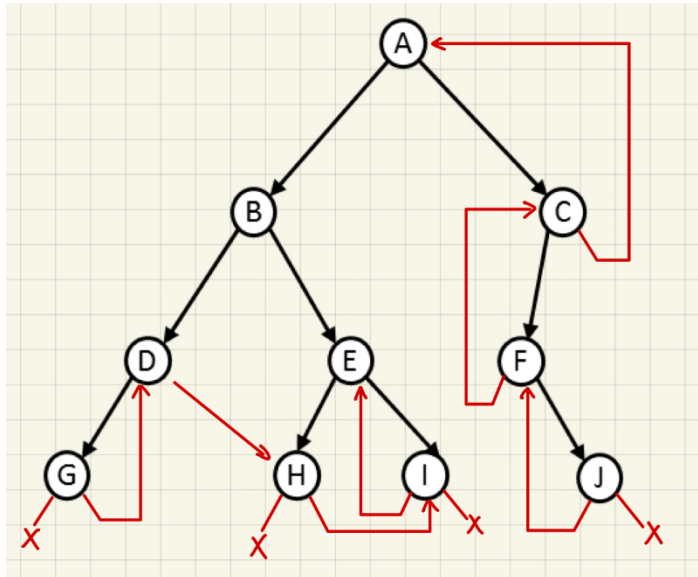
1a)



1b)

Same reason as 10.(b)

2a)



2b)

Same reason as 10.(b)