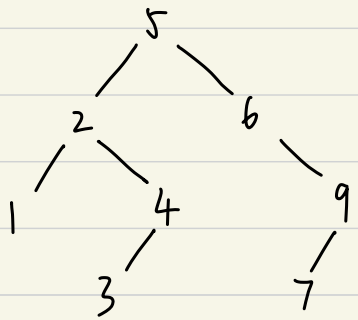
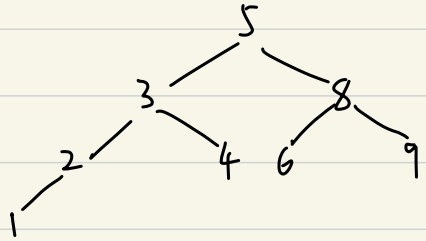
Assignment 2

Problem 1

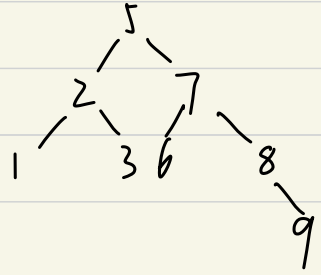
1. $S = \{2, 6, 4, 9, 7, 3, 1\}$



2. $S = \{8, 3, 9, 2, 6, 1, 4\}$



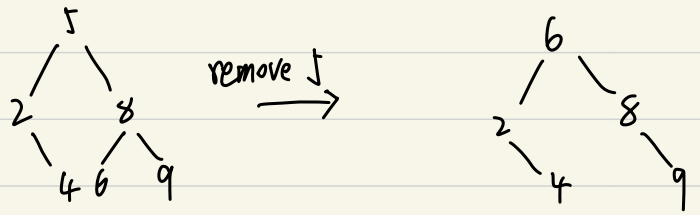
3. $S = \{7, 2, 1, 8, 3, 6, 9\}$



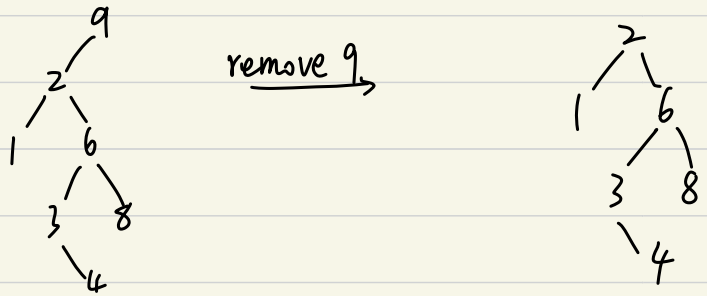
Problem 2

A is NOT a BST.

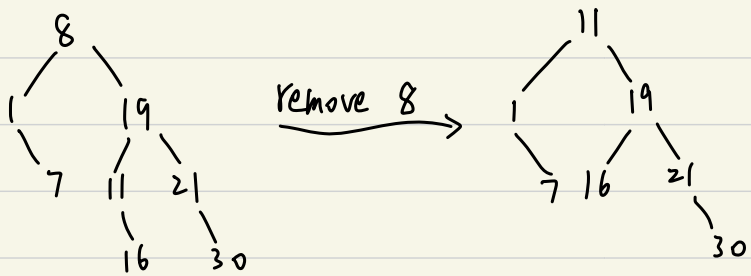
B.



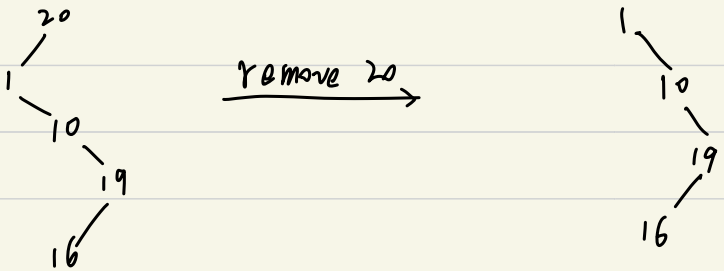
C.



D.



E.



Problem 3

2 is true.

Problem 4

1. We can record tree structure first. Then we traverse all the elements in the tree and choose a sorting algorithm applying on them. In the end, put the sorted element back to the tree with in-order traversal.

Time complexity: O(nlogn)

best approach: O(n)

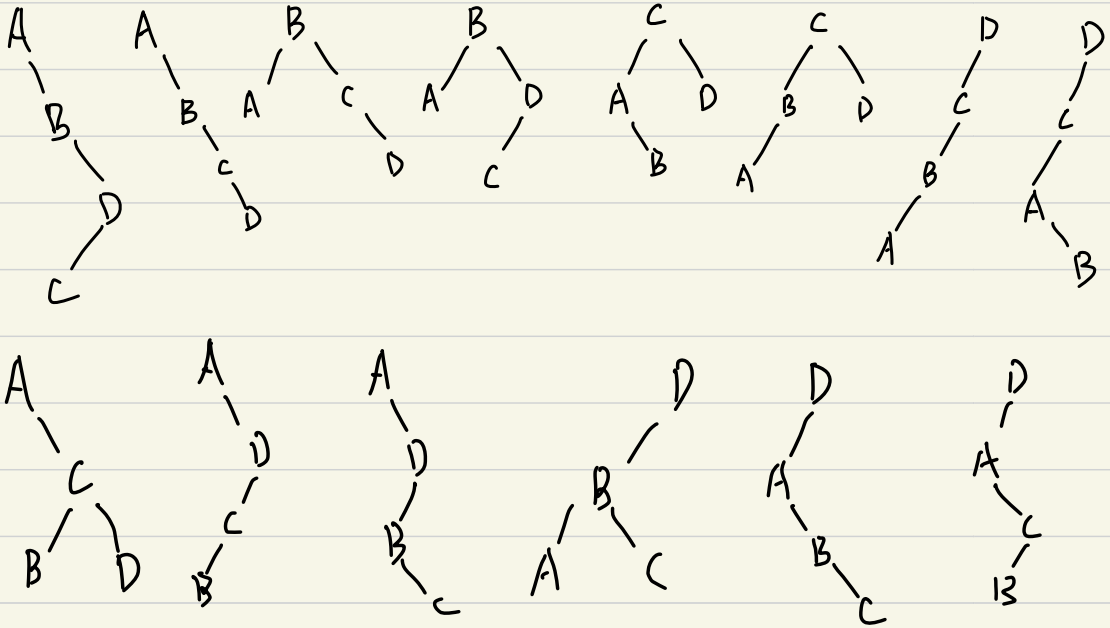
pseudo code:

```
def inorder(node):
    inorder(node.leftchild)
    array.append(node)
    inorder(node.rightchild)
inorder(root)
if isincreasing(array)==True: return 0
sorted_array = merge sort (array) // use merge sort algorithm
```

```
def construct_tree(node, i)
    construct_tree(node.leftchild)
    node = array[i]
    i = i+1
    construct_tree(node.rightchild)
construct_tree(root, 0)
```

Problem 5

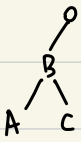
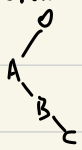
Suppose 4 keys are: A < B < C < D



14 BSTs.

Problem 6

1. True. In-order traversal prints out left child node before root node and then right child node for each layer. For a BST, left child node < root node < right child node. The sequence is always in increasing order.

2. False. It's NOT sufficient to build a BST when only be given one traversal result. We can't tell the left child from the right child. It could be  or  for example.

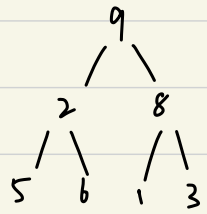
Problem 7

Time Complexity: O(nlogn)
Best approach: O(n)

1. Downheap for each node

{9, 2, 8, 5, 6, 1, 3}

0 1 2 3 4 5 6



pseudocode:

```
def minheap(heap, i):
    if heap[left]<heap[i] and left < length(heap):
        s = left    else: s = i
    if heap[right] < heap[s] and right < length(heap):
        s = right
    if  s != i :
        swap(heap[s], heap[i])
        minheap(heap, s)
heap = [9, 2, 8, 5, 6, 1, 3]
n = length(heap)//2 -1
for i in range(n, -1, -1):
    minheap(heap, i)
```

test:

9 2 (8) 5 6 (1) (3)

9 (2) 1 (5) (6) 8 3

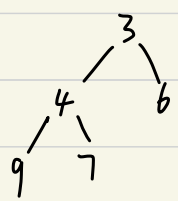
(9) (2) (1) 5 6 8 3

1 2 (9) 5 6 (8) (3)

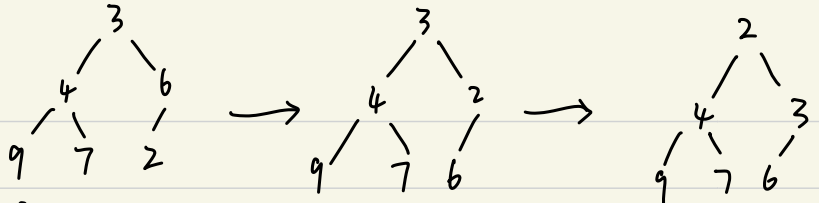
1 2 3 5 6 8 9

Problem 8

{null, 3, 4, 6, 9, 7}



Add 2 :



{null, 2, 4, 3, 9, 7, 6}

Problem 9

1. { 9, 12, 14, 3, 4, 21, 18 }
2. { 9, 14, 4, 18, 12, 3, 21 }
3. { 12, 14, 3, 9, 4, 18, 21 }
4. { 12, 3, 14, 18, 4, 9, 21 }
5. { 12, 9, 18, 3, 14, 21, 4 }

mod 9

	0	1	2	3	4	5	6	7	8	
1.	9	18		12	3	14	4	21		√
2.	9	18		12	4	14	3	21		X
3.	9	18		12	3	14	4	21		√
4.	18	9		12	3	14	4	21		X
5.	9	18		12	3	14	21	4		X
Given	9	18		12	3	14	4	21		

Answer : 1, 3

Problem 10

{12, 44, 13, 88, 23, 94, 11, 39, 20, 16}

29, 93, 31, 181, 51 193 27 83 45 37

$h(k) = 2k+5 \mod 11$

0	1	2	3	4	5	6	7	8	9	10
	20			16	11	94	23		13	
					88	39	12			
					44					

Problem 11

{12, 44, 13, 88, 23, 94, 11, 39, 20}

$h(k,i) = (h_0(k) + i + i^2) \mod 11$

where $h_0(k) = k \mod 11$

0	1	2	3	4	5	6	7	8	9	10
44	12	13	23	20		88	39	94	11	