

**Prince Sultan University**  
**CCIS - Department of Computer Science**

**Major Exam 2**  
**Term 232**

**Course Title: Data Structures and Algorithms**

**Course Code: CS 210**

**Exam Date: 29/04/2024**

**Exam Time: 50 minutes**

**Student Name:**

**Student ID:**

**Serial # :**

**Section #:**

Question Number	CLO	Question points	Score
Question 1	CLO 1	5	
Question 2	CLO 2	4	
Question 3	CLO 4	6	
<b>Total out of</b>		<b>15</b>	

**Instructions:**

- This exam contains three questions with multiple parts.
- **Time allowed: 50 minutes.**
- Closed Book, Closed Notes.
- Use of Calculators and / or computing devices / smartphones etc is strictly prohibited.
- Answer the problems on the exam sheets only. No additional attachments would be accepted.
- If you need extra space use the back of a page.
- When the “time is over” is called, it is students’ responsibility to submit his exam to the invigilator. Submitting completed exam 3 minutes after the “time is over” will incur a penalty of **5 points**.
- Do **NOT** use the erasable pens

**Few gentle reminders:**

- If you get stuck on some problem for a long time, move on to the next one.
- The ordering of the problems is somewhat related to their relative difficulty. However, the order might be different for you!
- You should be better off by first reading all questions and answering them in the order of what you think is the easiest to the hardest problem.
- Keep the points distribution in mind when deciding how much time to spend on each problem.

**Question 1**

[ / 5 points - CLO 1]

Note: In case you forgot your ABCs! use this to help you determine the order.



**Part A:**

( / 3 points)

Sort the following array using **Quicksort**. Trace all the compare and swap operations. Assume that pivot is always selected as the **FIRST** element of the array.

H	Θ	N	E	Y	€	O	M	B
E	B	C	H	Y	N	O	M	O
E	B	€		Y	N	O	M	Θ
C	B	E		O	N	O	M	Y
C	B			O	N	Θ	M	
B	C			M	N	O	O	
B				M	N			
B	C	E	H	M	N	O	O	Y

**Part B:**

( / 2 points)

Answer the following giving appropriate Big-Oh for each:

Statement	Big-oh Characterization
Number of swap operations using Selection Sort in a pre-sorted array.	O(1)
Given the following sequence: {22, 31, 42, 53, 53, 42, 31, 22}, what is the run time to sort the sequence using Insertion sort.	O(n <sup>2</sup> )
The number of comparisons made by Merge-sort algorithm for a sequence made of random values.	O(nlogn)
The number of comparisons made by Bottom-up Merge-sort algorithm for a sequence which is already sorted.	O(nlogn)

## Question 2

[ / 4 points - CLO 2]

### Part A:

( / 3 points)

Write a Java method called `isPalindrome( String word )` that checks whether a given string word is a palindrome using a stack and a queue.

A *palindrome* is a word that reads the same backward or forward. For example, the following words are palindrome: *Radar, Level, Noon and kayak*

Your method will split the characters of the String and add them one by one into the stack and into the queue. Once this is done, it will compare their content. It returns true if it is a palindrome, otherwise, it returns false.

You may use the following to convert the String into an array of characters:

```
String word = "radar";
```

```
char[] characters = word.toCharArray();
```

The array will have:

r	a	d	a	r
---	---	---	---	---

```
public boolean isPalindrome(String word) {  
    Stack S = new Stack();  
    Queue Q = new Queue();  
    for(int i=0;i<word.length; i++)  
    {  
        S.push(word.charAt(i));  
        Q.enqueue(word.charAt(i));  
    }  
    while(!S.empty())  
    {  
        if(S.pop() != Q.dequeue())  
            return false;  
    }  
    return true;  
}
```

The runtime for this is  $n + n = 2n$ , i.e.  $O(n)$

**Part B:** ( / 1 point)

What is the total cost of running your method given an input string of size  $n$ ?

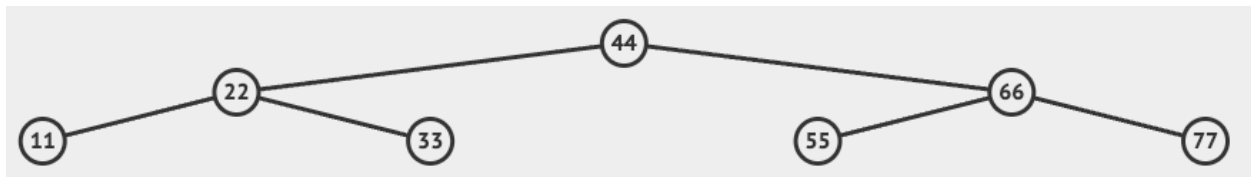
The runtime for this is  $n + n = 2n$ , i.e.  $O(n)$

**Question 3** [ / 6 points - CLO 4]

**Part A:** ( / 1 point)

An unknown AVL tree is given to you. We know that this tree is a complete tree. We also know that the POST-order traversal for this tree gives **11, 33, 22, 55, 77, 66, 44**.

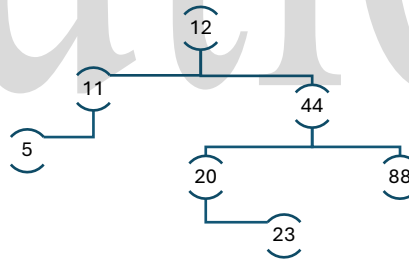
Draw the tree.



**Part B:** ( 5 / points)

**Question:**

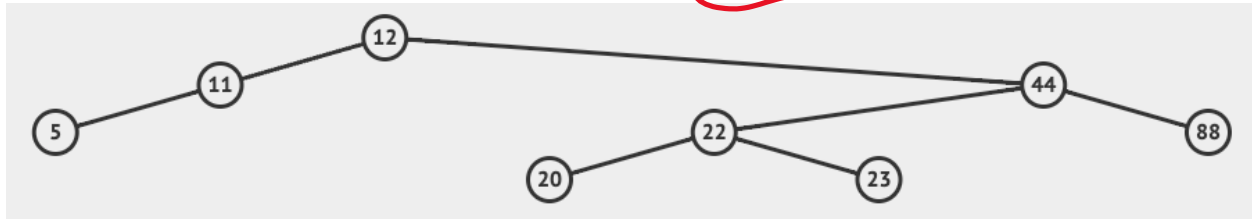
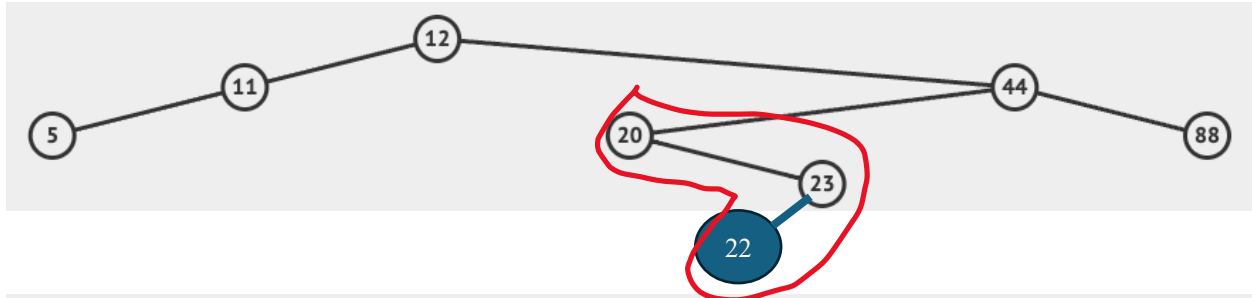
Consider the following AVL tree:



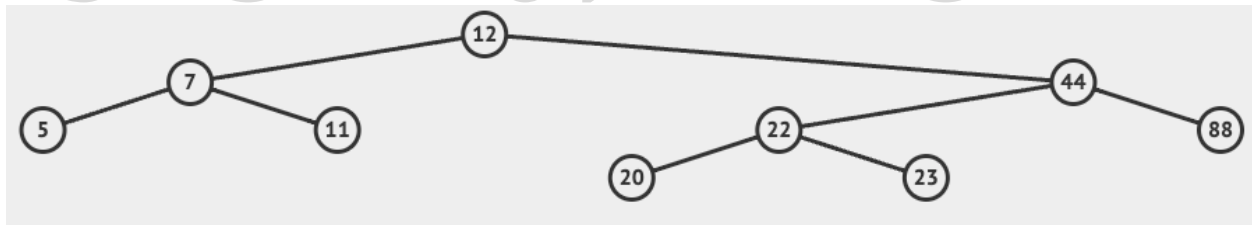
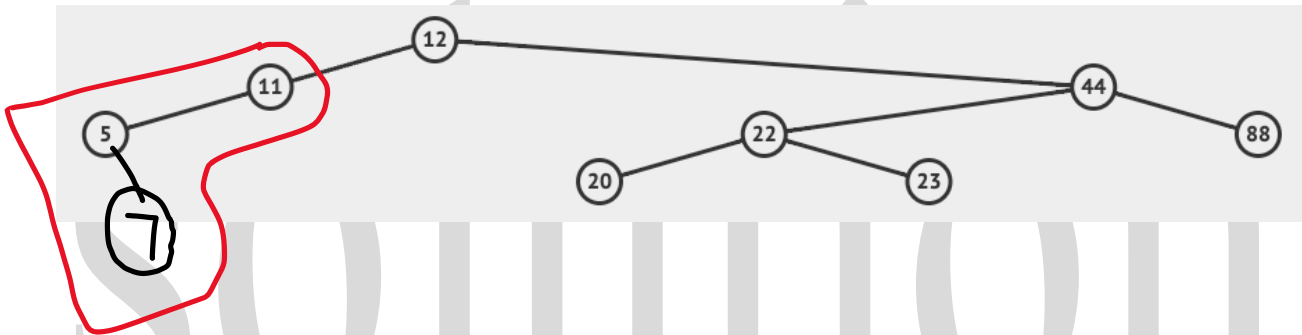
**A.** Give the array representation of the above tree. ( / 1 point)

0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
	12	11	44	5		20	88						23		

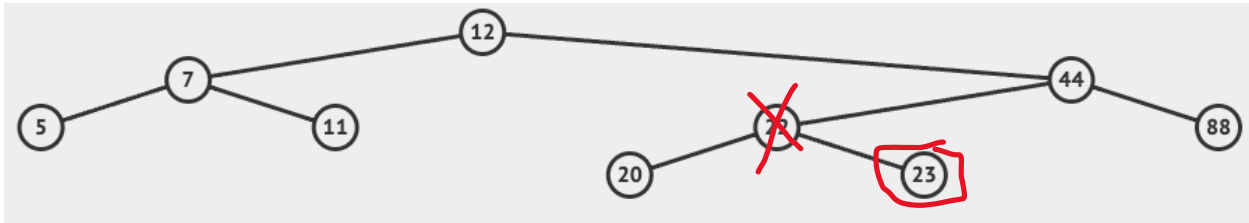
**B.** Insert node 22 to the tree in the previous page. Do the appropriate rotation. ( /1 point)



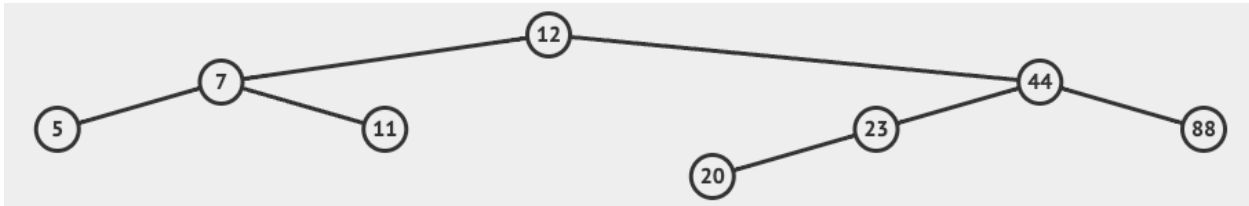
C. On the tree resulted from Step B, Insert node 7. Do the appropriate rotation. ( /1 point)



D. On the tree resulted from Step C, delete 22 and indicate which deletion case. ( /1 point)

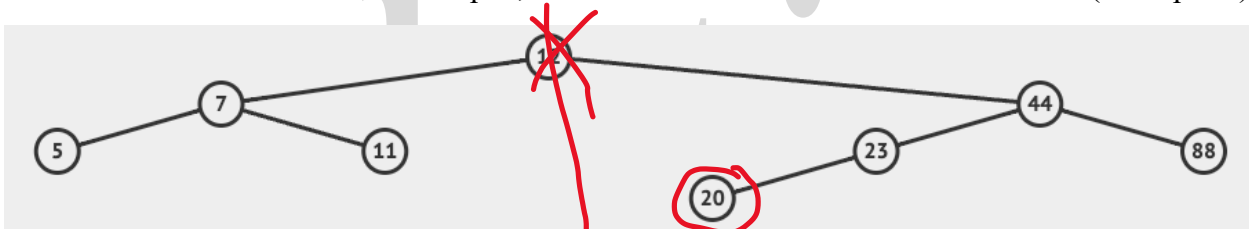


Case 2: Candidate is 23

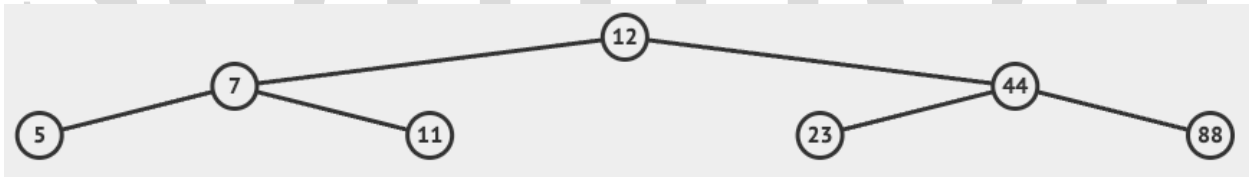


Is AVL : no need to rotate

E. On the tree resulted from Step D, delete 12 and indicate which deletion case. ( /1 point)



Case 2: candidate 20.



\*\*\*\*\* End of Exam \*\*\*\*\*