Homework M12: Trees

Due May 9 at 11:59pm **Points** 100 **Questions** 32
Available until May 30 at 11:59pm **Time Limit** None

Instructions

Review the **Homework FAQ** page for details about submitting homework. In this homework, you will:

- evaluate characteristics of trees
- traverse trees
- create and modify trees
- evaluate trees and heaps
- create and modify heaps
- create balanced trees (B-Trees)

Notes

Be careful about the type of tree!

- Read each question carefully to know what type of tree the question asks about.
  - Examples: a general tree, binary tree, a binary search tree (BST), a heap, a balanced tree, etc.
- Read carefully what is required in each answer (the tree, array, etc.). If you're not sure, post to the discussion board!

Image File Submission

- For the questions that ask you to draw a tree, you can create your answer in whatever way is easiest, but only submit an image file, word processing file, or PDF.
  - **Accepted file types: gif, jpg, jpeg, png, doc, docx, rtf, and pdf**
  - Do **not** submit a .txt, .xml, or .heic file
  - You can draw the answer on paper and then scan the answer or take a picture of it with a phone- however you can capture it is fine!
  - Just make sure the picture is clear enough for us to grade.
- If you want to submit digital answers, I recommend a free program called draw io: **https://www.draw.io/ (https://www.draw.io/)** .
  - This is the program I use to create the trees for the online notes. I find it very user-friendly.
  - If you do use this site, be sure to submit an exported image file or pdf of your results. Do **not** submit a .xml file.

This quiz was locked May 30 at 11:59pm.

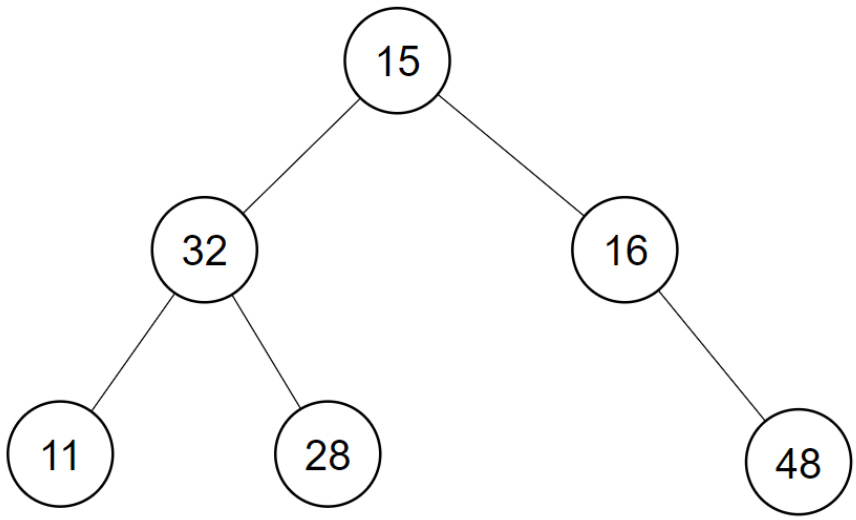
Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	7,085 minutes	108 out of 100

Score for this quiz: **108** out of 100
Submitted May 4 at 12:56pm
This attempt took 7,085 minutes.

File Upload and Short Answer Questions- Trees Part One

Use the binary tree below for the next three questions.



Question 1

3 / 3 pts

What is the **preorder** traversal of the binary tree?

To list the traversal, list each value visited separated by a space (example: 18 22 19 24 ...). Include no other characters.

Correct!

15 32 11 28 16 48

Correct Answers

15, 32, 11, 28, 16, 48

15 32 11 28 16 48

15,32,11,28,16,48

Question 2

3 / 3 pts

What is the **postorder** traversal of the binary tree?

To list the traversal, list each value visited separated by a space (example: 18 22 19 24 ...). Include no other characters.

Correct!

11 28 32 48 16 15

Correct Answers

11, 28, 32, 48, 16, 15

11 28 32 48 16 15

11,28,32,48,16,15

Question 3

3 / 3 pts

What is the **inorder** traversal of the binary tree?

To list the traversal, list each value visited separated by a space (example: 18 22 19 24 ...). Include no other characters.

Correct!

11 32 28 15 16 48

orrect Answers

11 32 28 15 16 48

11, 32, 28, 15, 16, 48

11,32,28,15,16,48

File Upload Questions- Trees Part Two

Question 4

8 / 8 pts

Draw the **binary search tree** that results from adding the following numbers **in this order.**

55, 63, 23, 28, 11, 32, 60, 66, 25, 62

java111c_question 4.PNG (https://ccsf.instructure.com/files/7777192/download)

Question 5

1 / 1 pts

Is the **binary search tree** you drew above unique?

In other words, is this the **only possible** binary search tree that could be created for the numbers inserted in this order?

Correct!

- (●) Yes
- ( ) No

Question 6

8 / 8 pts

Trace the creation of a **maxheap** from the numbers below.

18, 24, 26, 32, 39

Use the **upheap approach** described in the textbook (section "Using Add"), which is the first approach described in the lecture.

In this approach, you reheap after **each** addition.

In your answer, show:

1. what the tree looks like when each node is added (before the reheap)
2. what the tree looks like after reheap is called as a result of each addition

java111c_question 6.PNG (https://ccsf.instructure.com/files/7780704/download)

Question 7

8 / 8 pts

Trace the creation of the **maxheap** from the numbers below.

42, 68, 73, 82, 91

Use the **reheap approach** described in the textbook (Section "Using Reheap"), which is the second approach described in the lecture.

In this approach:

- add all elements to the tree at once and then repeatedly call reheap
- reheap is **not** called for every node.

In your answer show:

1. the initial tree after all elements are added
2. the array that represents this initial tree
3. for each call to reheap, show a) which index is being reheaped and b) what the tree looks like after the swapping that occurs

java111c_question 7.PNG (https://ccsf.instructure.com/files/7777685/download)

Question 8

10 / 10 pts

Draw the **2-3 tree** that results from adding the following numbers in this order to an initially empty tree.

37, 18, 19, 21, 42, 25, 45, 36, 29, 43

Show what the tree looks like after **each** element is added.

You are not required to show the splitting steps, but you must show what the tree looks like after each insertion. (It is okay if you do show the splitting steps.)

This means there should be at least **10 trees** in your answer to this question.

java111c_question 8.PNG (https://ccsf.instructure.com/files

/7780701/download)

Question 9

10 / 10 pts

Draw the **2-4 tree** that results from adding the following numbers in this order to an initially empty tree.

25, 36, 21, 45, 39, 40, 32, 42, 37, 43, 18

Show what the tree looks like after **each** element is added.

You are not required to show the splitting steps, but you must show what the tree looks like after **each** insertion. (It is okay if you do show the splitting steps.)

This means there should be at least **11 trees** in your answer to this question.

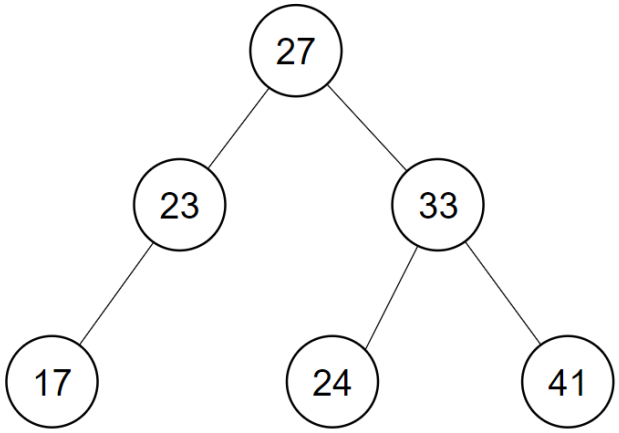
java111c_question 9.PNG (https://ccsf.instructure.com/files/7780705/download)

Multiple Choice Questions- Trees Part One

Question 10

3 / 3 pts

Select whether the graph below is each of the following.



Correct! **tree** yes

Correct! **binary tree** yes

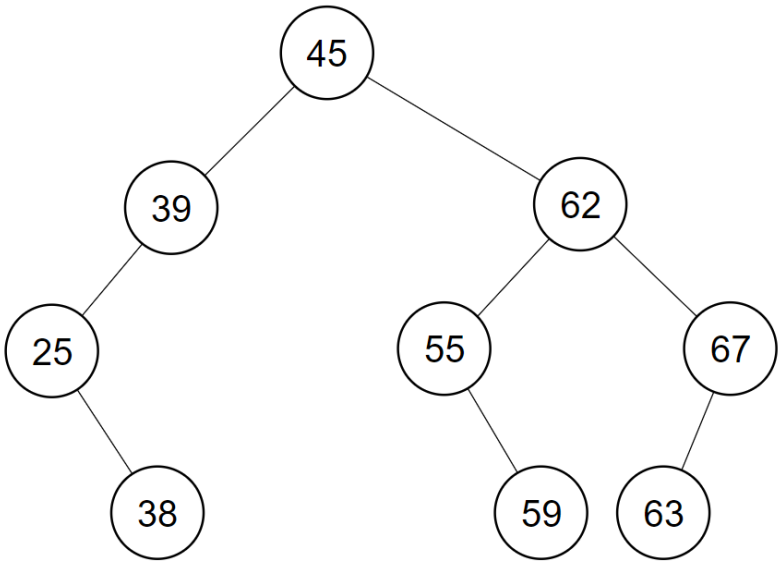
Correct! **binary search tree** no

The tree is not a binary search tree because 24 is in 27's left subtree.

Question 11

3 / 3 pts

Select whether the graph below is each of the following.



Correct! **tree** yes

Correct! **binary tree** yes

Correct! **binary search tree** yes

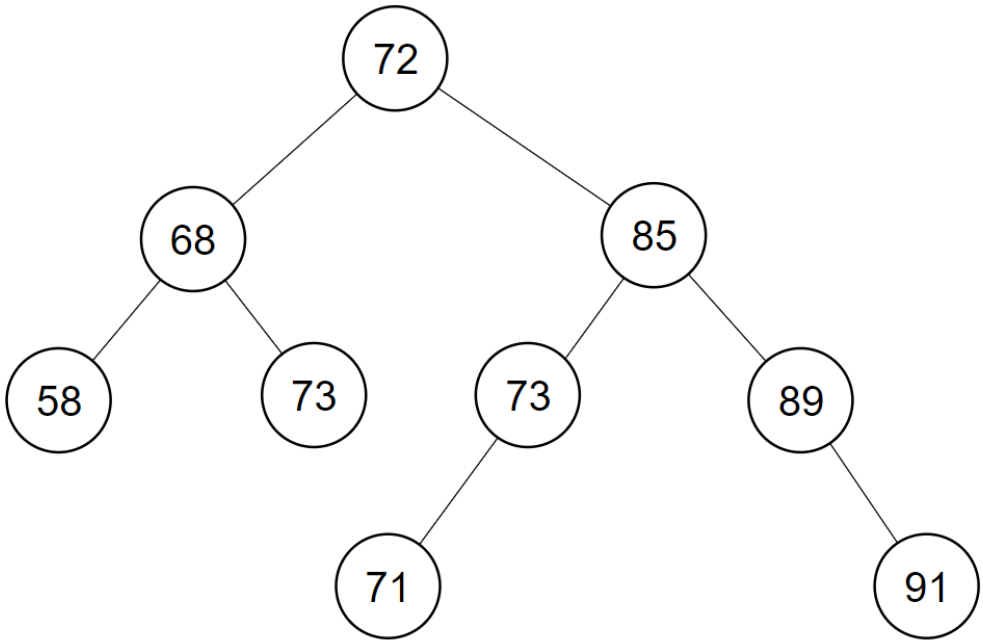
Other Incorrect Match Options:

- no

Question 12

3 / 3 pts

Select whether the graph below is each of the following.



Correct! **tree** yes

Correct! **binary tree** yes

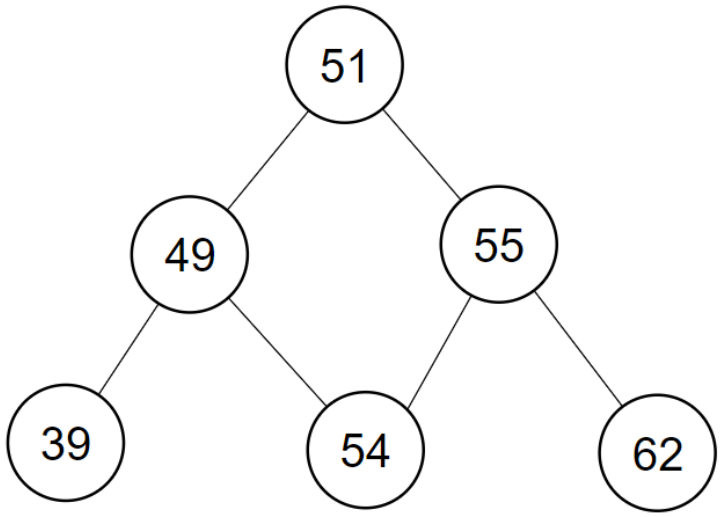
Correct! **binary search tree** no

The tree is not a binary search tree because 73 is in 72's left subtree and 71 is in 72's right subtree.

Question 13

3 / 3 pts

Select whether the graph below is each of the following.



Correct!	**tree**	no
Correct!	**binary tree**	no
Correct!	**binary search tree**	no

Other Incorrect Match Options:

- yes

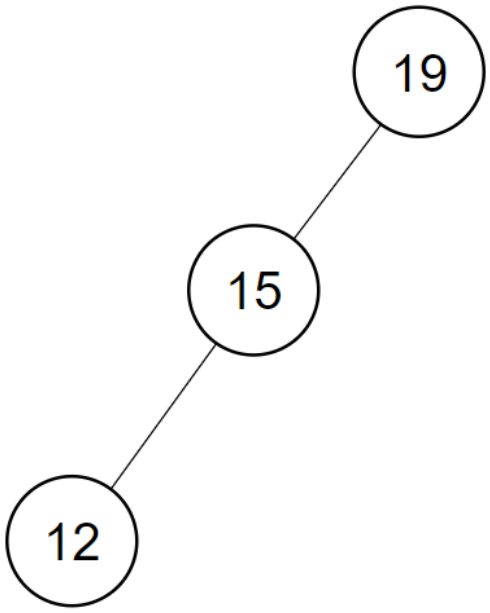
The graph is not a tree because it has a cycle.

It is not a binary tree or binary search tree because it is not a tree.

Question 14

3 / 3 pts

Select whether the graph below is each of the following.



Correct!	**tree**	yes
Correct!	**binary tree**	yes
Correct!	**binary search tree**	yes

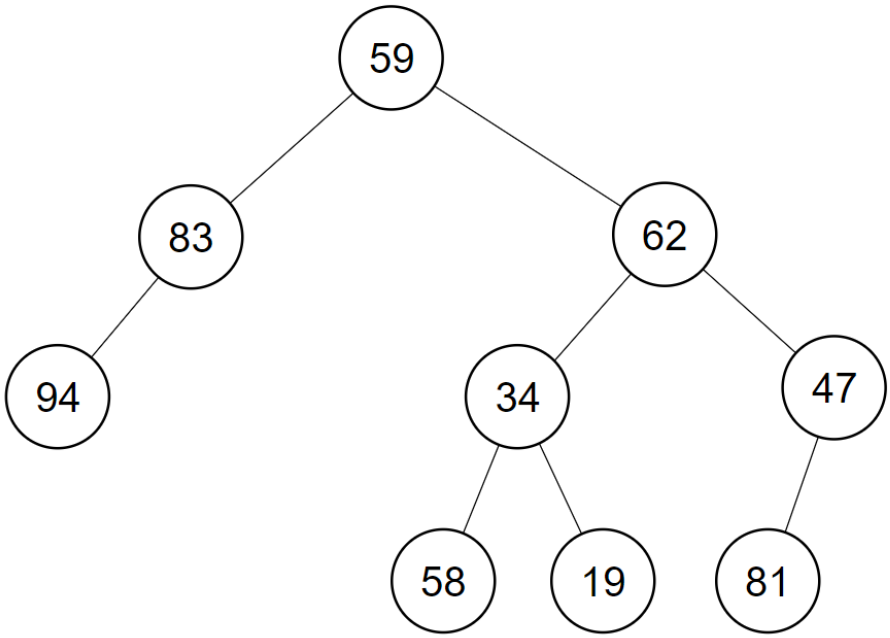
Other Incorrect Match Options:

- no

Question 15

3 / 3 pts

In the binary tree below, what is the type of each node?



Correct!	**62**	interior node
Correct!	**19**	leaf
Correct!	**83**	interior node
Correct!	**94**	leaf
Correct!	**59**	root node
Correct!	**34**	interior node

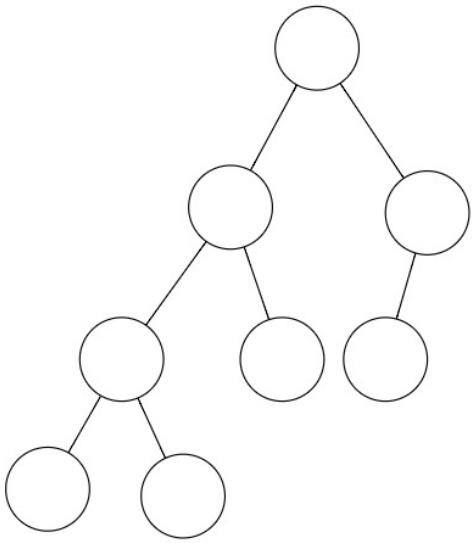
Other Incorrect Match Options:

- none of these is correct

Question 16

2 / 2 pts

Select whether the binary tree below is each of the following.



Correct!

full no

Correct!

complete no

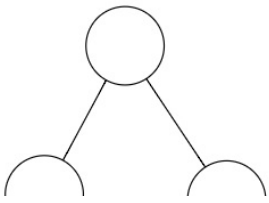
Other Incorrect Match Options:

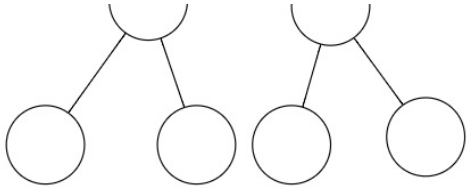
- yes

Question 17

2 / 2 pts

Select whether the binary tree below is each of the following.





Correct! **full** yes

Correct! **complete** yes

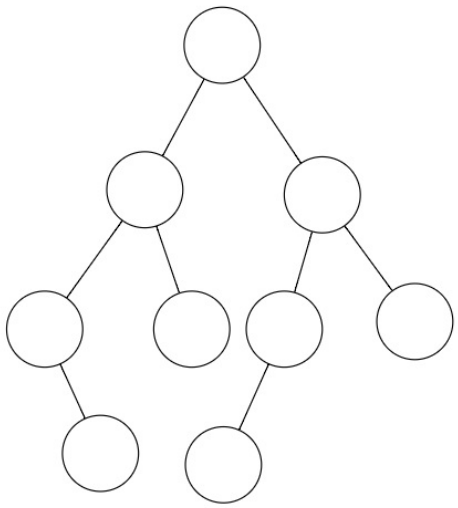
Other Incorrect Match Options:

- no

Question 18

2 / 2 pts

Select whether the binary tree below is each of the following.



Correct! **full** no

Correct!

complete no

Other Incorrect Match Options:

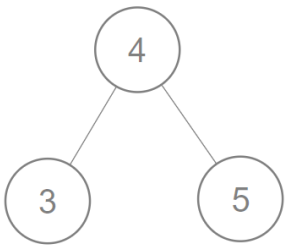
- yes

Question 19

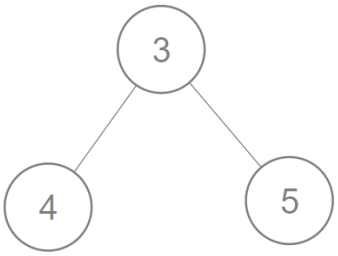
1 / 1 pts

Which binary tree would be represented by the heapform array below?

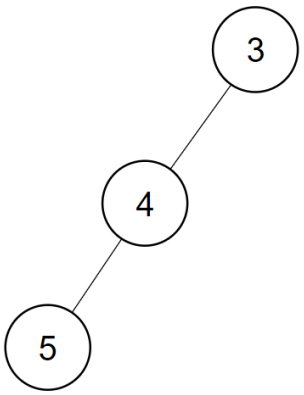
index	0	1	2	3	4	5	6	7
value		3	4		5			

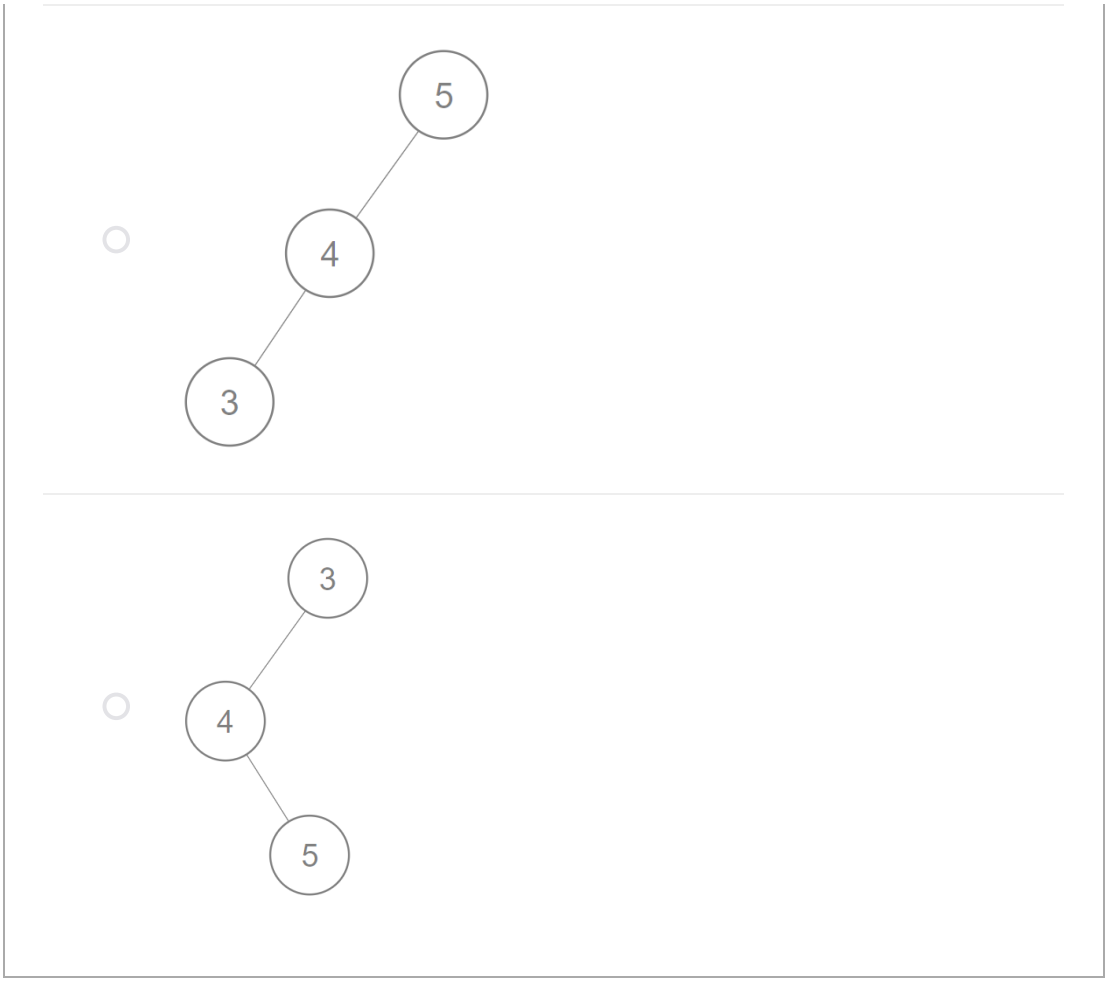


none of these is correct

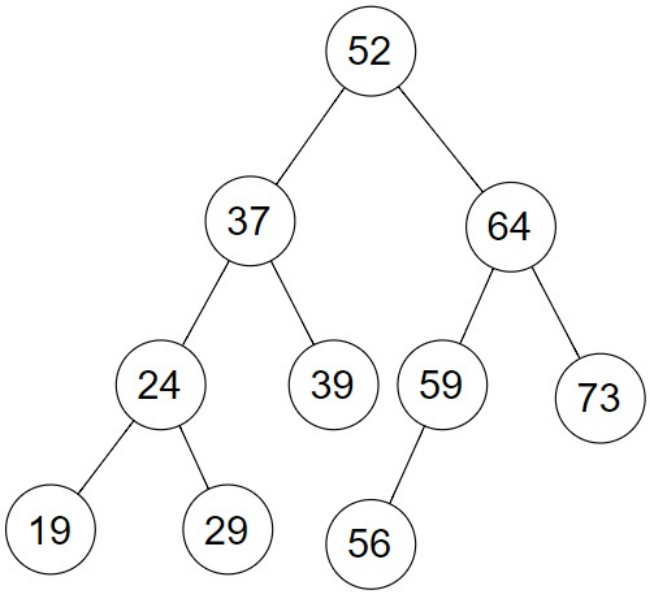


Correct!




5
4
3
3
4
5

Use the **binary search tree** below for the next two questions.



Question 20

1 / 1 pts

What nodes are visited and in what order when searching for the element 29?

- 52 37 24 19 29
- **Correct!** 52 37 24 29
- none of these is correct
- 52 29
- 52 37 24 19 24 29

Question 21

1 / 1 pts

What nodes are visited and in what order when searching for the element 58?

- none of these is correct
- 52 64 59 56 **Correct!**
- 52 64 59 56 58
- 52 64
- 52 37 64 59 56
- 52 64 59

Question 22

1 / 1 pts

Consider the two insertion sequences of a dataset into a **binary search tree**.

24 17 28 38 32 22

17 22 24 28 32 38

True or False: The two trees will be equally efficient to search for values.

- True
- False **Correct!**

The second (sorted) insertion order results in a tree that is essentially a linked list. The first tree is much more balanced, which will allow a more efficient search.

Question 23

5 / 5 pts

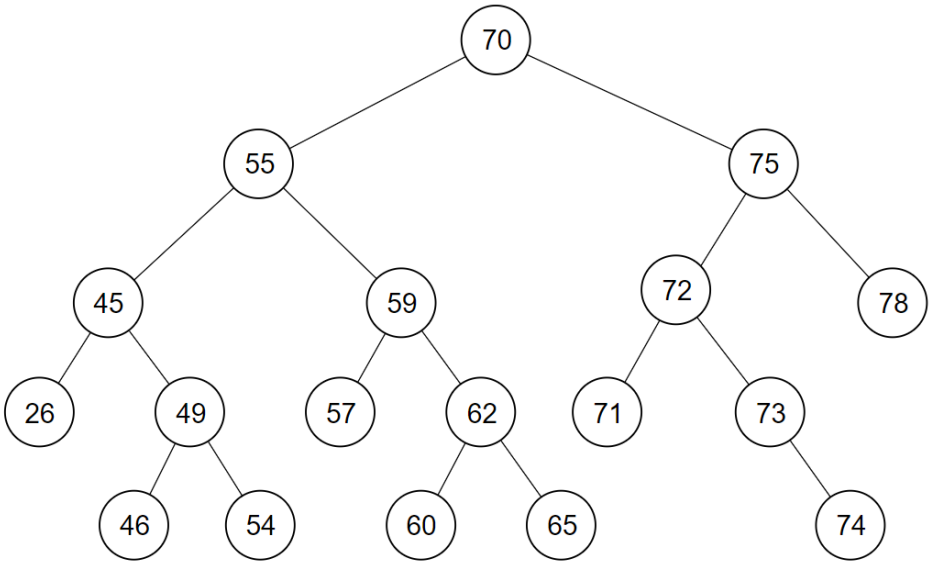
Use the **binary search tree** below and evaluate how you could remove various nodes.

For each removal, state whether it is a valid removal **using the two algorithms from the textbook (and lecture notes and videos).**

A removal can either be:

- valid: the removal uses one of the two algorithms and creates a binary search tree
- invalid: either a) the removal does not use one of the two algorithms or b) the removal results in a tree that is no longer a binary search tree

Assume the tree is as shown in the picture below for each removal. (In other words, no nodes have been removed yet.)

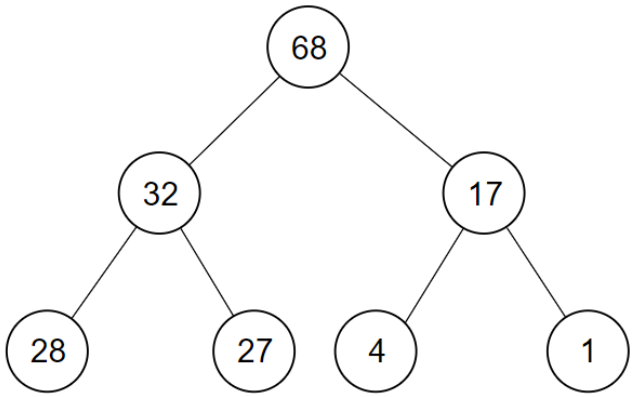


Correct!	remove 55 and replace with 57	valid
Correct!	remove 55 and replace with 54	valid
Correct!	remove 55 and replace with 46	invalid
Correct!	remove 72 and replace with 74	invalid
Correct!	remove 70 and replace with 65	valid
Correct!	remove 70 and replace with 78	invalid
Correct!	remove 70 and replace with 71	valid
Correct!	remove 49 and replace with 46	valid
Correct!	remove 49 and replace with 54	valid
Correct!	remove 59 and replace with 62	invalid

Multiple Choice Questions- Trees Part Two

Question 24

2 / 2 pts



Select whether the tree above is each of the following.

Correct!

complete yes

Correct!

full yes

Correct!

a maxheap yes

Other Incorrect Match Options:

- no

The tree is complete.

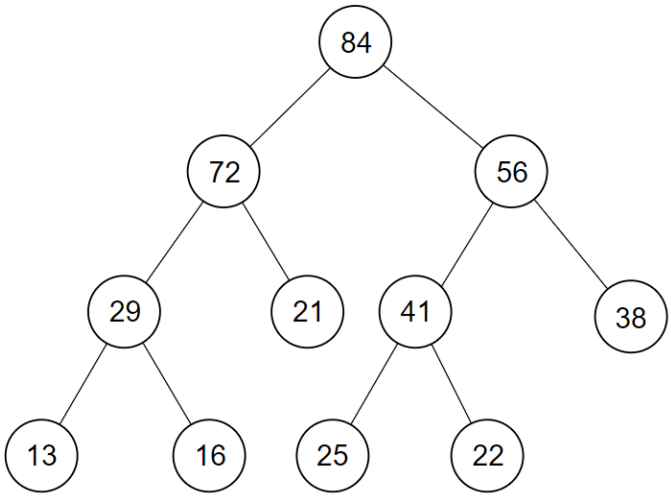
For every node, each node is greater than ALL nodes in the left and right subtrees.

Because of the two facts above, the tree is a maxheap.

The tree is full because the lowest level has all possible leaves.

Question 25

2 / 2 pts



Select whether the tree above is each of the following.

Correct! **complete** no

Correct! **a maxheap** no

Other Incorrect Match Options:

- yes

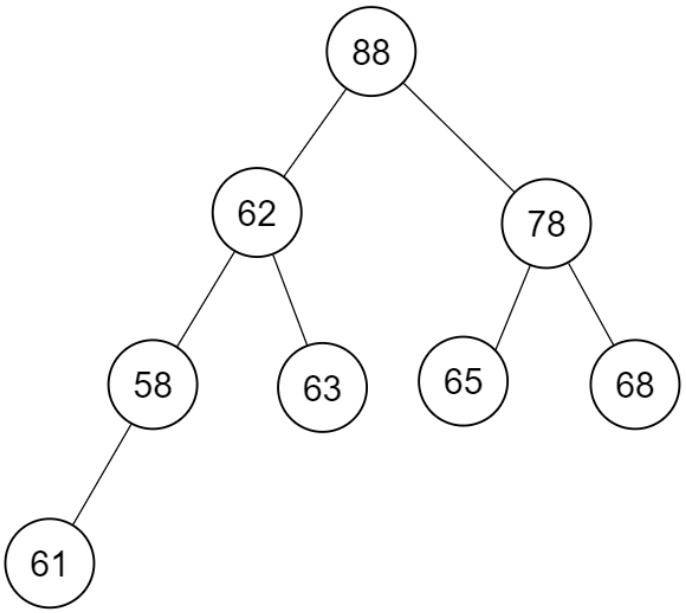
The tree is not complete because the leaves are not as far to the left as possible.

It's true that, for every node, each node is greater than ALL nodes in the left and right subtrees.

Even so, because the tree is not complete, it's not a maxheap.

Question 26

2 / 2 pts



Select whether the tree above is each of the following.

Correct! **complete** yes

Correct! **a maxheap** no

Correct! **full**

no

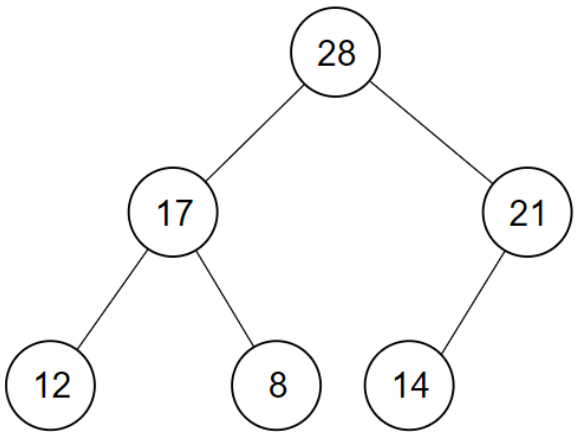
The tree is complete.

61 is below 58, and 63 is below 62, so it is not a maxheap.

It is not full because not all non-leaves have exactly two children (58 has one child).

Question 27

2 / 2 pts



Select whether the tree above is each of the following.

Correct! **complete** yes

Correct! **a maxheap** yes

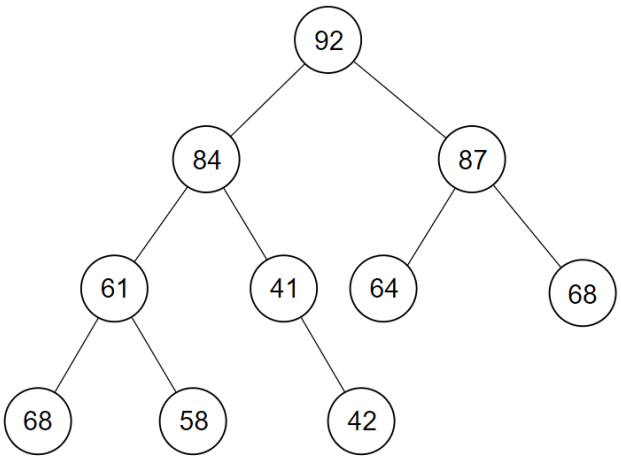
Correct! **full** no

The tree is complete. For every node, each node is greater than ALL nodes in the left and right subtrees. The tree is a maxheap.

The tree is not full because the lowest level does not have all possible leaves.

Question 28

2 / 2 pts



Select whether the tree above is each of the following.

Correct! **complete** no

Correct! **a maxheap** no

Other Incorrect Match Options:

- yes

The tree is not complete because the leaves are not as far to the left as possible.

The tree is not a maxheap because of that. Also, 68 is below 61 and 42is below 41, so it is also not a maxheap for this reason.

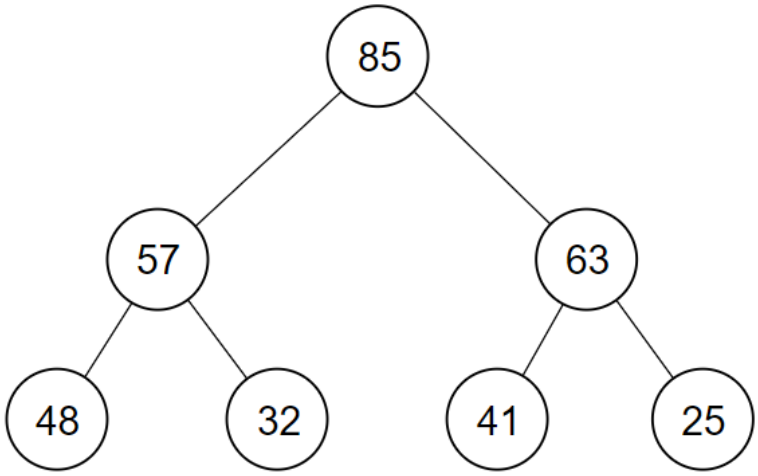
Question 29

1 / 1 pts

The following numbers are inserted into a tree in *some order*. (The numbers shown are the contents of the tree, **not** the insertion order.)

25 32 41 48 57 63 85

Which of the following is true about the picture below?



Correct!

- (●) The tree is a valid maxheap for these numbers but there are other valid maxheaps that could be created from the same numbers.
- ( ) The tree is not a valid maxheap for these numbers.
- ( )

The tree is a valid maxheap for these numbers and there are no other valid maxheaps that could be made for these numbers.

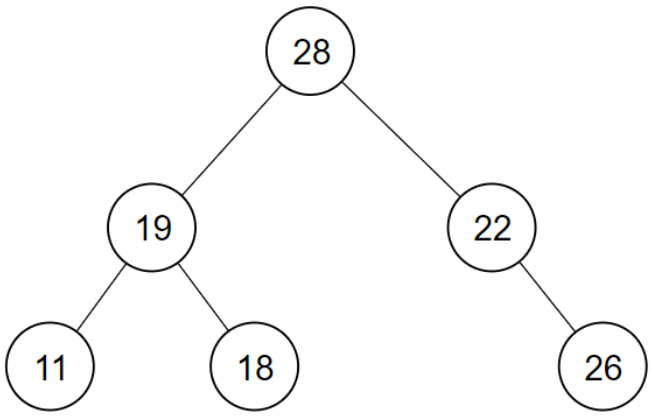
Question 30

1 / 1 pts

The following numbers are inserted into a tree in *some order*. (The numbers shown are the contents of the tree, **not** the insertion order.)

11 18 19 22 26 28

Which of the following is true about the picture below?



- The tree is a valid maxheap for these numbers and there are no other valid maxheaps that could be made for these numbers.
- The tree is a valid maxheap for these numbers but there are other valid maxheaps that could be created from the same numbers.

Correct!

- (●) The tree is not a valid maxheap for these numbers.

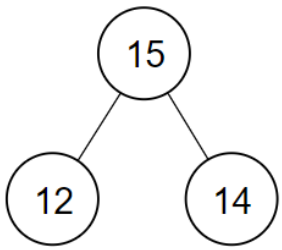
Question 31

1 / 1 pts

The following numbers are inserted into a tree in *some order*. (The numbers shown are the contents of the tree, **not** the insertion order.)

12 14 15

Which of the following is true about the picture below?



Correct!

- (selected) The tree is a valid maxheap for these numbers but there are other valid maxheaps that could be created from the same numbers.
- The tree is a valid maxheap for these numbers and there are no other valid maxheaps that could be made for these numbers.
- The tree is not a valid maxheap for these numbers.

Question 32

8 / 0 pts

Optional Extra Credit (10 points)

The preorder traversal of a **binary search tree** is below.

20 10 5 3 8 30 25 21 22

What is the postorder traversal?

To answer, list each value visited separated by a space (e.g., 52 64 73...). Include no other characters.

ou Answered

3 5 8 10 21 25 22 30 20

orrect Answers

3,8,5,10,22,21,25,30,20

3 8 5 10 22 21 25 30 20

3, 8, 5, 10, 22, 21, 25, 30, 20

Quiz Score: **108** out of 100