

Due May 30 at 11:59pm**Points** 100**Questions** 40**Available** May 24 at 12am - May 30 at 11:59pm 7 days**Time Limit** 240 Minutes

Instructions

Final Exam

Instructions

- You have four hours to complete the exam.
- **You must complete the test in one sitting.**
- This test is open-note/open-book/open-web, but **must be completed individually.**
- Communicating (in person or online) with anyone other than me is considered cheating and will result in a failing grade and disciplinary procedures.
- For the questions that ask you to draw a tree or graph, 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 .heic or .paint files- I cannot view these files.**
- For coding questions, I recommend completing the exam questions in your own IDE and then pasting the answers into Canvas when you are ready to submit. You can review the [Posting Code to Canvas](#) page if needed.

* Canvas Resources

🌀 [How do I take a quiz? \(https://community.canvaslms.com/docs/DOC-10645-421241977\)](https://community.canvaslms.com/docs/DOC-10645-421241977)

🌀 [How do I submit a quiz? \(https://community.canvaslms.com/docs/DOC-10583-421250759\)](https://community.canvaslms.com/docs/DOC-10583-421250759)

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

Attempt History

	Attempt	Time	Score
LATEST	Attempt 1	240 minutes	97 out of 100

Score for this quiz: **97** out of 100

Submitted May 27 at 1:22am

This attempt took 240 minutes.

Unanswered

Question 1

0 / 0 pts

I understand that I must complete this test **on my own**.

The following activities are considered plagiarism/cheating and will result in a grade of 0 for the exam and possible disciplinary action:

- Discussing the test with anyone else, including students and tutors (in person or online)
- Communicating with others in person or online
- Posting code from the exam to study/question/discussion sites
- Submitting code that you did not write
- Copying code from a website or from another person

Note: You are allowed to discuss the test with the instructor, although there is no guarantee that she will be online while you are taking the exam.

If you have questions about this, stop immediately and contact Prof. Masters.

Correct Answer

- ☐ I understand and agree to all of the above.

Question 2

0 / 0 pts

I understand that I must take the final exam in a single sitting- **all at once**.

If you have questions about this, stop immediately and contact Prof. Masters.

Correct!☒ I understand

Section 1: Coding Questions

Provided Files

I've provided files for the coding questions. **I strongly recommend that you use the provided tester program to test your code.**

[FinalExamFiles.zip](https://ccsf.instructure.com/courses/47904/files/7254663/download?download_frd=1)  (https://ccsf.instructure.com/courses/47904/files/7254663/download?download_frd=1)

For all coding questions, you are writing code at the **implementation level**. This means you have direct access to the instance data variables (the array or nodes).

For full credit:

- Write an efficient solution that takes advantage of being able to write code at the implementation level.
- Only write the required method (or a helper method). Do not change anything else in the class.
- Account for all possible times when the method could be invoked, including with an empty or singleton datasets.
 - Your code should not crash or throw a runtime exception in these situations.

- Do **not** invoke the `toArray()` method.
- Use only the data structure asked about in the question. Do **not** use a different data structure.
- Follow general best practice of coding, including:
 - Reduce duplicated code
 - Follow naming conventions
- Use correct syntax (the code compiles) and correct semantics (the code accomplishes the task).

Write a complete **recursive** method from the **implementation perspective** that will go inside the **LinkedList** class. The method will determine if a queue contains duplicate entries (two entries that are equal- logically equivalent) and next to each other ("consecutive").

The method header is:

```
public boolean hasConsecutiveDuplicates()
```

Notes:

- Having a recursive solution is worth 5 out of the 15 points.
- You **can** create a helper method. If you do, be sure to submit both your `hasConsecutiveDuplicates()` method **and** your helper method.

Question 3

12 / 15 pts

Paste your complete `hasConsecutiveDuplicates` method for `LinkedList` here. If you used a helper method, be sure to submit both methods here.

Your Answer:

```
public boolean hasConsecutiveDuplicates() {  
    if (this.isEmpty()) {
```

```

        return false;
    }
    // traverse linked list
    boolean hasDups = false;
    Node current = null;
    current = firstNode;
    hasDups = hasConsecutiveDuplicatesHelper(current, hasDups);
    return hasDups;
}

private boolean hasConsecutiveDuplicatesHelper(Node current, boolean hasDups) {
    // base case
    if (hasDups == true) {
        return true;
    }

    if (current != null && current.next != null) {
        if (current.data.equals(current.next.data)) {
            hasDups = true;
        }
    }

    if (hasDups != true && current != null) {
        current = current.next;
        hasConsecutiveDuplicatesHelper(current, hasDups);
    }

    return hasDups;
}

```

-3 the recursive method call must be returned or else the recursive calls will not be linked together and the result will not be correct

Write a complete method from the **implementation perspective** that will go inside the **ArrayStack** class. The method determines the same thing as above: whether a stack contains two duplicate entries (entries that are equal- logically equivalent) and next to each other ("consecutive").

Note: This method does **not** have to be recursive.

The method header is:

```
public boolean hasConsecutiveDuplicates()
```

Question 4**10 / 10 pts**

Paste your complete hasConsecutiveDuplicates method for ArrayStack here.

Your Answer:

```
public boolean hasConsecutiveDuplicates() {
    if (this.stack[0] == null) {
        return false;
    }
    int i = 0;

    while (stack[i] != null && i < (stack.length - 1)) {
        if (stack[i].equals(stack[i + 1])) {
            return true;
        }
        // increment index
        i++;
    }
    return false;
}
```

Section 2: Drawing Trees and Graphs

Question 5**4 / 4 pts**

Draw the **binary search tree** that is formed by inserting the following numbers **in the order listed**.

15 4 11 18 26 33 7 21

↓ [Java_Final_Question 5.PNG \(https://ccsf.instructure.com/files/7931508/download\)](https://ccsf.instructure.com/files/7931508/download)

Question 6

4 / 4 pts

Draw the **2-3 B-Tree** that results after adding **each** of the following numbers.

Your answer should contain at least **six trees** to show the tree after **each** addition.

Answers that show only the final tree will **not** get full credit.

13 17 11 21 26 20

↓ [Java_Final_Question 6.PNG \(https://ccsf.instructure.com/files/7931542/download\)](https://ccsf.instructure.com/files/7931542/download)

Question 7

4 / 4 pts

Draw the directed graph from the following adjacency matrix.

	A	B	C	D	E
A		3		4	
B	3				2
C					
D	4				
E				5	

↓ [Java_Final_Question 7.PNG \(https://ccsf.instructure.com/files/7931542/download\)](https://ccsf.instructure.com/files/7931542/download)

</files/7931518/download>

Section 3: Short Answer Questions

Question 8

2 / 2 pts

Evaluate the following postfix expression:

3 4 2 - 5 + *

Correct!**Correct Answers**

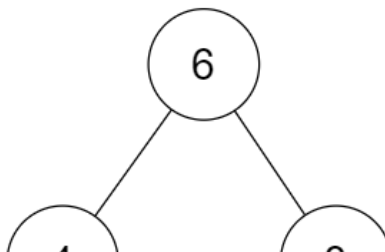
21 (with margin: 0)

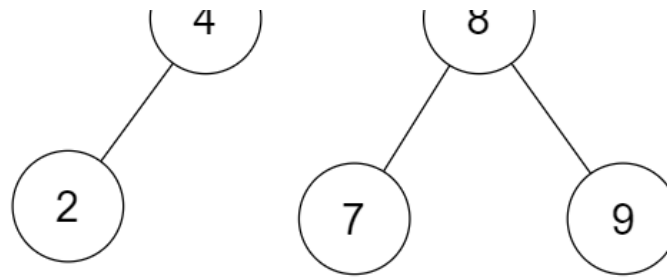
Question 9

9 / 9 pts

What is the preorder, inorder, and postorder traversal of the binary tree below?

To list a traversal, list each value visited separated by a space (example: 8 2 9 4 ...). Include no other characters.





What is the **preorder** traversal?

What is the **inorder** traversal?

What is the **postorder** traversal?

Answer 1:

Correct! 6 4 2 8 7 9

Incorrect Answer 6, 4, 2, 8, 7, 9

Incorrect Answer 642879

Incorrect Answer 6,4,2,8,7,9

Answer 2:

Correct! 2 4 6 7 8 9

Incorrect Answer 2, 4, 6, 7, 8, 9

Incorrect Answer 246789

Incorrect Answer 2,4,6,7,8,9

Answer 3:

Correct! 2 4 7 9 8 6

Incorrect Answer 2, 4, 7, 9, 8, 6

Incorrect Answer 247986

Correct Answer

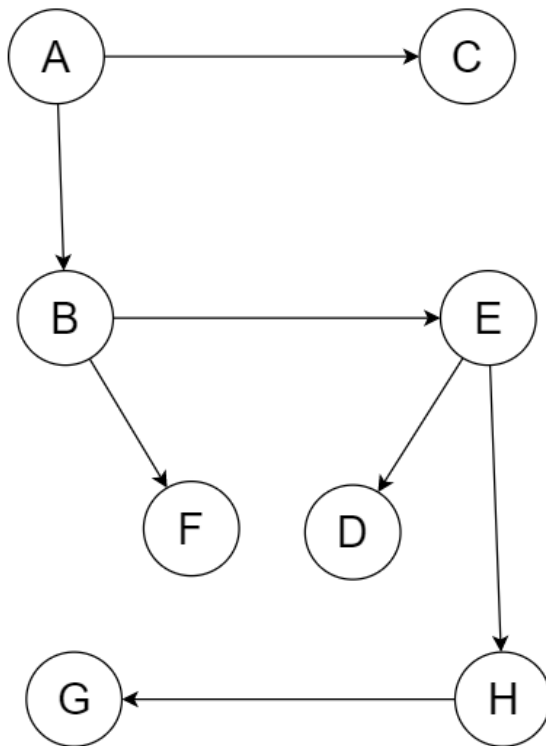
2,4,7,9,8,6

Question 10

6 / 6 pts

What is a depth-first and breadth-first traversal of the following graph, when starting at Vertex A?

To list a traversal, list each value visited separated by a space (e.g., A B C...). Include **no** other characters.



What is the depth-first traversal?

A B E H G D F C

What is the breadth-first traversal?

A B C E F D H G

Answer 1:

Correct!

A B E H G D F C

Incorrect Answer ABEDHGF C

Incorrect Answer ABFEHGD C

Incorrect Answer ABFEDHGC

Incorrect Answer ACBEDHGF

Incorrect Answer ACBEHGD F

Incorrect Answer ACBF EHG D

Incorrect Answer ACBFEDHG

Answer 2:

Correct! ABCEFDHG

Incorrect Answer ABCEFH D G

Incorrect Answer ABCFEDHG

Incorrect Answer ABCFEH D G

Incorrect Answer ACBEFDHG

Incorrect Answer ACBEFH D G

Incorrect Answer ACBFEDHG

Incorrect Answer ACBF EHG D

Section 4: Multiple Choice Questions- Recursion

Question 11**1 / 1 pts**

It is always possible to convert a recursive algorithm to an iterative algorithm.

Correct!☒ True☐ False**Question 12****1 / 1 pts**

The following method will run under any condition without infinite recursion and without throwing any errors or exceptions.

```
public void mystery(Node node) {  
    if (node == null) {  
        System.out.println("Data: " + node.data);  
    } else {  
        System.out.println("Next Data: " + node.next.data);  
        mystery(node.next);  
    }  
}
```

☐ True**Correct!**☒ False**Question 13****1 / 1 pts**

The following method will run under any condition without infinite

recursion and without throwing any errors or exceptions.

```
public int mystery(int n) {  
    if (n == 0) {  
        return n;  
    } else {  
        return 1 + mystery(n-1);  
    }  
}
```

☐ True

Correct!

☒ False

Question 14

1 / 1 pts

What gets printed by the following pseudocode if the method is invoked with the following array of chars and index 0?

[a, b, c]

```
public void mystery(char[] array, int index) {  
    if(index < array.length) {  
        print array[index]  
        mystery(array, index+1)  
        print array[index]  
    }  
}
```

☐ a b a

☐ a b

☐ a b a b

Correct!

☒ a b c c b a

☐ a b c a b c☐ a b c☐ a b b a**Question 15****1 / 1 pts**

What is returned from invoking the method with n=4?

```
public int mystery(int n) {  
    int result = 0;  
    if(n<=0) {  
        result = n;  
    } else if(n%2== 0) {  
        result = 1 + mystery(n-1);  
    } else {  
        result = mystery(n-1);  
    }  
    return result;  
}
```

Correct!**Correct Answers**

2 (with margin: 0)

Question 16**1 / 1 pts**

What is returned from invoking the method with n=3?

```
public int mystery(int n) {  
    if(n <= 1) {  
        return n;  
    } else {
```

```
        return n * mystery(n-1);
    }
}
```

Correct!**Correct Answers**

6 (with margin: 0)

Question 17**1 / 1 pts**

Does the following method correctly calculate the sum of numbers between $n1$ and $n2$ (inclusive)? (For example, the sum of numbers between 10 and 12 should be $10+11+12 = 33$).

```
public int sumUp(int n1, int n2) {
    int sum = n1;
    if(n1 < n2) {
        sumUp(n1+1, n2);
    }
    return sum;
}
```

Correct!☒ No☐ Yes

Section 5: Multiple Choice Questions- Sorting and Searching

Question 18**2 / 2 pts**

Here is an unsorted array.

6	7	3	4	2
---	---	---	---	---

What are the contents of the array after the **first pass** of **selection sort**?

- ☐ [6, 7, 3, 4, 2]
- ☐ none of these is correct
- ☐ [2, 3, 4, 6, 7]
- ☐ [3, 6, 7, 4, 2]
- ☐ [2, 6, 7, 3, 4]
- ☒ [2, 7, 3, 4, 6]

Correct!**Question 19****1 / 1 pts**

After 3 passes of the outer loop of **selection sort**, an array looks like this:

2	4	6	8	11	12	15	17	19	21	22	24	25
---	---	---	---	----	----	----	----	----	----	----	----	----

Will the selection sort continue to loop or will the sort end?

- ☐ The algorithm will end because the array is already sorted.
- ☒ The sorting algorithm will continue to loop.

Correct!

Question 20**2 / 2 pts**

Here is an unsorted array.

5	2	6	1	7
---	---	---	---	---

What are the contents of the array after the **first pass of insertion sort**?

- ☐ [1, 2, 6, 5, 7]
- ☐ [1, 2, 5, 6, 7]
- ☐ [1, 5, 2, 6, 7]
- ☐ none of these is correct
- ☒ [2, 5, 6, 1, 7]

Correct!**Question 21****1 / 1 pts**

After 3 passes of the outer loop of **insertion sort**, an array looks like this:

2	4	6	8	11	12	15	17	19	21	22	24	25
---	---	---	---	----	----	----	----	----	----	----	----	----

Will the insertion sort continue to loop or will the sort end?

- ☐ The algorithm will end because the array is already sorted.
- ☒ The sorting algorithm will continue to loop.

Correct!

Question 22**1 / 1 pts**

One pass of the Quicksort's partition method is performed with the partition in index 4 (the value 38).

Could the array below be the result of one pass of partition?

index:	0	1	2	3	4	5	6	7	8
value:	37	31	12	18	38	46	41	59	39

Correct!☒ True☐ False**Question 23****2 / 2 pts**

What is the sequence of indexes when performing an **optimized sequential (linear)** search for 22 on the sorted list of values below ?

index:	0	1	2	3	4	5
value:	12	18	19	24	27	30

☐ 0, 1, 2, 3, 4, 5, 6**Correct!**☒ 0, 1, 2, 3☐ 0, 1, 2

☐ 0, 1, 2, 3, 4, 5☐ none of these is correct

Use one of these methods for the next questions.

```
public static int binarySearchIterative(int[] numbers,
int target) {
    boolean found = false;
    int first = 0;
    int last = numbers.length - 1;

    while (first <= last && !found) {
        int mid = (first + last) / 2;

        if (numbers[mid] == target) {
            targetLocation = mid;
            found = true;
        } else if (numbers[mid] < target) {
            first = mid + 1;
        } else { // numbers[mid] > target
            last = mid - 1;
        }
    }
    return targetLocation;
}

public static int binarySearchRecursive(int[] numbers,
int target) {
    return binarySearchRecursiveHelper(numbers, target,
0, numbers.length - 1);
}

private static int binarySearchRecursiveHelper(int[] nu
mbers, int target, int first, int last) {
    int mid = ((last - first) / 2) + first;

    if (first > last) {
        return -1; // indices cross over
    } else if (target == numbers[mid]) {
        return mid; // we found it!
    } else if (target < numbers[mid]) {
        return binarySearchRecursiveHelper(numbers, targe
t, first, mid - 1);
    }
}
```

```
    } else { // target > numbers[mid]
        return binarySearchRecursiveHelper(numbers, target, mid + 1, last);
    }
}
```

Question 24**2 / 2 pts**

Using the code above, what is the sequence of indexes visited when performing a binary **search** on the sorted array for 32?

In other words, what are the values of mid?

index:	0	1	2	3	4	5	6	7	8
value:	12	19	32	36	41	43	52	66	79

- ☐ 4, 0, 2
- ☐ none of these is correct
- ☐ 4, 3, 2
- ☐ 4, 2
- ☒ 4, 1, 2

Correct!**Question 25****2 / 2 pts**

What is the sequence of indexes visited when performing

a **binary search** on the sorted array for 68?

In other words, what are the values of mid?

index	0	1	2	3	4	5	6	7	8
value:	7	12	23	26	35	48	52	61	68

Correct!

- ☒ 4, 6, 7, 8
- ☐ 4, 8
- ☐ none of these is correct
- ☐ 4, 6, 8
- ☐ 4, 7, 8

Question 26

1 / 1 pts

Binary searches can be used on unsorted data.

☐ True

Correct!

☒ False

Section 6: Multiple Choice Questions-

Stacks and Queues

Question 27

2 / 2 pts

The stack variable is initially empty. What is the contents of the stack after the following code is executed? Answer choices are listed

BOTTOM ... TOP

```
stack.push(4);  
stack.push(5);  
stack.pop();  
stack.push(1);
```

☐ bottom 4 5 1 top

☒ bottom 4 1 top

☐ bottom 5 1 top

☐ bottom 1 5 top

☐ bottom 5 4 top

☐ bottom 1 5 4 top

☐ none of these is correct

☐ bottom 4 5 top

☐ bottom 1 4 top

Correct!

Question 28**2 / 2 pts**

The stack variable is initially empty. What is the contents of the stack after the following code is executed? Answer choices are listed

BOTTOM ... TOP

```
stack.push(6);  
stack.push(2);  
stack.push(stack.pop());
```

☐ bottom 6 2 2 top☐ bottom 2 6 6 top☒ bottom 6 2 top☐ none of these is correct☐ bottom 2 6 2 top☐ bottom 6 top☐ bottom 2 2 top☐ bottom 6 2 6 top☐ bottom 6 6 top☐ bottom 2 6 top☐ bottom 2 top**Correct!****Question 29****2 / 2 pts**

A queue is initially empty. What are the contents of the queue after the following code is executed? Answer choices are listed **FRONT ... BACK**.

```
queue.enqueue(1);  
queue.enqueue(6);  
queue.enqueue(queue.dequeue());
```

Correct!

☐ front 1 6 back

☒ front 6 1 back

☐ front 1 back

☐ front 1 1 back

☐ front 6 back

☐ front 6 6 back

☐ none of these is correct

Question 30

2 / 2 pts

A queue is initially empty. What are the contents of the queue after the following code is executed? Answer choices are listed **FRONT ... BACK**.

```
queue.enqueue(4);  
queue.enqueue(2);  
queue.enqueue(queue.getFront());
```

Correct!

☒ front 4 2 4 back

- ☐ front 2 4 4 back
- ☐ front 2 4 back
- ☐ none of these is correct
- ☐ front 4 2 back
- ☐ front 4 4 back
- ☐ front 2 2 back
- ☐ front 2 4 2 back
- ☐ front 4 2 2 back

Question 31**2 / 2 pts**

A deque is initially empty. What are the contents of the deque after the following code is executed? Answer choices are listed **FRONT ... BACK**.

```
deque.addToFront(2);  
deque.addToBack(3);  
deque.addToFront(1);  
deque.addToBack(4);  
deque.removeFront();
```

- ☐ front 2 3 1 4 back
- ☒ front 2 3 4 back
- ☐ front 1 2 3 back
- ☐ none of these is correct

Correct!

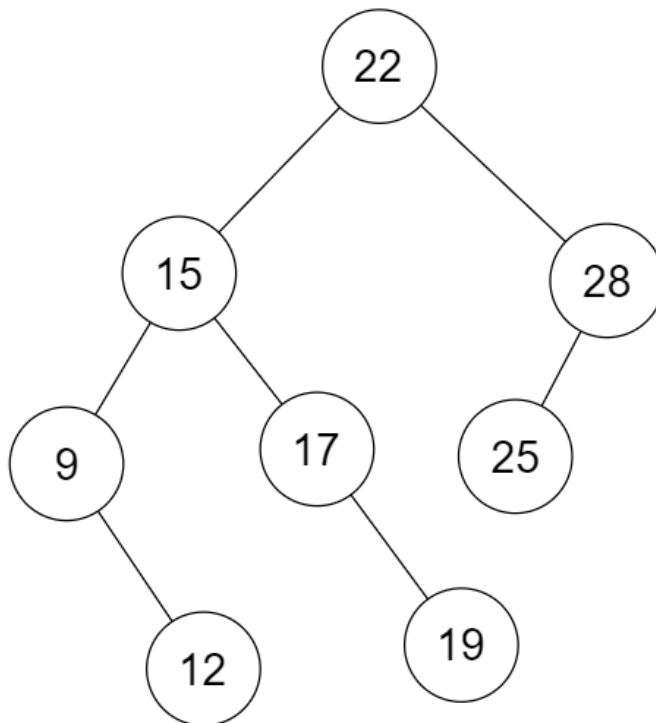
☐ front 1 2 3 4 back☐ front 2 3 1 back

Section 7: Multiple Choice Questions-Trees

Question 32

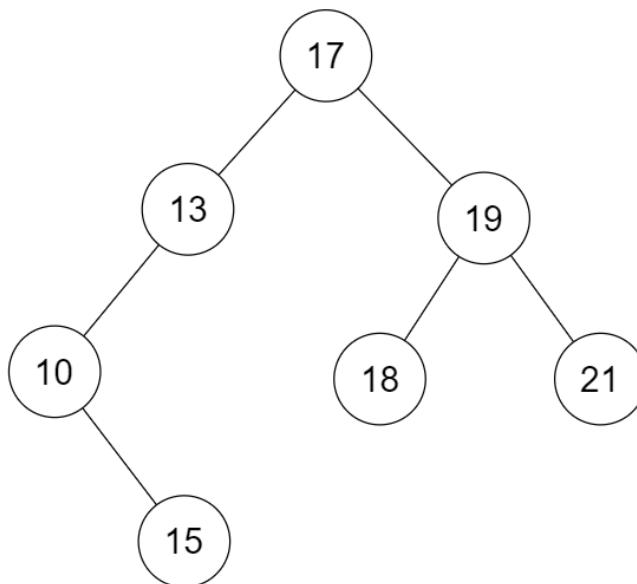
1 / 1 pts

Is the picture below a valid **binary search tree**?

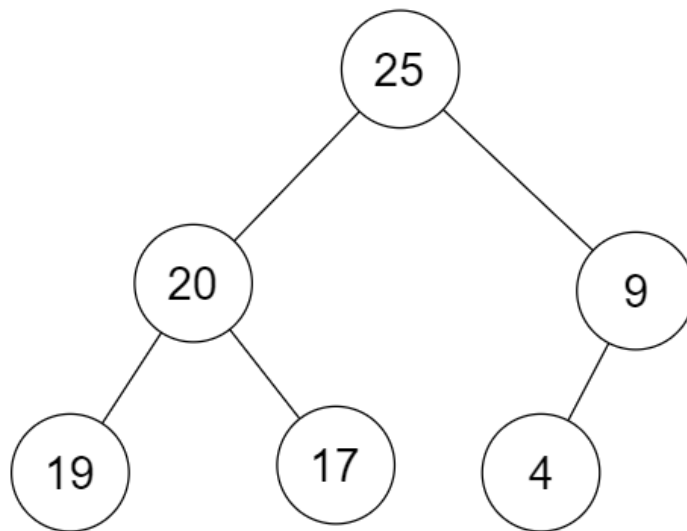


Correct!☐ No☒ Yes**Question 33****1 / 1 pts**

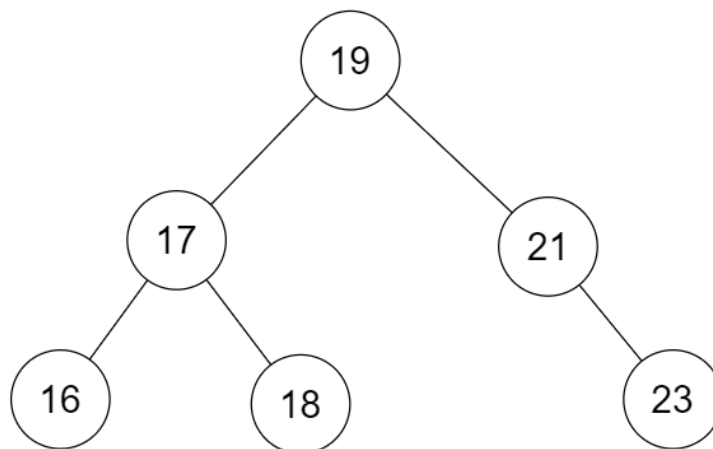
Is the picture below a valid **binary search tree**?

☐ Yes☒ No**Correct!****Question 34****1 / 1 pts**

Is the picture below a valid **maxheap**?

☐ No☒ Yes**Correct!****Question 35****2 / 2 pts**

Is each statement about this **binary search tree** true or false?

**Correct!**

Correct!**The bst is complete.**

false

**The bst is full.**

false



Other Incorrect Match Options:

- true

Question 36**2 / 2 pts**

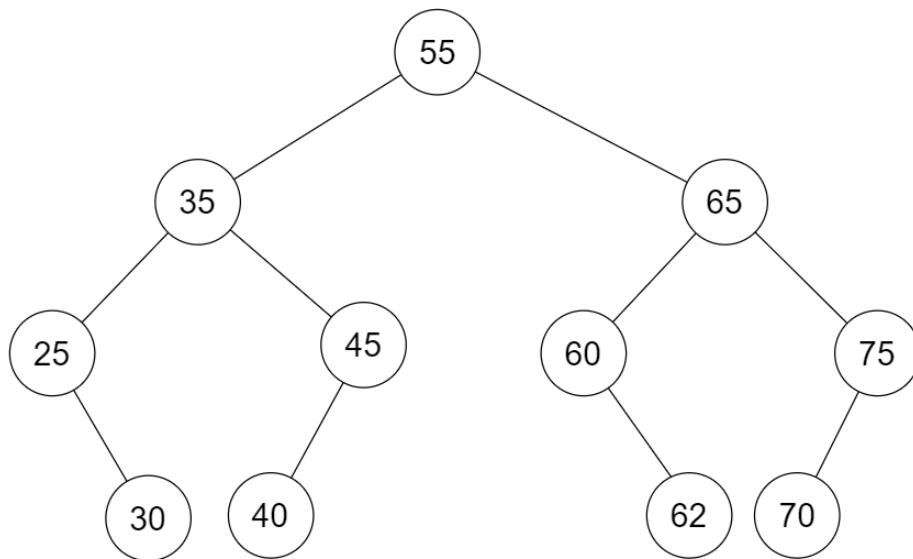
Use the **binary search tree** below and evaluate how you could remove the root node 55.

For each possible removal, state whether it is a valid removal approach **using one of the two algorithms from the textbook (and lecture notes and videos)** discussed in [Module 12](#).

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, the root has not yet been removed.)

**Correct!****remove 55 and replace
with 30**

invalid

**Correct!****remove 55 and replace
with 40**

invalid

**Correct!****remove 55 and replace
with 25**

invalid

**Correct!****remove 55 and replace
with 45**

valid

**Correct!****remove 55 and replace
with 62**

invalid

**Correct!****remove 55 and replace
with 70**

invalid

**Correct!****remove 55 and replace
with 75**

invalid

**Correct!****remove 55 and replace
with 60**

valid

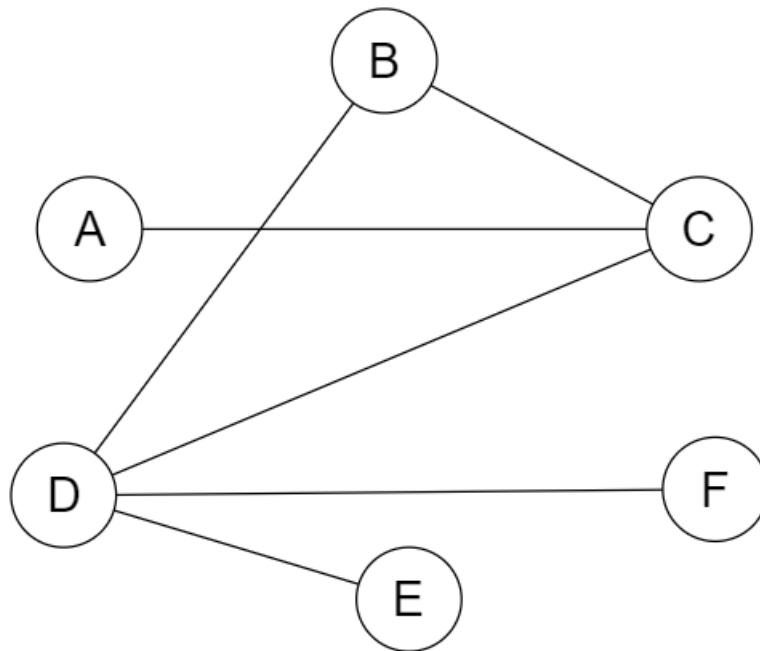


Section 8: Multiple Choice- Graphs

Question 37

3 / 3 pts

Is the graph below each of the following?



Correct!

directed

no



Correct!

weighted

no



Correct!

connected

yes



Correct!

complete

Correct!

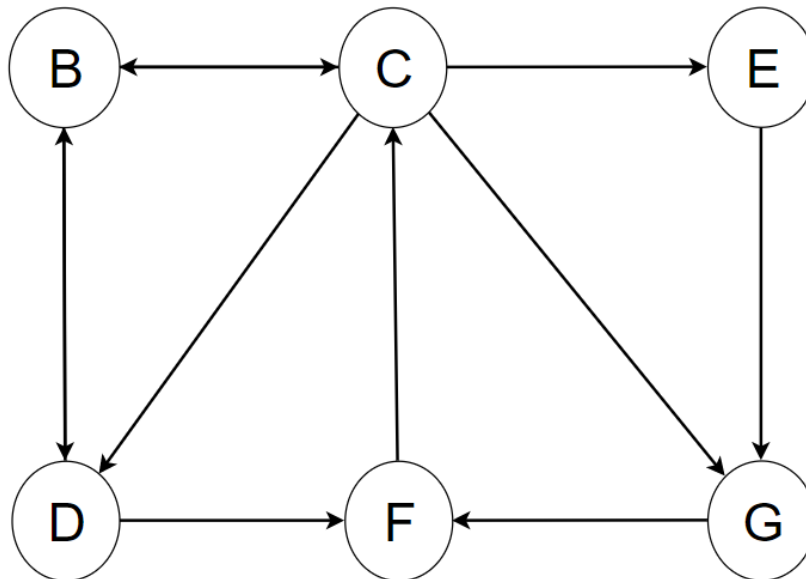
no

**cyclic**

yes

**Question 38****2 / 2 pts**

You are trying to find the **shortest path** from D to C in the directed graph below. Which of the following statement is true?



☐ D-B-C is the only shortest path from D to C.

Correct!

D-B-C is a shortest path and there are other valid shortest paths from D to C.

☐ D-B-C is not a shortest path from D to C.

Question 39

1 / 1 pts

You could find a topological order for the graph above.

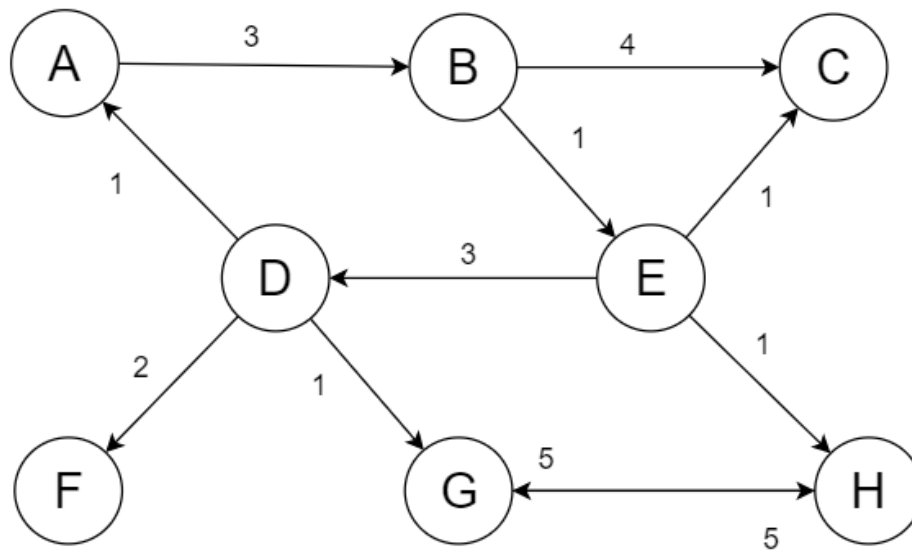
☐ True☒ False

Correct!

Question 40

2 / 2 pts

What is the cost of the **cheapest** path from vertex B to C in the directed, weighted graph below?



Correct!

2

Correct Answers

2 (with margin: 0)

Quiz Score: **97** out of 100

