Final Exam Part 1, Multiple Choice and Matching, 50 points

Started: Apr 24 at 3pm

Quiz Instructions

Exam Instructions:

- Signing and/or taking this exam signifies you are aware of and in accordance with the Academic Honor Code of Georgia Tech and the Georgia Tech Code of Conduct.
- The final exam is in two parts for a total of 2 hours 50 minutes of test taking time. Each of the two parts of the final exam is 1.5 hours in length. (You may or may not need the full hour and half to complete each part.) The extra time has been allotted for any transition time you find necessary. The two parts must completed in the order that they appear in the module.
- There is an ***24 hour*** window, 12:01 am to 11:59 pm EDT, on exam day in which to take your final exam. Canvas will automatically close the exam when time is up. Be sure to enter your answers in Canvas before it closes.
- Allow a full 3.5 hours for you final exam.
- We suggest beginning your exam before 8:30 pm in order to finish on time.
- All exams are administered via Canvas. There is no proctoring during this exam.
- You are responsible for a working a machine and internet connection in order to complete the exam.
- The final is open notes and open book. You are not to consult another individual at all during the final.
- All code must be in Java.
- Efficiency matters. For example, if you code something that uses O(n) time or worse when there is an obvious way to do it in O(1) time, your solution may lose credit. If your code traverses the data 5 times when once would be sufficient, then this also is considered poor efficiency even though both are O(n).
- Style standards such as (but not limited to) use of good variable names and proper indentation is always required.
- Comments are not required unless a question explicitly asks for them.
- ADDITIONAL Instructions:
    - Because this is an online exam, many of the questions have specific instructions for formatting your answers. These instructions will be preceded by Answer Format in all cases. Make sure to read all questions carefully so that you do not miss anything important!

Question 1 — 2 pts

Below is a Failure Table for use during the KMP Algorithm. Which of the pattern choices will produce this Failure Table?

i	0	1	2	3	4	5	6	7	8	9

f[i]	0	0	0	1	2	3	4	1	2	3

- baababababa
- aaccaacaac
- abaacabaab
- acabaacaca
- abbabbaabb
- cbccbacbcb

Question 2

2 pts

Five (5) elements are added to a mystery data structure. The elements are added in this order: elementA first, elementB second, elementC third, elementD fourth, and elementE last. After calling the remove method TWICE on the mystery data structure, the only three elements that remain are elementB, elementD, and elementE. What could the mystery data structure be?

- Deque
- Queue
- Binary Heap
- Stack

Question 3

2 pts

You are the proud owner of Jack's Tree Farm with quite an extensive inventory of trees and related wares for sale. Being a savvy entrepreneur, you want to make your business as efficient as possible. To do this, you assign each of your trees a unique inventory ID number,

and store that ID along with other tree information such as height, price, and age. What structure is best suited to store the Tree Farm data in order to make lookup, addition, and removal operations based on inventory ID numbers as efficient as possible?

- (●) HashMap
- ( ) Graph
- ( ) Binary Search Tree
- ( ) Queue
- ( ) AVL Tree

Question 4

2 pts

You are a bioinformatics intern working on analyzing DNA sequences for your lab. DNA sequences are represented by strings of characters representing nucleobases. Each nucleobase is represented by a single, unique alphabetical character. The number of possible unique characters is fairly small at only 4 types of nucleobases. You are tasked with identifying the occurrences of a small, specific DNA sequence among hundreds of sample sequences that are each 1000+ characters long. What pattern matching algorithm is best suited to perform this task?

- ( ) Quickselect
- ( ) Dijkstra's
- ( ) Brute Force
- ( ) KMP
- (●) Boyer Moore

Question 5

2 pts

Starting from an empty HashMap with backing array of **length 8** that uses **linear probing** for collision resolution, perform the operations below and choose the resulting backing array.

The elements below are keys, and their corresponding values are omitted. The hash function is the key itself. The load factor is 1.0.

1. Add 0
2. Add 2
3. Add 8
4. Add 9
5. Remove 2
6. Add 16

A)

0	8		9				16

B)

0	8	16	9				

C)

0	8		9	16			

D)

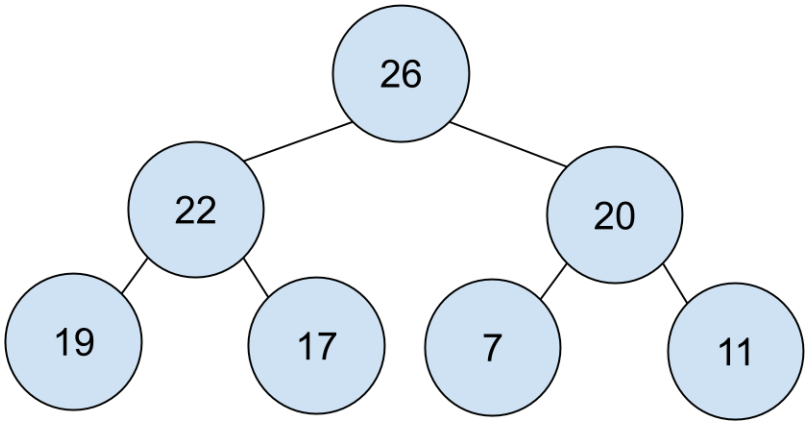
0	9	16					8

- (●) B
- ( ) C
- ( ) D
- ( ) A

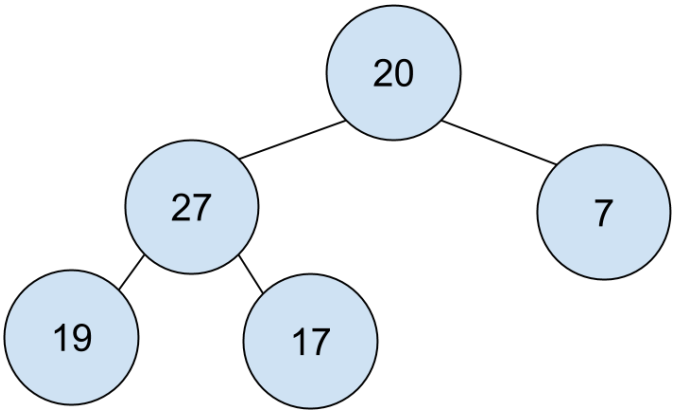
Question 6

2 pts

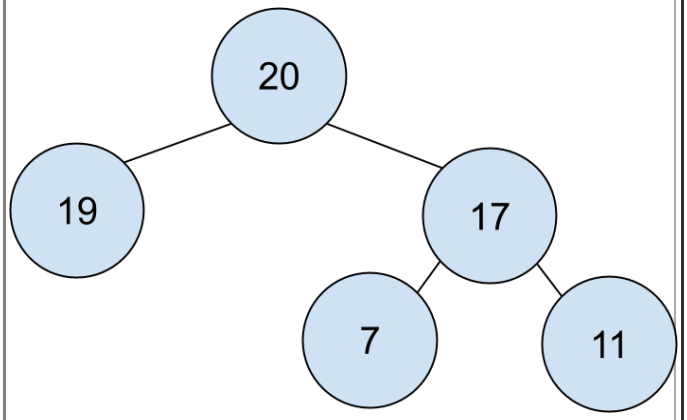
Given the following initial **Max** Heap, perform two removals and choose the resulting Heap from the options in the table.



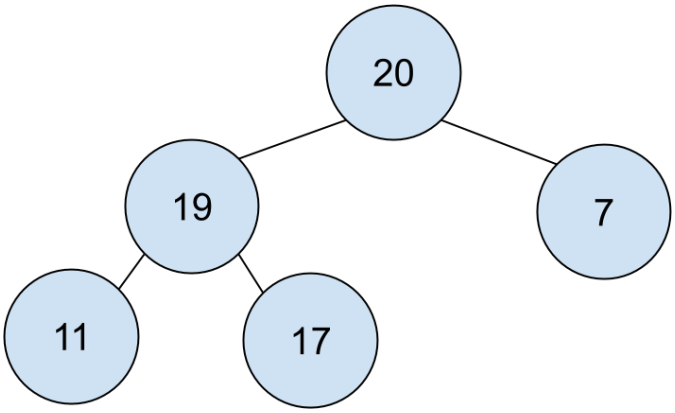
A)



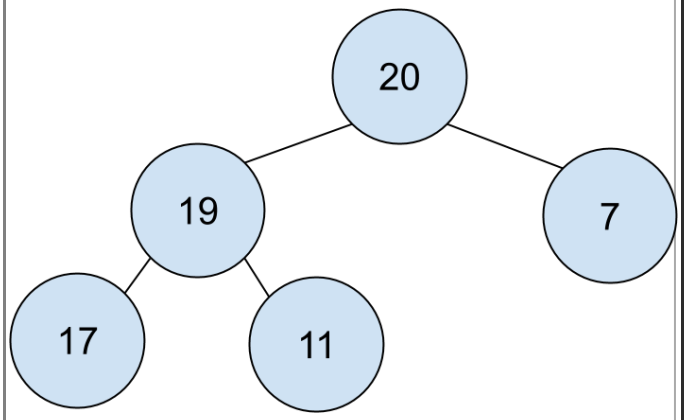
B)



C)



D)



- [x] C
- [ ] D
- [ ] A

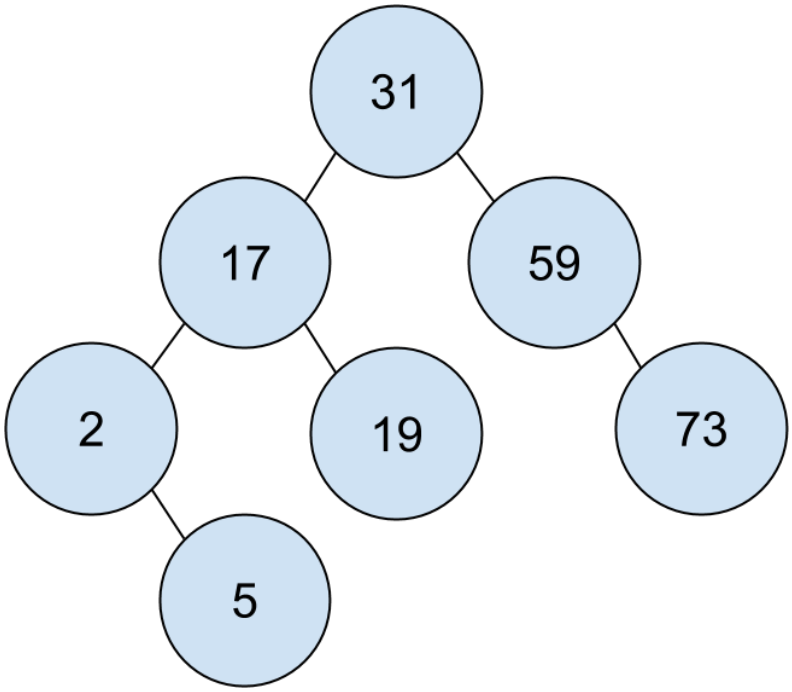
B

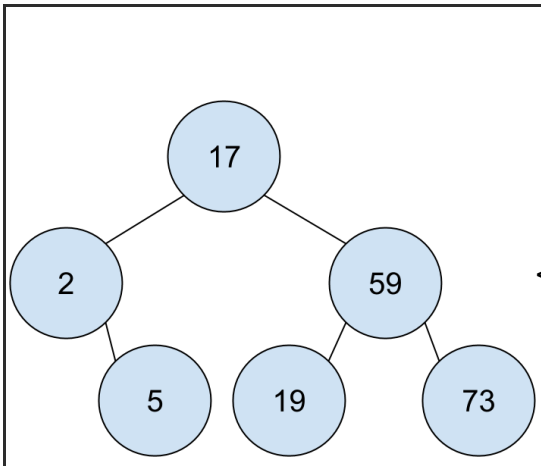
Question 7

2 pts

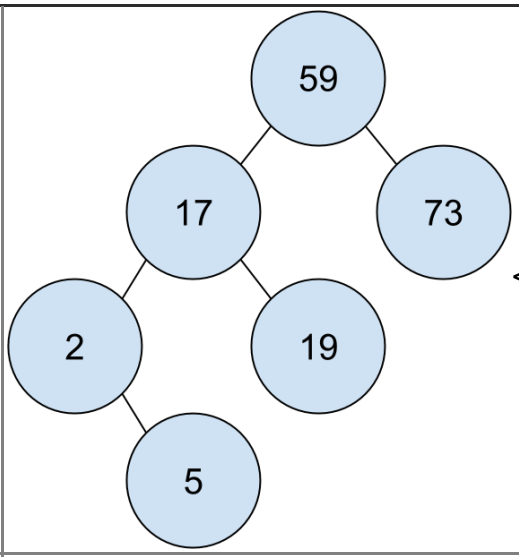
What is the result of removing **31** from the AVL Tree displayed below? (Use the predecessor, if necessary)

Starting AVL:

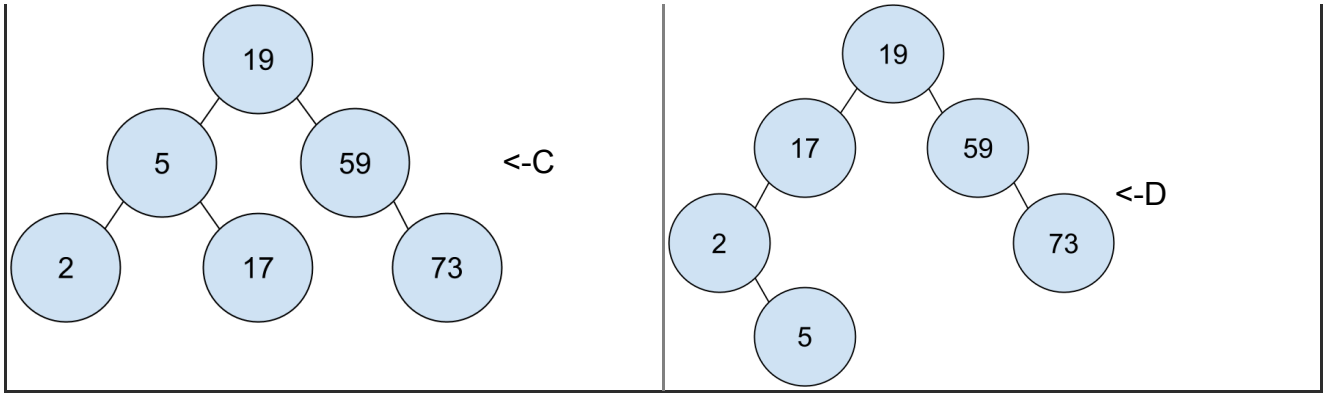




<-A



<-B



- [x] C
- [ ] A
- [ ] D
- [ ] B

Question 8

2 pts

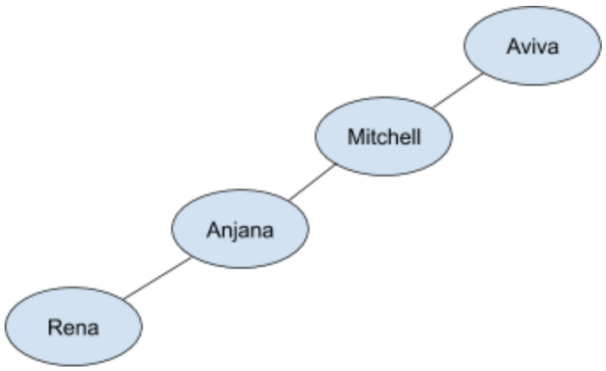
Which of the following statements about Linked Lists and BSTs is **TRUE**?

- [ ] It is always more efficient to add an arbitrary element to a BST than to add an arbitrary element at an arbitrary index to a Linked List.
- [ ] The average runtime of searching for a given element in both a Linked List and a BST is O(log n).
- [ ] None of these choices.
- [ ] A BST can hold more data elements than a Linked List.
- [ ] The runtime of completing a traversal of both a Linked List and a BST is O(n).

Question 9

2 pts

You are given the following binary tree and a traversal output of the tree. **Choose ALL traversals that could produce the output**. If none apply, then leave all check boxes empty.



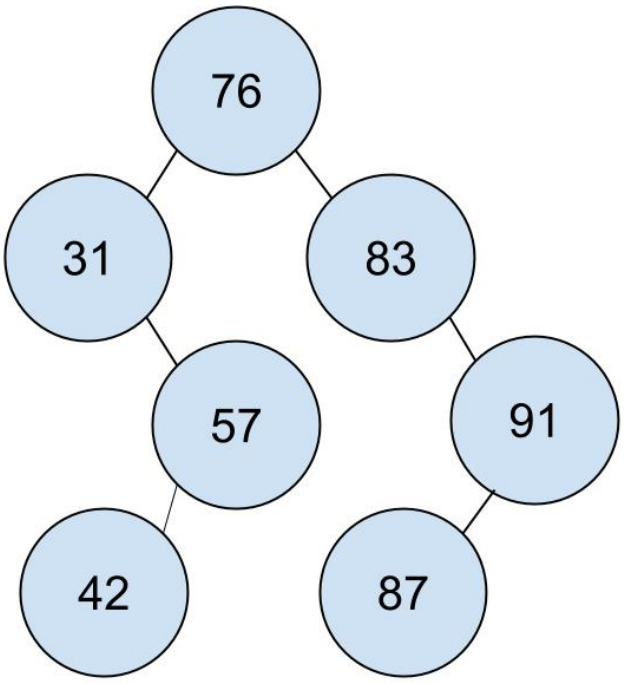
Output: [Aviva, Mitchell, Anjana, Rena]

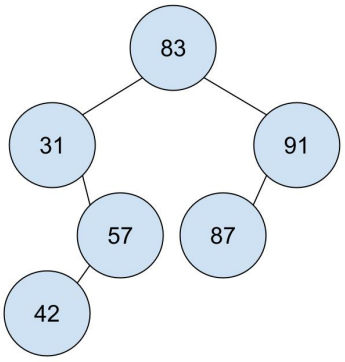
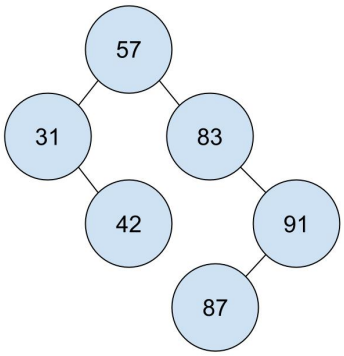
- [ ] KMP
- [ ] Inorder
- [ ] Preorder
- [ ] Postorder
- [ ] Levelorder

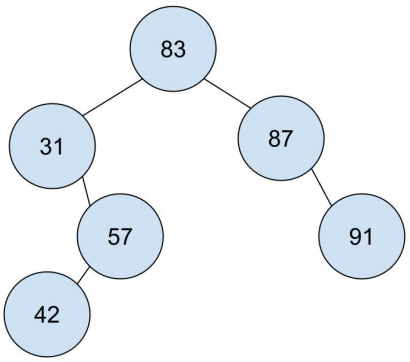
Question 10

2 pts

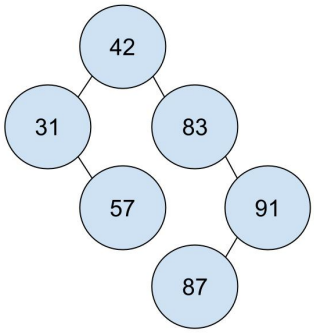
What is the result of removing **76** from the BST displayed below? If necessary, use the **predecessor**.


76
31
83
57
91
42
87

 83 31 91 57 87 42 <- A	 57 31 83 42 91 87 <- B



<- C



<- D

- B
- C
- D
- A

Question 11

2 pts

Perform **2** complete iterations of LSD Radix sort on the initial array shown below. Then select the choice that matches the resulting array. The first row displays the indices of the array, the second (bottom) row displays the actual array elements that are being sorted.

0	1	2	3	4	5	6	7
6177	6471	1771	6715	3635	2415	6905	4240

0	1	2	3	4	5	6	7
4240	6471	1771	6715	3635	2415	6905	6177

0	1	2	3	4	5	6	7
4240	1771	6471	3635	6715	6905	2415	6177

0	1	2	3	4	5	6	7
6905	2415	6715	3635	4240	1771	6471	6177

0	1	2	3	4	5	6	7
6905	6715	2415	3635	4240	6471	1771	6177

Question 12

2 pts

Which of the following sorting algorithms are stable?

- [ ] Insertion Sort
- [ ] Cocktail Shaker Sort
- [ ] Bubble Sort
- [ ] LSD Radix Sort
- [ ] Merge Sort

Question 13

2 pts

What will the array below look like after **THREE** iterations of Bubble Sort are performed?

```
Initial Array: [110, 38, 102, 63, 44, 56, 101, 43]
```

- [ ] `[38, 44, 56, 63, 43, 101, 102, 110]`
- [ ] `[38, 63, 44, 56, 101, 43, 102, 110]`
- [ ] `[38, 43, 110, 44, 102, 63, 56, 101]`
- [ ] None of these choices.
- [ ] `[38, 43, 44, 56, 63, 101, 102, 110]`

Question 14

2 pts

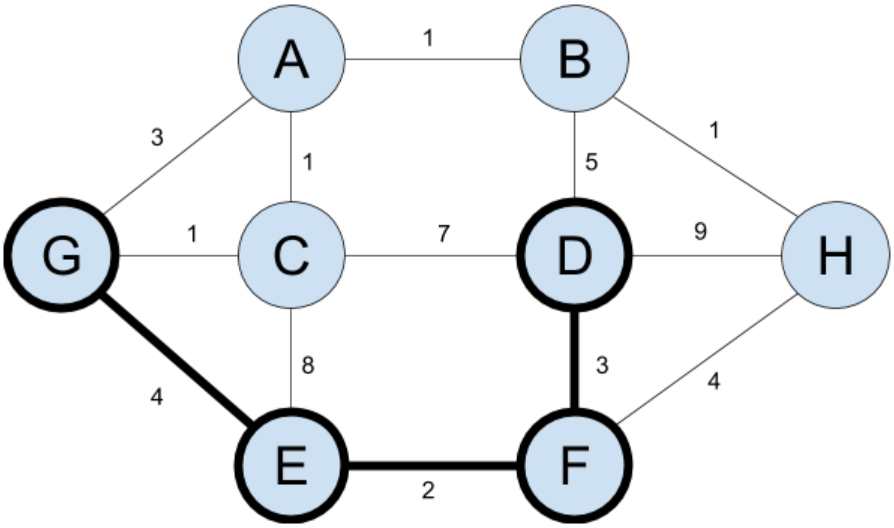
Which of the following is **TRUE** about 2-4 trees?

- After a transfer (aka. rotation) operation to balance the tree, at least one fusion operation must occur to ensure that the final tree is balanced.
- A node in a 2-4 tree can have up to 6 data elements.
- The best case runtime of 2-4 tree remove operation is O(n).
- Promotion is the 2-4 tree balancing strategy used only during a 2-4 tree add operation.
- None of these choices.

Question 15

2 pts

You are trying to find an MST for the graph below, so you decide to run Prim's Algorithm. Highlighted in the graph below are the edges that have already been added to the MST at this point in the algorithm. Which of the following edges would be added to the MST next if you continue with the algorithm?



- GA
- GC
- EC
- AB
- FH

Question 16

2 pts

For the algorithm listed below, determine the time complexity of the implementation, and choose from the selections provided. Make sure you choose the tightest Big-O upper bound possible for the operation.

When performing the **Breadth First Search** algorithm on a dense graph with |V| vertices and |E| edges that has an adjacency list implementation, what is the **worst case** time complexity of the algorithm?

- O(|V|^2)
- O(|E|)
- O(|E|^2)
- O(|V|)

- O(|V| + |E|)

Question 17

2 pts

For the algorithm listed below, determine the time complexity of the implementation, and choose from the selections provided. Make sure you choose the tightest Big-O upper bound possible for the operation.

When performing the **Depth First Search** algorithm on a dense graph with |V| vertices and |E| edges that has an adjacency list implementation, what is the **worst case** time complexity of the algorithm?

- O(|V|)
- O(|V|^2)
- O(|E|^2)
- O(|V| + |E|)
- O(|E|)

Question 18

2 pts

For the algorithm listed below, determine the time complexity of the implementation, and choose from the selections provided. Make sure you choose the tightest Big-O upper bound possible for the operation.

Given a graph with |V| vertices, |E| edges, an adjacency list representation, and a populated Min Heap Priority Queue. What is the time complexity to remove each vertex from the Priority Queue?

- O(|E| + |V|log|V|)
- O(|V| log |V|)
- O(|E| log |V|)

- O((|V| + |E|) log |V|)
- O(|E| + |V|)

Question 19

2 pts

For the algorithm listed below, determine the time complexity of the operation, and choose from the selections provided. Make sure you choose the tightest Big-O upper bound possible for the operation.

When performing the Boyer Moore algorithm on text of length *n* and a pattern of length *m* where every character in the text is not in the alphabet of the pattern, what is the worst case time complexity of the algorithm?

- O(m + n)
- O(mn)
- O(n)
- O(m)
- O(m + n/m)

Question 20

2 pts

For the algorithm listed below, determine the time complexity of the operation, and choose from the selections provided. Make sure you choose the tightest Big-O upper bound possible for the operation.

What is the worst case of searching for the last occurrence of a pattern of length *m* in a string of length *n* using the KMP algorithm?

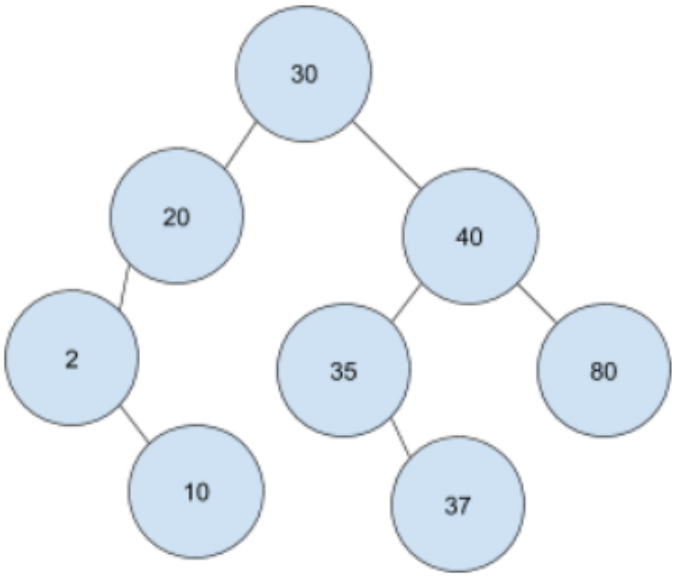
- O(m + n)
- O(m)
- O(mn)

- O(n)
- O(m + n/m)

Question 21

2 pts

Given the following tree:



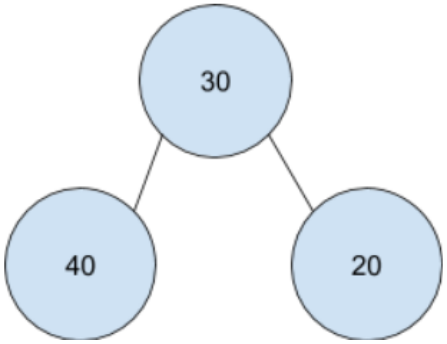
Select **all of** the options below that describe the tree.

- [ ] BST
- [ ] AVL
- [ ] Heap
- [ ] Binary Tree

Question 22

2 pts

Given the following tree:



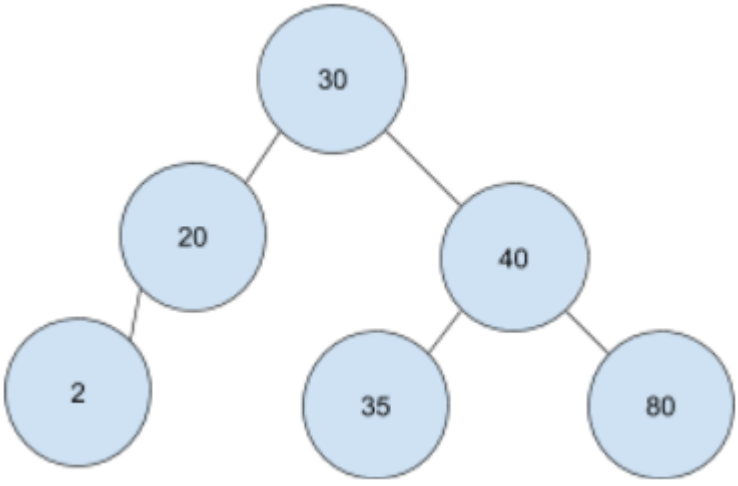
Select **all of** the options below that describe the tree.

- [ ] Binary Tree
- [ ] AVL
- [ ] BST
- [ ] Heap

Question 23

2 pts

Given the following tree:



Select **all of** the options below that describe the tree.

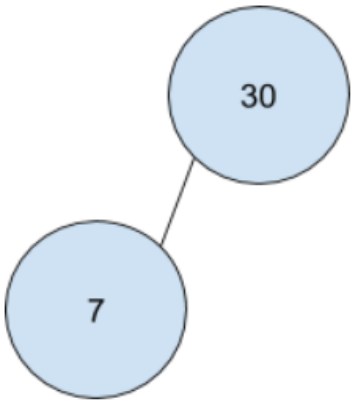
- [ ] AVL
- [ ] Binary Tree
- [ ] Heap

- [ ] BST

Question 24

2 pts

Given the following tree:



Select **all of** the options below that describe the tree.

- [ ] AVL
- [ ] BST
- [ ] Heap
- [ ] Binary Tree

Question 25

2 pts

The following array **has had 2 full iterations of Cocktail Shaker Sort** performed on it. The first row displays the indices of the array, the second (bottom) row displays the actual array elements that have been partially sorted.

0	1	2	3	4	5	6
9	16	49	22	37	75	87

Select from the below options the **original array before 2 full iterations of Cocktail Shaker Sort** were performed.

0	1	2	3	4	5	6
37	49	22	9	16	87	75

0	1	2	3	4	5	6
22	9	87	16	75	49	37

0	1	2	3	4	5	6
22	9	87	75	16	37	49

0	1	2	3	4	5	6
75	87	49	22	16	9	37

Question 26

1 pts

For at least one TA that was particularly helpful to you this semester, write the name of a song or write them an original song. If you write an original song, do not write explicit lyrics or you will not get credit. Original songs that surpass the expectations of the TAs are eligible to receive **an additional point**. You can write the name of the TA(s) your song is for in the entry box below, and write the name of the song under the TA name(s). The final exam is **not** an appropriate medium for making **advances** or **inappropriate comments** towards a teaching assistant. Inappropriate submissions will earn a **0** for the entire exam.

Head TA - Adrianna Brown

Senior TA/B3 Grading - David Wang

Senior TA/B3 Grading - Rodrigo Pontes

Online Head TA - Caroline Kish

O1 - Jacob Allen

O1 - Landon Ryan

O1 - Isaac Weintraub

A1 - Paige Ryan

A1 - Tillson Galloway

A2 - Yotam Kanny

A2 - Aviva Kern

A3 - Destini Deinde-Smith

A3 - Siddu Dussa

A4/A5/A6/GR1 - Neha Deshpande

A4/A5/A6/GR1 - Isaac Tomblin

B1 - Reece Gao

B1 - Rena Li

B2 - Miguel de los Reyes

B2 - Smita Mohindra

B3 - Sanjana Tewathia

B4 - Mitchell Gacuzana

B4 - Eunseo Cho

B5 - Elena May

B5 - Ivan Leung

B6 - Anjana Nandagopal

B6 - Alex McQuilkin

C1 - Brooke Miller

C1 - Cliff Panos

C2/GR2 - Brandon Vu

C2/GR2 - Ila Vienneau

C3/C4 - Nick Worthington

C3/C4 - Keely Culbertson

HTML Editor

12pt Paragraph

0 words

No new data to save. Last checked at 3:16pm

Submit Quiz