Data Structures Midterm

NAME: _____

1 Directions

- Read carefully, write legibly, check work and complete in 1 hour.
- Before you start, visit the restroom if necessary.
- Turn off your phone, do not use a calculator or talk with other students.
- If you do not understand a question, raise your hand and wait for the instructor to come to you.
- Do not ask questions like: "Am I doing this right?"

2 Abstract Data Types (ADTs)

1. Match these abstract data types to their definition.
   A. Stack B. Bag C. Set D. Queue E. Dictionary (Map) F. Sequence

   (a) An ordered collection of elements.

   (a) _____

   (b) Last in, first out.

   (b) _____

   (c) An unordered collection of elements.

   (c) _____

   (d) One to one association between unique keys to values.

   (d) _____

   (e) First in, first out.

   (e) _____

   (f) An unordered collection of unique elements.

   (f) _____

2. Using the same options above, which ADT does depth-first search use?

   2. _____

3. Using the same options above, which ADT does breadth-first search use?

   3. _____

3 Big O

4. Algorithm X has runtime complexity $O(\log n)$ operations. Roughly how many more operations are required for X to process 256 elements compared to processing 128 elements?

   A. 1 B. 2 C. 7 D. 8 E. 128

   4. _____

   Choose among the following runtime complexities for the remaining questions in this section.

   A. $O(\log n)$ B. $O(n^2)$ C. $O(1)$ D. $O(2^n)$ E. $O(n)$

5. For large n, which of these options is the fastest?

   5. _____

6. For large n, which of these options is the second fastest?

   6. _____

7. For large n, which of these options is the second slowest?

   7. _____

8. An algorithm that takes the same time to run, no matter how much data is given.

   8. _____

9. An algorithm that takes twice as long to run for each additional datum given to it.

   9. _____

10. An algorithm that takes twice as long to run if the amount of data doubles.

    10. _____

11. Linear.

    11. _____

12. Exponential.

    12. _____

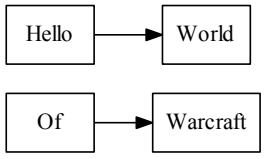
13. Constant.

    13. _____

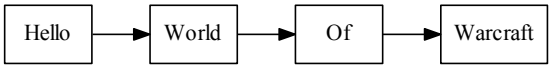
4 Linked Lists

Suppose we have the following implementation of a list:

```
class MyList {
    Node start, end; int size;
    static class Node {
        String data; Node next;
    }
}
```



Before `concatenate`



After `concatenate`

Choose among these implementations of `concatenate` to answer the following questions.

A.
```
public void concatenate(MyList otherList) {
    end.next = otherList.start;
    size += otherList.size;
}
```

B.
```
public void concatenate(MyList otherList) {
    for (Node current = start; current != null; current = current.next) {
        if (current.next == null) {
            current.next = otherList.start;
            size += otherList.size;
        }
    }
}
```

C.
```
public void concatenate(MyList otherList) {
    for (Node current = otherList.start; current != null; current = current.next) {
        end.next = new Node();
        end.next.data = current.data;
        end = end.next;
        size++;
    }
}
```

D. All of the above

E. None of the above

14. Which implementation is the fastest, regardless of whether it works?

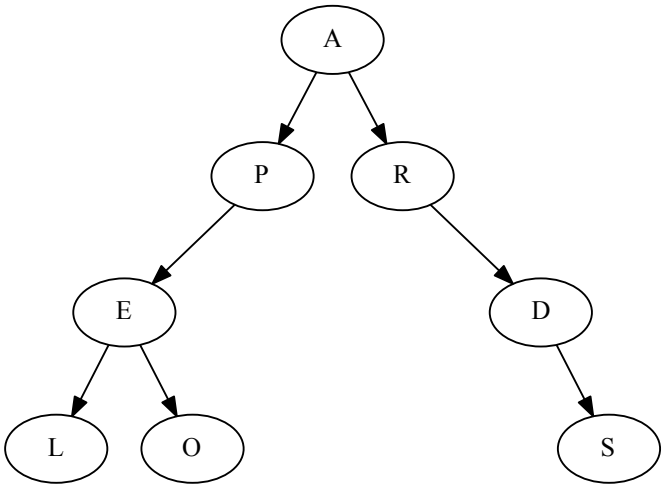
14. _____

15. Which implementation has potential bugs?

15. _____

5 Binary trees

Refer to this binary tree to answer questions in this section.



Choose among these options.

A. LEOPARDS B. LOSER PAD C. APE LORDS D. APR ED LOS E. LOEPSDRA

16. What is the infix traversal of this tree?

16. _____

17. What is the prefix traversal of this tree?

17. _____

18. What is the postfix traversal of this tree?

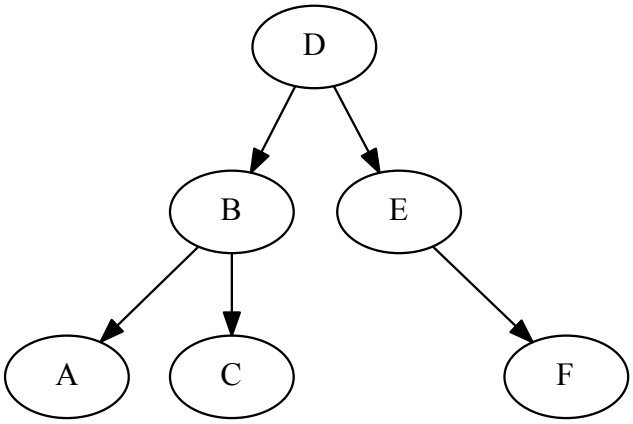
18. _____

19. Which is a breadth-first traversal of this tree?

19. _____

6 Binary search trees

Refer to this binary search tree to answer questions in this section.



20. How many value comparisons[1] are necessary to determine whether G is in this tree?

20. _____

21. Suppose we delete D. What node becomes the root?

21. _____

22. Suppose, after deleting D, we then delete B. What replaces B?

22. _____

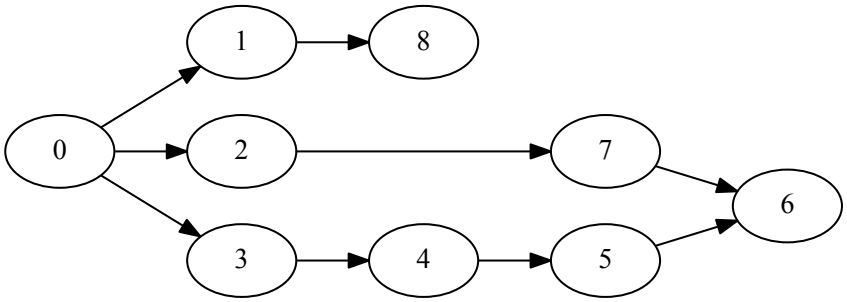
23. Suppose we delete E. What replaces it?

23. _____

[1] Exclude pointer comparisons from your number.

7 Graphs

Refer to this graph to answer questions in this section.



Choose among these options.

A. 0, 1, 2, 3, 4, 5, 6, 7, 8
B. 6, 5, 4, 3, 7, 2, 8, 1, 0
C. 0, 1, 8, 2, 7, 6, 3, 4, 5
D. 0, 1, 2, 3, 8, 7, 4, 6, 5

24. Which of these is a depth-first search?

24. _____

25. Which of these is a breadth-first search?

25. _____

Check your work.

8 Extra challenges

Nothing in this section counts for credit (not even extra credit). Do not attempt unless you're done. If you can complete these questions, kudos to you; if not, think about them a bit until we meet again.

26. What bug(s) did you find in the `concatenate` implementation(s), if any?

27. One binary tree can represent an anagram pair (e.g., mars, rams) by swapping the traversal algorithms (e.g., from prefix to infix). Are there any anagram pairs that one binary tree cannot represent by changing the traversal algorithm alone?