

CS151 Intro to Data Structures

Midterm Review

Announcements

Exam Wednesday

HW04 Due Sunday

- Deadline extended
- Binary trees will be on your exam in some capacity

Grade Breakdown Emails Sent

Exam Format

- 1 page 8.5/11in cheat sheet allowed (front and back)
- Format:
 - 11 points T/F questions
 - 8 points reading and understanding code
 - 51 points open ended programming
 - I won't be harsh on syntax grading
 - I won't try to compile your code :)
 - Be careful with types! You'll get points deducted if you code doesn't type check
- Tips:
 - For the coding portion, DRAW! It will help you

Topics

Data Structures

- Arrays
- Expandable Arrays
- Stacks
- Queues
- Linked Lists
- Binary Trees

Other concepts:

- Generics
- Iterators
- Big-O analysis
- OOP & Inheritance
- Interfaces

Practice Problems (T/F)

1. A Linked list of 5 elements uses more memory than a java array of the same 5 elements
 - a. TRUE
2. If an exception occurs in a catch block, it will be handled gracefully and will not crash the program
 - a. FALSE
3. Binary search on a sorted linked list can be implemented in $O(\log n)$ time
 - a. FALSE
4. A stack with $O(1)$ push, pop, and peek operations can be implemented with either a linked list or an array
 - a. TRUE
5. A dequeue is a FIFO data structure. An element cannot be removed until the elements which have been added before it have been removed.
 - a. FALSE

Practice Problems (reading code)

```
/**
 * @precondition: ll contains at least 1 element.
 */
public static int mystery(LinkedList<Integer> ll) {
    int count = 0;
    Node n = ll.head.next;
    Node n2 = n.next;

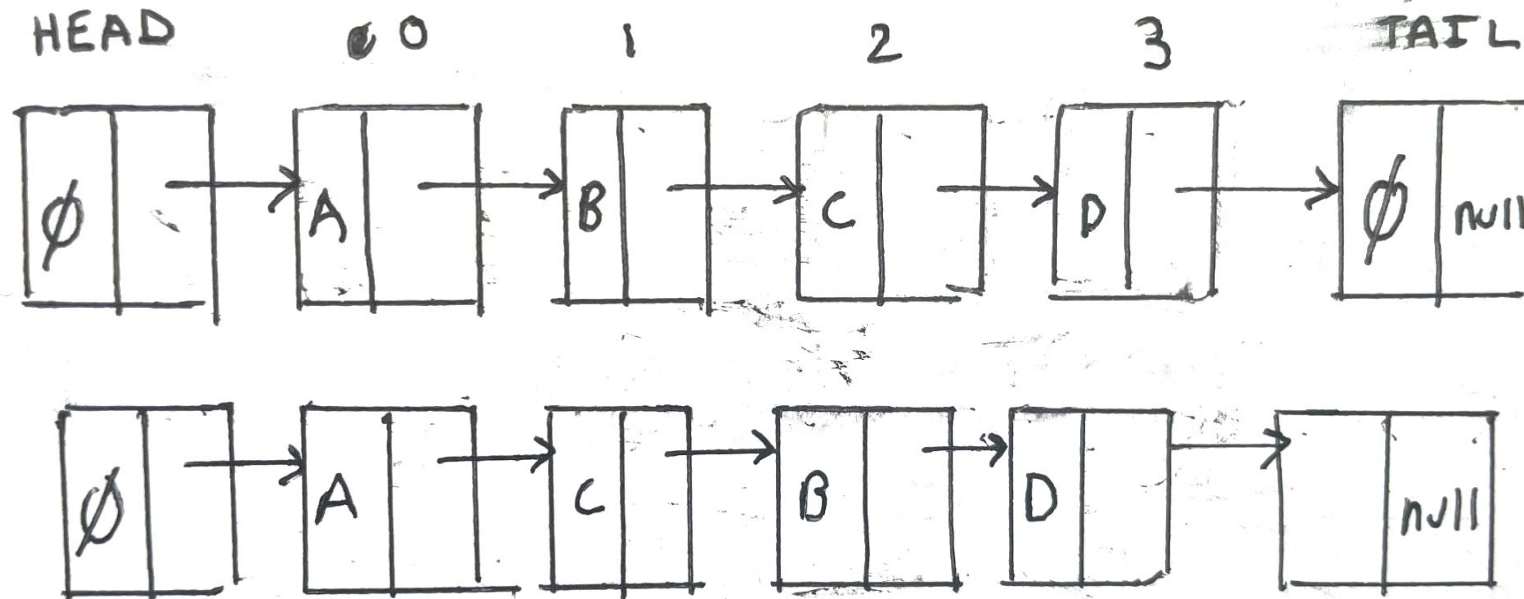
    while (n != null && n2 != null) {
        if (!n.data.equals(n2.data)) {
            return count;
        } else {
            count += 1;
        }
        n = n.next;
        n2 = n2.next;
    }
    return count;
}
```

1. What is the runtime complexity in the best case? Use big o notation. Give an example input that would achieve this runtime.

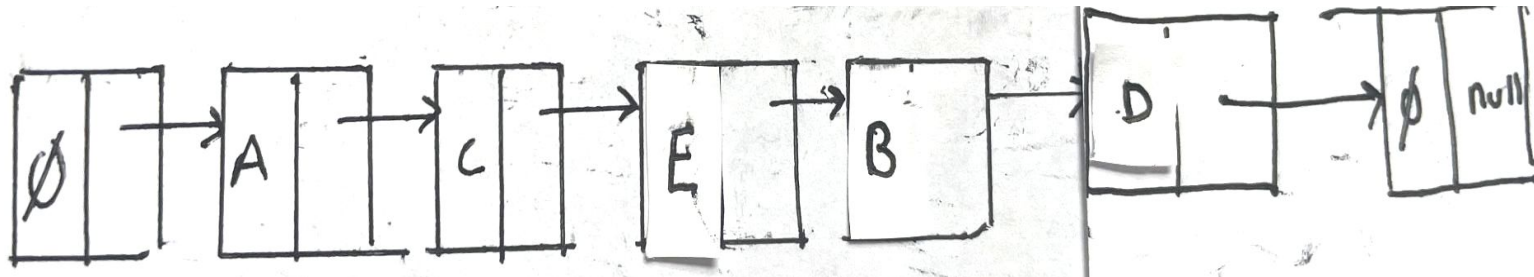
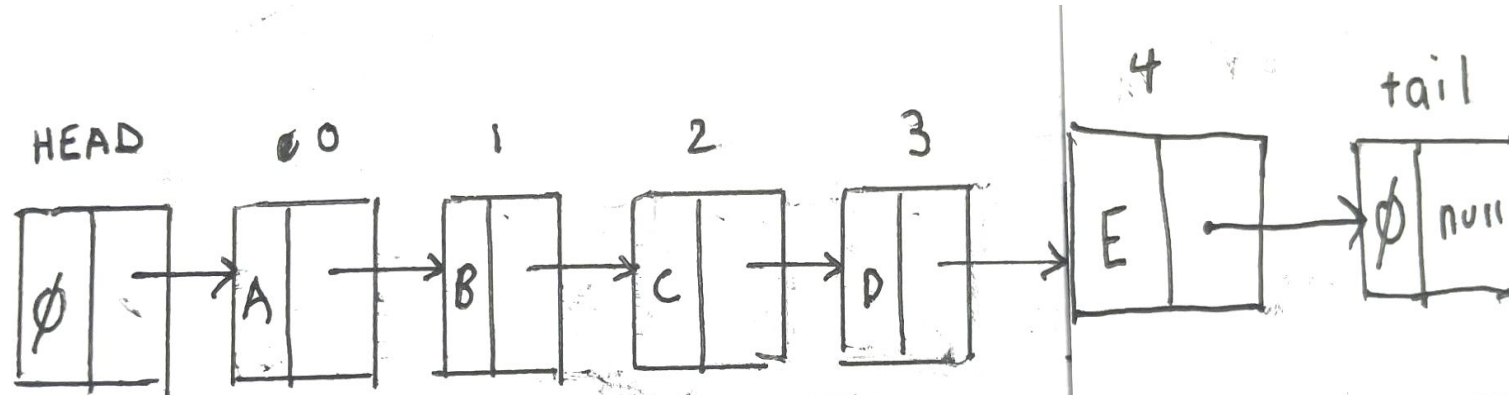
2. What is the runtime complexity in the worst case? Use big o notation. Give an example input that would achieve this runtime.

Practice Problems - Coding Question

`rearrangeEvenOdd` modify the linked list in such a way that nodes with even indices (0, 2, 4, ...) appear before nodes with odd indices (1, 3, 5, ...). Ensure that the rearrangement is done in-place.



Coding Question - Linked Lists



Coding Question - Linked Lists

Edge cases!

Head -> Tail

Head -> A -> Tail

Head -> A -> B -> Tail

Do we handle these?

Coding Question - Linked Lists

Runtime complexity?

Memory complexity?

Coding Question - Stacks

You are given a string containing a combination of square brackets `[]`, curly braces `{ }`, and parentheses `()`. **Use a stack** to determine if the input string is valid in terms of bracket balancing.

`"[{ }]"` \Rightarrow True

`"[{]]"` \Rightarrow False

`"{ [()] }"` \Rightarrow True

`"{ [(]) }"` \Rightarrow False

`"[{ }]]"` \Rightarrow False

Coding Question #2

Runtime complexity?

Memory complexity?

Coding Question 3:

name,	intensity,	kind
splash,	50,	water
fireball,	100,	fire
ignite,	15,	fire
terraform,	20,	earth

Design four classes: `water`, `fire`, `earth`, and `spell`.

Write a method **`castSpells`** takes two `Spells` and returns a `int` indicating which spell won

Water always beats fire regardless of intensity

Otherwise, compare intensity

Data Structures

Expandable Arrays

search

- How do we implement?
- Best case?
- Worst case?

insertion

- How do we implement?
- Best case?
- Worst case?

removal

- How do we implement?
- Best Case?
- Worst Case?

LinkedList

search

- How do we implement?
- Best case?
- Worst case?

insertion

- How do we implement?
- Best case?
- Worst case?

removal

- How do we implement?
- Best Case?
- Worst Case?

Stacks - LinkedList implementation

Search?

- How do we implement?

Insertion?

- How do we implement?
- Best Case?
- Worst Case?

Removal?

- How do we implement?
- Best Case?
- Worst Case?

Stacks - Array implementation

Search?

- How do we implement?

Insertion?

- How do we implement?
- Best Case?
- Worst Case?

Removal?

- How do we implement?
- Best Case?
- Worst Case?

Queues - LinkedList implementation

Search?

- How do we implement?

Insertion?

- How do we implement?
- Best Case?
- Worst Case?

Removal?

- How do we implement?
- Best Case?
- Worst Case?

Queues - Array implementation

Search?

- How do we implement?

Insertion?

- How do we implement?
- Best Case?
- Worst Case?

Removal?

- How do we implement?
- Best Case?
- Worst Case?

Binary *SEARCH* Trees

(assume balanced)

Search?

- How do we implement?
- Best Case?
- Worst Case?

Insertion?

- How do we implement?
- Best Case?
- Worst Case?

Removal?

- How do we implement?
- Best Case?
- Worst Case?

Other Concepts

Generics

What is a generic?

How do we declare a generic class?

What can a generic class hold?

Iterators

What methods can we call on iterators?

Advantages / disadvantages of iterators vs loops?

Runtime Complexity

Sort these from fastest to slowest:

- $O(n)$
- $O(n^2)$
- $O(\log n)$
- $O(1)$
- $O(2^n)$