

M15 On-Your-Own Practice Questions

These questions are presented for practice. You are not required to post an answer.

Question 1: Evaluating Recursive Methods

What is the output of the following recursive method calls? I recommend you do **not** simply run them! Instead, trace out what is happening by hand to make sure you understand.

Choice A:

```
System.out.println(recMethod1(5, 1));

recFactorial1(4);
recFactorial2(4);

public int recMethod1(int x, int y) {
    if(x==y) {
        return 0;
    } else {
        return recMethod1(x-1, y) +1;
    }
}

public int recFactorial1(int x) {
    System.out.print(x);
    if(x > 1) {
        return x * recFactorial1(x-1);
    } else {
        return 1;
    }
}

public int recFactorial2(int x) {
    if(x > 1) {
        int fac = x * recFactorial2(x-1);
        System.out.print(x);
        return fac;
    } else {
        return 1;
    }
}
```

Choice B:

```
int[] a = {6, 2, 4, 6, 2, 1, 6, 2, 5};
```

```
System.out.println(recMethod2(a, 2, 0));
System.out.println(recMethod2(a, 2, 9));

System.out.println(recMethod3(a, 0));

public int recMethod2(int[] arr, int b, int j) {
    if(j<arr.length) {
        if(arr[j] != b) {
            return recMethod2(arr, b, j+1);
        } else {
            return 1+recMethod2(arr, b, j+1);
        }
    } else {
        return 0;
    }
}

public int recMethod3(int[] arr, int n) {
    int sum = 0;
    if(n<arr.length-1) {
        recMethod3(arr, n+1);
    } else {
        sum = arr[n];
    }
    return sum;
}
```

Choice C:

```
chainA: 2 -> 3 -> 1 -> 6 -> 4
chainB: 3 -> 5 -> 4 -> 2
chainC: 4
chainD: an empty chain

call recMethod4 with each chain

public void recMethod4(Node<Integer> current) {
    if(current.next!=null) {
        System.out.println(current.data);
        current.data = current.data+1;
        recMethod4(current.next);
        System.out.println(current.data);
    }
}
```

Question 2: Properties of Recursive Methods

1. What features must you have in **all** recursive methods?

2. Are any of the methods in the question above missing any of these requirements (under all conditions or under only certain conditions)?
3. What will happen if you invoke these methods under these conditions?

Question 3: Writing Recursive Methods

- a. take three parameters (a String, a character, and an initial index) and return the number of times the character appears in the string starting at the initial index
- b. take one Node parameter that represents the head of a list and returns a count for how many elements are in the chain
- c. take an array of integers and an integer key and return the number of elements in the array that are greater than the key

Question 4: Tracing Simple Sorts

Trace out following sorting algorithms with the datasets below.

1. insertion sort: 4, 1, 8, 7, 2, 5, 3
2. selection sort: 5, 3, 9, 7, 6, 2, 1
3. Shell sort: 5, 6, 9, 10, 12, 2, 3, 8, 7

Question 5: Search and Sort Properties

1. Can you perform a sequential search on each of the following? What about a binary search?
 - unsorted array, sorted array, unsorted linked list, sorted linked list
2. What is the efficiency (big-oh) of sequential search? binary search?
3. What are the big-ohs of merge sort, quick sort, and radix sort? How do these compare to the big-oh of insertion sort, selection sort, and Shell sort?
4. In quick sort, what is true about the array after a single pass of the partition method?

Question 6: Tracing Sequential Search

Trace a sequential search of the following sorted list:

A D H M X Y

1. Search for E
2. Search for M
3. Search for Z

Trace a sequential search of the following unsorted list:

X A H Y D M

1. Search for E

2. Search for D
3. Search for Z

Question 7: Tracing Binary Search

Trace a binary search of the following sorted list. List the values of first, last, and mid for each pass through the list.

A D H M X Y

1. Search for A
2. Search for E
3. Search for L
4. Search for P
5. Search for Z

Question 8: Tracing Stacks and Queues

1. Trace the contents of a queue after each the following statements:

```
queue.enqueue(2);
queue.enqueue(3);
queue.enqueue(7);
queue.dequeue();
queue.enqueue(queue.getFront());
queue.enqueue(4);
queue.enqueue(queue.dequeue());
queue.getFront();
queue.enqueue(1);
queue.enqueue(5);
queue.enqueue(queue.dequeue());
```

2. Trace the contents of a priority queue after each statement executes.

pq is a priority queue such that lower numbers have higher priority. Assume that the first character of the object specifies its priority. For example, 1a has priority 1. 2b has higher priority than 3a; 2b has the same priority as 2e.

```
pq.add(2d);
pq.add(1c);
pq.add(2a);
pq.add(3b);
pq.add(pq.getFront());
pq.add(1a);
pq.add(pq.remove());
pq.add(3a);
```

3. Trace the contents of a deque after each of the following statements:

```
deque.addToFront(2);
deque.addToBack(3);
deque.addToFront(7);
deque.removeFront();
deque.addToFront(deque.getBack());
deque.addToBack(4);
deque.addToFront(deque.removeBack());
deque.addToBack(deque.getFront());
deque.addToBack(1);
deque.addToFront(5);
```

4. Trace the contents of a stack after each of the following statements is executed:

```
stack.push(1);
stack.push(5);
stack.push(3);
stack.pop();
stack.push(stack.peek());
stack.push(7);
stack.pop();
stack.push(6);
stack.peek();
stack.push(stack.pop());
stack.pop();
```

5. Trace the contents of a stack and queue after the following statements are executed:

```
queue.enqueue(10);
queue.enqueue(5);
stack.push(queue.getFront());
stack.push(7);
queue.enqueue(stack.pop());
queue.enqueue(stack.peek());
stack.push(queue.dequeue());
stack.push(queue.dequeue());
queue.enqueue(queue.dequeue());
```

Question 9: Balanced Parentheses

What is left on the stack when the trace is done and how does this tell you whether or not the parentheses are balanced?

1. $a * (b + c * (d - e) + f)$
2. $(a * (b + (c * d) - e + f)$

3. $a * (b + c * d) - e + f$
4. $(a * b) + (c * d) - (e + f)$

Question 10: Evaluating Post-Fix Expressions

Determine if each of the following is a valid post-fix expression. If it is, evaluate the expression.

1. $3\ 4\ *\ 6\ 2\ /\ +$
2. $3\ 5\ 3\ +\ *\ 6\ /\$
3. $2\ 4\ 8\ 2\ /\ +\ *$
4. $1\ 6\ 9\ +\ 3\ /\ +\ *$

Question 11: Coding Stacks and Queues

For all questions, make sure the code will not crash under special conditions.

1. Client Perspective: Write a complete method that takes a stack as a parameter. Put the top two elements of the stack in a queue. Return the queue. Do not destroy the stack. (Or, if you destroy it, rebuild it.)
2. Client Perspective: Write a complete method that takes a queue as a parameter and returns a new queue that contains the reverse contents. Do not destroy the queue. (Or, if you destroy it, rebuild it.)
3. Client Perspective: Write the same method as #2 for a stack.
4. Write client-level code to count how many times a particular value appears in a queue. Do not destroy the queue. (Or, if you destroy it, rebuild it.) Write the same code from the implementation perspective for an array-based queue and a linked queue.
5. Write the same three methods (one client-level and two implementation-level) as #4 for a stack.

Question 12: Tree Definitions

1. What is a leaf node, root node, and interior node?
2. What is a binary tree? a binary search tree? a heap?
3. What does it mean for a tree to be complete? full?
4. What does it mean for a binary search tree to be balanced?
5. What is the maximum number of nodes in a binary tree with height h ?
6. What is the maximum number of nodes on any given level of a binary tree?
7. What is the efficiency of searching a binary search tree? a binary tree that is balanced?
8. Draw: a) a binary tree that is not a bst; b) a bst; c) a complete binary tree that is not a heap; d) a heap.

Question 13: The Heapform Array

Draw the trees represented by the heapform arrays below. (* represents an empty space in

the array.)

Is each of these tree a heap? If so, it is a maxheap or minheap?

1. [*, 1, 2, 3, 4, *, *, 7, 8]
2. [*, 10, 2, 5, 1, 6, 3, 4, 8]
3. [*, 19, 14, 13, 9, 10, 8, 6, 3, 1, 4]

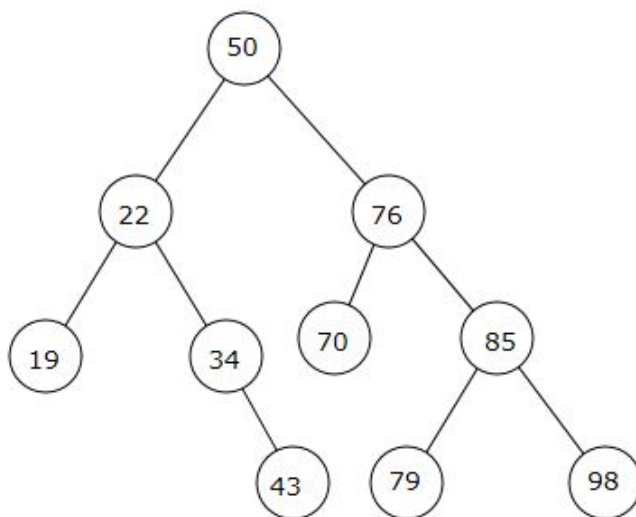
Question 14: Binary Search Tree Creation

Draw the binary search trees that result from inserting the sequences in the specified order. Are your trees unique for this order? What about for these numbers inserted in a different order?

- 77 62 88 19 79 55 59 25 57
- 20 7 9 12 5 14 6 28 44 50 32 26

Question 15: Tree Traversals

What is the pre-order, in-order, and post-order of the binary search tree below?



Question 16: Tree Add and Delete

Draw the binary search tree above from the question above after inserting value 75 and then 32.

Now draw the binary search tree after deleting the value 43, then 85, then 50.

Question 17: B-Trees

Draw the 2-3 tree that results from the following sequence. Show each tree that results from each addition.

4 9 1 3 2 7 8 5 6

Draw the 2-4 tree that results from the following sequence. Show each tree that results from each addition.

4 11 1 3 2 7 10 5 6 8

Question 18: Graphs Definitions

1. What is a path? A simple path? A cycle? A simple cycle?
2. What does it mean for a graph to be connected? complete? acyclic? a tree?
3. Draw a graph that is connected but not complete.
4. Draw a graph that is complete.
5. Draw a graph that is acyclic.

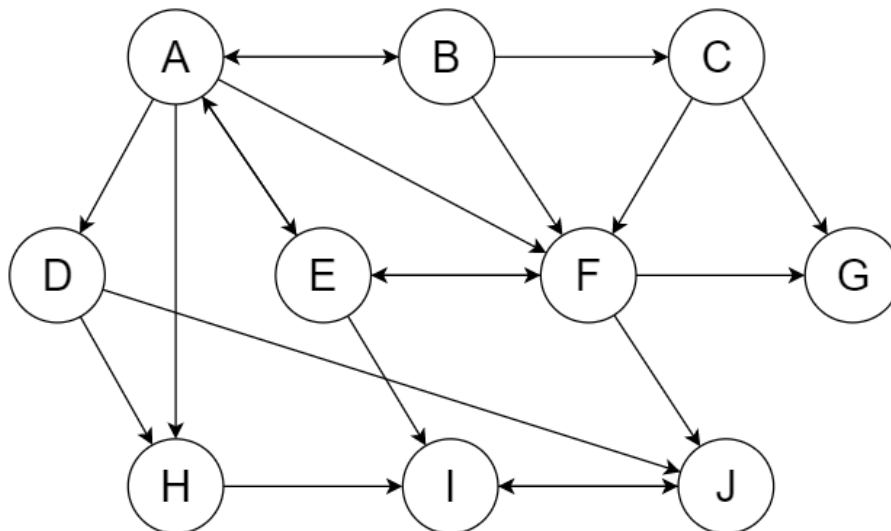
Question 19: Graph Traversal and Ordering

What is a depth-first traversal of the graph below, starting at A? starting at D? Is this the only possible correct traversal?

What is a breadth-first traversal of the graph below, starting at A? starting at D? Is this the only possible correct traversal?

What is a shortest path from A to G? from E to J? Is this the only possible correct path?

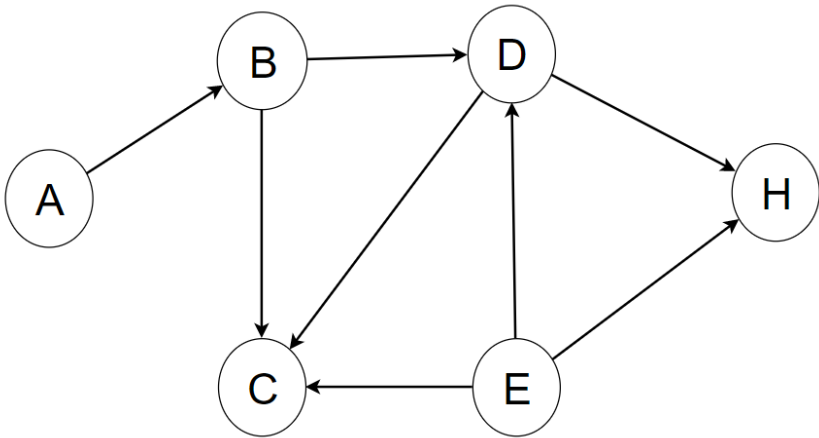
Can you create a topological ordering for this graph? If yes, what is a valid ordering? Is it the only possible correct ordering? If not, why not?



What is a depth-first traversal of the graph below, starting at A? starting at E? Is this the only possible correct traversal?

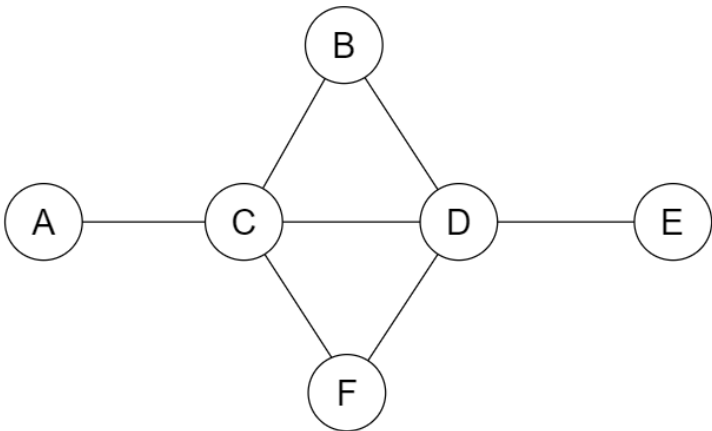
What is a breadth-first traversal of the graph below, starting at A? starting at E? Is this the only possible correct traversal?

Can you create a topological ordering for this graph? If yes, what is a valid ordering? Is it the only possible correct ordering? If not, why not?



Question 20: Graph Representations

What is the adjacency matrix of the graph below? What is the adjacency list? Is the graph weighed? directed? connected? complete?



Draw a graph with the adjacency matrix provided below. Is the graph weighed? directed? connected? complete?

	A	B	C	D	E	F
A			1			
B			4	6	5	
C	3					
D			8		2	

E						
F						

M15 On-Your-Own Practice Questions Selected Answers and Examples

(<https://ccsf.instructure.com/courses/47904/pages/m15-on-your-own-practice-questions-selected-answers-and-examples>)