Assignment 2 - Due: **Wednesday,** October 14 at **6:45 PM** through **eCompanion**

Full Name:_____

Please show your work. Can use extra sheets of paper. **Writing by hand is absolutely fine.**

1- We are performing the following operations on the ADTs below. Show the value returned by the peek and remove operations.
Add 20, 7, 5, 8. Remove (a). Add 10, 14. Peek (a). Add 9. Remove (b). Add 3, 11. Peek (b).
a) FIFO Queue
Remove (a)= _____ Peek (a) = _____ Remove (b) = _____ Peek (b) = _____
b) Stack
Remove (a)= _____ Peek (a) = _____ Remove (b) = _____ Peek (b) = _____

2- We are performing the following operations on a Deque implemented using an array. The initial capacity of the array is 5 and the array doubles whenever the deque reaches its capacity. Please show the **physical state** (draw the array cells) of the array after each of the operations. For the array elements with "junk" values, leave any existing value in the array but cross it out. Show next to the array the values of the **lo** and **hi** indices.
a) pushFront C lo= hi=

b) pushFront A

c) pushBack Z

d) popFront

e) pushBack U

f) pushFront K

g) popBack

h) pushBack J

i) pushFront H

j) pushBack P

k) popFront

3 - Write a **recursive** function called printNums that takes as a parameter an int n and prints all the numbers from 0 to n. Run your code to ensure it does what you intended to do.
e.g.: printNums(5) should print 0 1 2 3 4 5.
- Point out the base case and the recursive step.
- How many stack frames deep is this for a given number n?

4 - Write a **recursive** function called **linear**Search that takes as a parameter an array of ints, array size and the value to look for and returns the index of that value in the array if the value is present, or -1 if not found. Run your code to ensure it does what you intended to do.

e.g.: linearSearch([2, 9, 6, **4**, 7], 5, 4) should return 3
linearSearch([**2**, 9, 6, 4, 7], 5, 2) should return 0
linearSearch([2, 9, 6, 4, 7], 5, 3) should return -1

– Point out the base case and the recursive step.
– How many stack frames deep is this for a given number n?

5- We are using the binary search algorithm to look for a value in the array below. **For each step** of the algorithm write **the value that is looked at** and **the values in the array that are left in consideration for the next step (cross out what is "thrown away")**. No code to write.

3	4	8	11	17	21	25	30	33	39

a - Look for the value 25. How many steps does it take until we find it?
b – Look for the value 5. How many steps does it take to find out that 5 is not present?