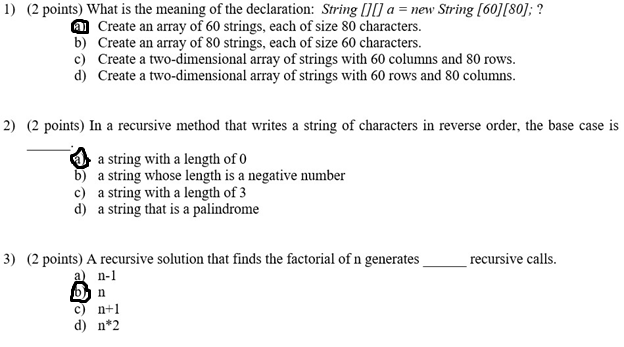
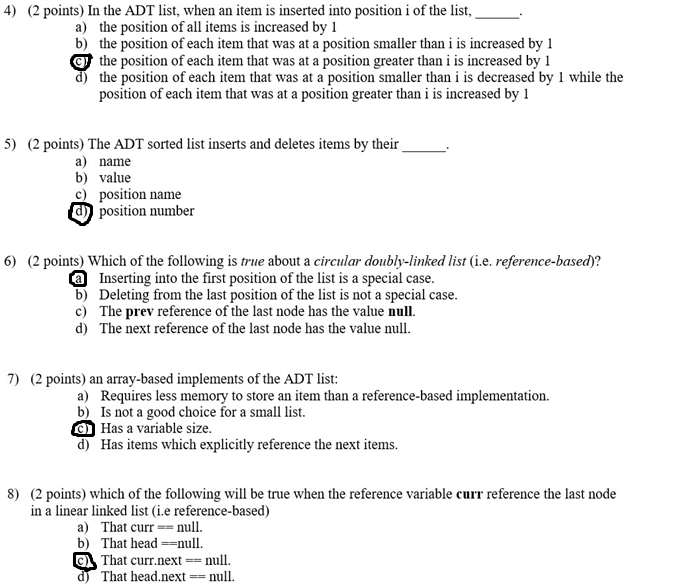
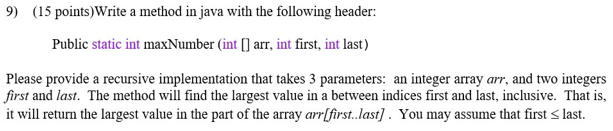
Name Darek Konpka CS 102 – Midterm

There are 13 questions. Submit your answers on blackboard as one document (pdf or word). (80 points)

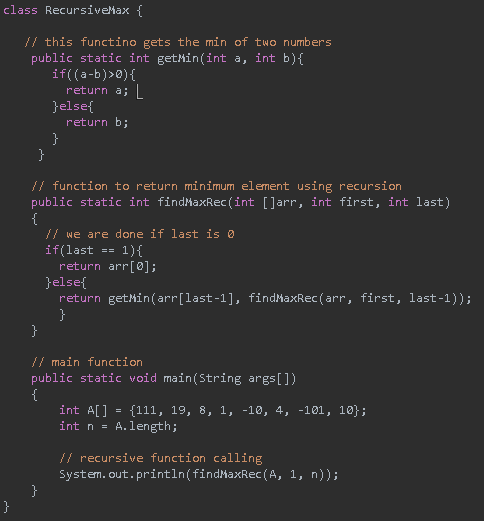
Honor Code Statement for Exam: I, Darek Konopka\_, agree to neither give nor receive any help on this exam from other students. I understand that providing answers to questions on this exam to other students is an academic misconduct violation as is taking or receiving answers to questions on this exam from other students. It is important to me to be a person of integrity and that means that ALL ANSWERS on this exam are my answers. Signed **Darek Konopka**

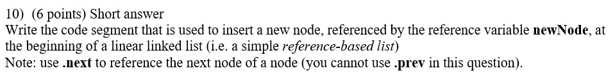






This file will be attached to the ZIP file.

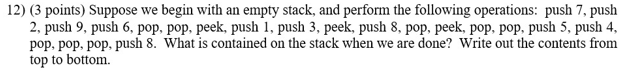




public void addAtStart(int data) {   
 //Create a new node   
 Node newNode = new Node(data);   
   
 //Checks if the list is empty   
 if(head == null) {   
 //If list is empty, both head and tail will point to new node   
 head = newNode;   
 tail = newNode;   
 }   
 else {   
 //Node temp will point to head   
 Node temp = head;   
 //newNode will become new head of the list   
 head = newNode;   
 //Node temp(previous head) will be added after new head   
 head.next = temp;   
 }   
 }   

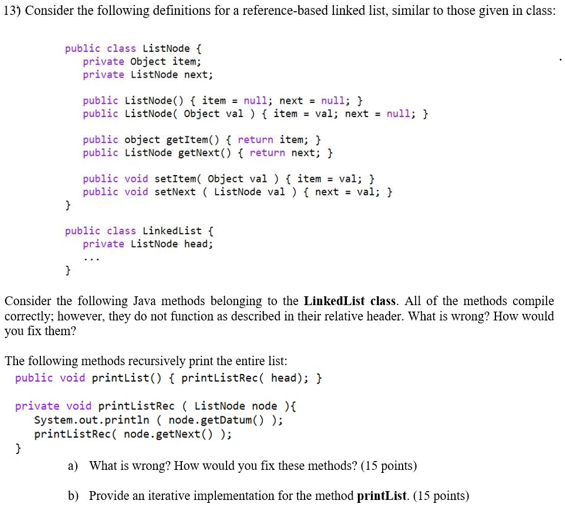

1. The main difference between Stack and Queue is that Stack is LIFO (last item inserted is first) and Queue FIFO (first item inserted is last).
2. For stack you can push elements into the stack or pop them off, for queue you can enqueue or dequeue elements from the queue.
3. **Stack Example:** A stack of plates, you can put plates on top of the stack of plates, or you can remove them.

**Queue Example:** A fast food line, you can be added to the back, but when you take your order, you leave from the front.



8

7



The printListRec method is missing a base-case, so it will iterate forever, or be interrupted by an exception (which is not recommended). To fix this, you will add a base case checking if the node is null.

public void printList() {  
 ListNode node = head;  
 while (node != null) {  
 System.out.println(node.getDatum());  
 node = node.getNext();  
 }  
}