CS 111 Lab 6: Stacks and Queues

20 points – 10 for stack 10 for queue

Goal: To test your understanding of stacks, and queues.

Stack and a queue implementation are provided to you, MyStack and MyQueue, you must use these implementations and not an outside library, do not alter any code provided to you.

Part 1: Stack Methods 10pts.

You will be implementing the following methods using the implementation of the stack provided to you, make sure you understand what each method does before starting. Feel free to use as many auxiliary stacks and queues as needed to complete these methods.

- getCurrentSize() 2pt. gets and returns the size of the stack – this will be more difficult than it used to be, as you have no way of just getting the size, but rather will have to pop everything from the stack, then push it back into the stack

  #hint:

  1) pop and count the number of elements as they area popped; store them in a temp stack
  2) push the elements back onto the original stack
- last() 2pts. Will return the last value in the stack, leaving the stack unchanged
- reverse() 2pts. Will reverse the stack, the last item in the stack will be the top, and the first will be at the bottom

  #hint:

  1) pop items off original stack into first temp stack (they are now in reverse order)
  2) pop items off first temp stack into second temp stack (they are now in original order)
  3) pop items off second temp stack into original stack (they are now in reverse order once again, which is desired)
- deleteAll() 4pts. Will take in a search value and delete all instances of that value while the rest of the stack is left in the same order. It will return the number of the instances of the element that were deleted.

Part 2: Queue Methods 10pts

You will be implementing the same methods using the implementation of the queue this time. make sure you understand what each method does before starting, as they are different then stack methods. Feel free to use as many auxiliary queues and stacks as needed to complete these methods.

- getCurrentSize() 2pt. gets and returns the size of the queue
- last() 2pts. Will return the last value in the queue, leaving the queue unchanged

- reverse() 2pts. Will reverse the queue, the last item in the queue will be the front, and the first will be at the last

  #hint:

  1) poll all items from the queue and pushes them onto a temporary stack (the elements are now in reverse order)
  2) pop the elements off the stack and adds them back into the queue (still in reverse order, which is desired)
- deleteAll() 4pts. Will take in a search value and delete all instances of that value while the rest of the queue is left in the same order. It will return the number of the instances of the element that were deleted

- **Additional requirements: will lose up to 5 pts.**

- Name included at the top of your code
- Commented code
- Clean readable code (if there were no comments can I still understand what each variable and method is for? No magic numbers. Use best coding practices.)