COSC 1437: Programming Lab & Assignment (Spring 2015)

Queue Implementation (in three ways)

+ Stack Implementation (in three ways)

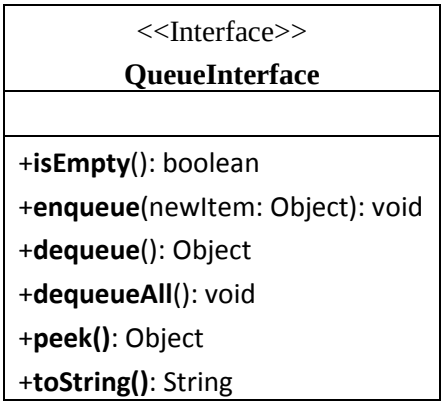
+ isPalindrome (using Stack & Queue)

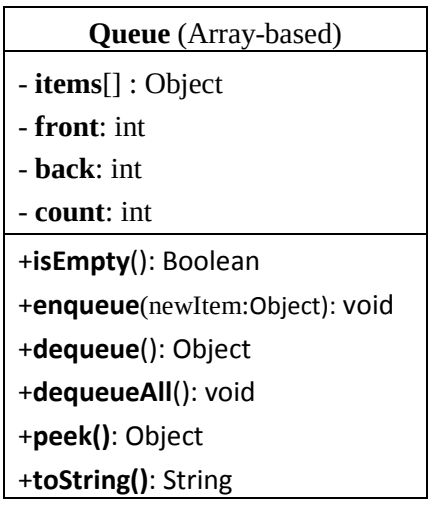
Topics

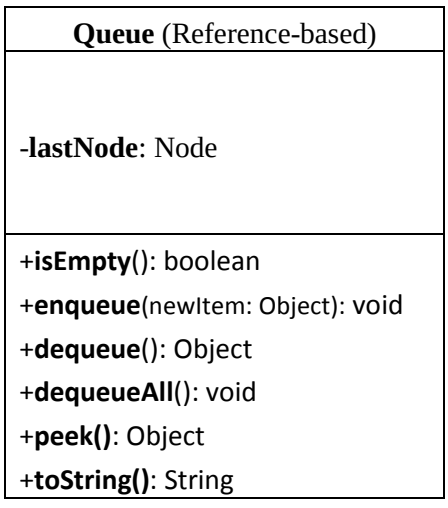
- In Ch. 8, you learned (circular) Queue implementation in three ways, including array-based, reference-based, and list-based.
- In this lab, you are asked to complete the three Queue implementations. Also, you'll implement one simple application, where both Stack and Queue are utilized to check whether the given word is in a symmetric language (i.e., palindrome).

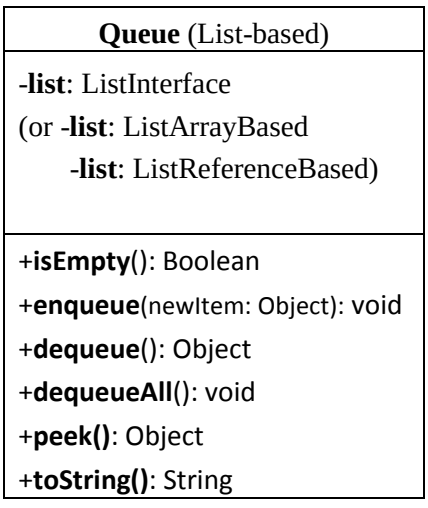
  Notice that Ch.7 will discuss details of Stack implementation in three ways, including array-based, reference-based, and list-based.

Prerequisite Activities









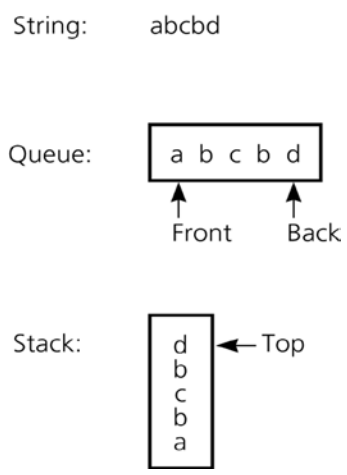
- NOTE: We assume that every index always starts from 0, not 1.
- Three Queue implementations (i.e., "QueueArrayBased.java", "QueueReferenceBased.java", and "QueueListBased.java") and their required files are provided with one simple driver, "TestQueue.java". Notice that the given "TestQueue.java" won't run because of intended errors in the three Queue implementations. Thus, your first mission is to correct errors in the three implementations, referring to "**handout_chap8.ppt**". If all errors are corrected, "TestQueue.java" will generate the following output:

```
5  6  7  8  9
a5  a6  a7  a8  a9
5.0  6.0  7.0  8.0  9.0
Press any key to continue  .  .  .
```

Main Activity

- Using one of your Queue implementation (in Ch. 8) and also one of your Stack implementation (in Ch. 7), implement an application that recognizes Palindromes, say "IsPalindrome.java".
- By definition, palindrome = {***str*** | ***str*** reads the same left to right as right to left}
- Implement "isPal()" in "IsPalindrome.java" class and then call "isPal()" in main() of "IsPalindrome.java".
- Please demonstrate that your implementation is correct, with a couple of your own strings in your main().
- In class, we will discuss how to implement isPal(). Please refer to "**Example: Recognizing Palindromes**" in "**handout_chap8.pdf**". Below is the pseudo code that evaluate whether the given string is palindrome or not.

(Example)




```
boolean isPal(str)
{
        for  (i = 1 to str.length( )) {
                queue.enqueue(i^th character of str);
                stack.push(i^th character of str);
        }
        // start to compare
        while  (!queue.isEmpty( )) {
                if  (queue.dequeue( ) != stack.pop( ))
                        return false;
        }
        // finished w/ empty queue (and empty stack)
        return true;
}
```

What to Hand in

- Turn in your program (i.e., "**IsPalindrome.java**") via Blackboard.