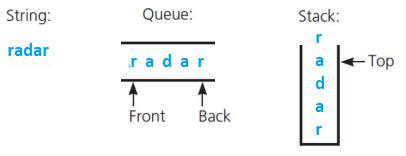
**CSc 2720 - Data Structures: Lab 4**



**Deadline to Submit: [February 5] [11:00pm] ET(US)**

Failure to submit will result in a zero for this lab.

**Problem 1: [100 points]**

A palindrome is a string of characters that reads the same from left to right as it does from right to left. Write a function called checkPalindrome that takes a string s1 and evaluate if the string is a palindrome or not using **only a stack and a queue** data structures. You should make use of the built-in java implementation of the stack and queue data structure under java.util.Stack and java.util.Queue **.**

**Example 1:**



**Input:** s1 = "radar"



**Output:** true

**Example 2:**



**Input:** s1 = "HitTheRoadJack"



**Output:** false

----------------------------------------Function Template------------------------------------------

**import** java.util.Stack;

**import** java.util.LinkedList;

**import** java.util.Queue;

**public class** Lab4 {

**public static void** main (String[]args){

String s1 = "radar";

**boolean** ans= checkPalindrome(s1);

System.***out***.println(ans); // Should be True

}

**public static boolean** checkPalindrome(Strings1){Stack<Character> s = **new** Stack<Character>(); Queue<Character> q = **new** LinkedList<Character>();

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| // Example of stack | | | push | s.push("r") |
| // Example of | | stack | pop | s.pop() |
| // Example of | | queue enqeue | | q.add("r") |
| // | Example of queue dequeue | | | q.poll() |
| // | Check if stack or queue is empty | | | s.isEmpty() , q.isEmpty() |

* INSERT YOUR CODE HERE

}

}