Algorithms

CMPT 435

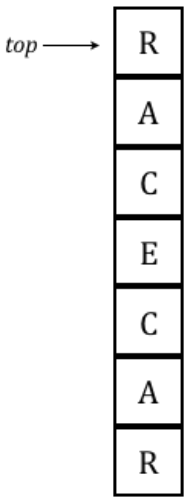
Assignment 1 - 100 points

Goals

- to program a few elementary data structures so we can experiment with them later on.

Requirements and Notes

- Develop and test a singly linked list (essentially, a Node class). [20 points]
- Using the elements of your linked list, develop a stack. You must implement it yourself; you **may not** use any built-in features of the language or its libraries for the stack. (You **may** use language libraries for other things, like file operations and the like.) [20 points]



Stack

- Using the elements of your linked list, develop a queue. You must implement it yourself; you may not use any built-in features of the language or its libraries for the queue. [20 points]
- Download the the text file `magicitems.txt` from our web site. [30 points]
- Read it line-by-line into **array**. (You will need this code again.)
- Check each line of the array to see if it's a palindrome, ignoring spaces and capitalization. Print it out **only** if it is a palindrome.
  - To check whether or not a given string is a palindrome, take it character by character and push each character on a stack and enqueue each character on a queue. When every character is on the stack and in the queue, pop the stack and dequeue the queue one character at a time. If they always match then the string is a palindrome.
- Create a LaTeX document that includes code listings (with line numbers) for your node, stack, queue, and main program. **Explain the good parts of how each works**, as if you were teaching it to someone else. Reference line numbers for pedantic clarity. [10 points]

Your code must …

- **not** be in a Java package; it makes it harder for me to compile. [−∞ if not]
- separate structure from presentation.
- be professionally formatted yet uniquely yours (show some personality)
- demonstrate best practices and make me proud to be your teacher.

Resources

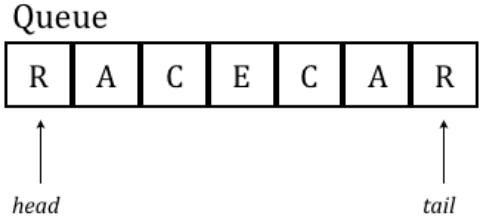
- Linked lists are described in our text in chapter 10.2, starting on page EC.
- Stacks and queues are described in our text in the beginning of chapter 10, starting on page 1110 1000.

Hints

Make sure that I have approved of your programming language (the one about which you wrote a limerick in Assignment 0) before you begin.

Submitting Your Work

Make **many** commits to GitHub. I do not want to see one massive "everything" commit when I review your code. (It's −∞ if you do that.) Commit early and often. And make sure your commit messages are descriptive and informative, and — if possible — entertaining.



Be sure that you make your final commit for this assignment on or before the due date. (See our syllabus for those details.)

© 2019-2112 Alan G. Labouseur, All Rights Reserved