

Assignment 7: Data Structures

COP3330 Fall 2017

Due: Friday, December 8, 2017 at 11:59 PM

ABSOLUTE DEADLINE, no late submissions will be accepted.

Objective

This assignment is broken into two exercises and will provide experience in working with linked lists and stack classes.

Before You Start

You will need a copy of `listnode.h`, `list.h`, and `stack.h` to complete this assignment. They are linked on blackboard as well as on my site.

Exercise 1

Filename: `palindrome.cpp`

Write a program that uses a stack object to determine if a string is a palindrome (i.e. the string is spelled identically backward and forward).

Go ahead and start your program by reading in a C-style string from standard input using the `getline` function. You may assume a limit of 100 characters on the string. Your algorithm must make use of a stack of type `char`. Use the Deitel implementation of `Stack` from `stack.h` (don't change this file).

Ignore space, punctuation, special characters, and digits. That is, only count letters as part of a palindrome. Account for upper/lower case (e.g. 'B' and 'b' are matching letters).

Since you are reading data into a C-style string to begin, you may use any of the libraries `<iostream>`, `<cstring>`, and `<cctype>`.

Sample runs: (user input underlined)

Please enter a string:

> ABCDEFGHIHGFEDCBA

"ABCDEFGHIHGFEDCBA" IS a palindrome

Please enter a string:

> The quick brown fox

"The quick brown fox" is NOT a palindrome

Please enter a string:

> Cigar? Toss it in a can. It is so tragic.

“Cigar? Toss it in a can. It is so tragic.” IS a palindrome

Exercise 2

Modify the `List` class (file `list.h`) so that it has two more functions which will allow inserts and removes from anywhere in the linked list. Your functions should be called:

- `insertMiddle`
- `removeMiddle`

Your functions should have all the same features as the given `insert` and `remove` functions, except that yours each have one additional parameter. The second parameter on each of your functions should be of type `int`, representing the position at which to insert/delete. Sample calls for a list of integers:

```
list.insertMiddle(345, 5); //attempts to insert the value 345 as the 5th item in the list
list.removeMiddle(x, 10); //attempts to delete 10th item in the list, storing its value in x
```

For `insertMiddle`, if the position number is larger than the number of items in the list, just insert the item at the back. If it's too small, insert at the front.

For `removeMiddle`, return `false` if the position is invalid and don't remove anything.

You can use the provided `menu7.cpp` (modified version of the menu program found in Figure 21.5) for testing these functions.

Submitting

Archive your `palindrome.cpp`, `list.h`, and `README` files into a simple tar ball (no compression). Submit to the assignment 7 link on blackboard. Make sure the submitted files are named as specified by the syllabus and this writeup, and that you do not include any extra files.

General Advice

- Make sure to double check your blackboard submission to make sure everything works when downloaded.
- Email a copy of your finished homework files to your own FSU account. This email will have a time stamp that shows when they were sent and will also serve as a backup. Useful in case something happens to blackboard.
- Periodically (e.g. nightly) make a backup of your assignment to another machine (e.g. personal computer, linprog, email). Computers die and accidents happen, having a backup prevents you from having to start from scratch. This is also a good feature to have in your makefile as you can abstract the details behind a single call.
- Make sure to include the `README` file as specified in the assignment syllabus http://ww2.cs.fsu.edu/~dennis/teaching/2017_fall_cop3330/docs/syllabus.pdf

COP 3330 Fall 2017 Assignment 7**Due: 2017-12-08 11:59 PM****Student Name:****Grader:****Grade:****Date Submitted:****Date Graded:**

Description	Earned	Possible	Comments
1. Palindrome Correctness		20	
2. Palindrome I/O		10	
3. Palindrome String		10	
4. List Insert		15	
5. List Remove		15	
6. List Management		5	
7. README		10	
8. Submission		10	

Notes: