CS 280
Fall 2021
Recitation Assignment 7
November 10, 2021

Due Date: Sunday, April 14, 2020, 23:59
Total Points: 6

Consider the language

L = {*SS'* , where *S* is a string of characters, and *S'* = *reverse(S)*}

For example,

- The strings AA, ABCCBA, and are in L.
- The strings ABAB, ABC, ABCCB, ABCCBB are not in L
- Empty string is not in the language.
- Note: A string with an odd number of characters is not in the language

Write a C++ function to recognize strings in this language, called *inLanguage*, that uses both a queue and a stack. The function takes a string as an argument and returns true if the input string is in the language, and false otherwise. Thus, as you traverse the input string, insert each character of *S* into a queue and each character of *S'* into a stack.

If the input string is empty, the `inLanguage` function should print out the message "Empty string is not in the Language.", then return false. While if the input string is not an odd number of characters, it prints "A string with an odd number of characters is not in the Language.", then return false.

The signature of the function is defined as:

```
bool inLanguage(string & inputString);
```

Note: Use the queue and stack classes from the Standard Template Library (STL) to create a queue and stack containers.

Vocareum Automatic Grading

- A driver program is provided for testing the implementation, called "RA7prog.cpp", on Vocareum. The "RA7prog.cpp" will be propagated to your Work directory. The driver program opens a provided file name from the command line for reading. For each sentence read from the file, it calls the function `inLanguage` to check whether that sentence is in L or not. For example, if the sentence is "AA", it prints the message "AA" is a sentence in the Language. Otherwise, if the sentence is "ABCBCA ", it prints the message "ABCBC" is not a sentence in the Language.
- You are provided by a set of 5 testing files associated with Recitation Assignment 7. Vocareum automatic grading will be based on these testing files. You may use them to

check and test your implementation. These are available in a compressed archive "RA7 Test Cases.zip" on Canvas assignment. The testing case of each file is defined in the Grading table below.

- The testing is established based on the correct results returned by the call to your `inLanguage()` function and an error message generated by your implementation for some cases. Examples of testing cases are shown in the slides.
- "RA7prog.cpp" is available with the other assignment material on Canvas.

Submission Guidelines

- Please upload your implementation to Vocareum as a "inLanguage.cpp" file or any other file name you may choose. The file should include the implementation of the function *inLanguage*.

- **Submissions after the due date are accepted with a fixed penalty of 25%. No submission is accepted after Tuesday 11:59 pm, November 16, 2021.**

Grading Table

Item	Points
Compiles Successfully	1
emptystr: checks for an empty string	1
emptyfile: All whitespace	1
valid1: valid strings	1
oddlength: a string with odd number of characters	1
Invalid1: Invalid string	1
Total	6