Palindromes

Points

Points	
85	Running Program
15	Code Review
100	**TOTAL**

Objectives

- Use Recursion
- Use command line arguments
- Process multiple command line arguments
- Parse information from a string letter by letter

Submission

1. **Source Code**: Submit the source code (**main.cpp , functions.h , and functions.cpp** files) to Mimir

Program

You will write a program that uses a **recursive function** to determine whether a string is a character-unit palindrome. Moreover, flags can be used to indicate whether to do case sensitive comparisons and/or not ignoring spaces. For example "A nut for a jar of tuna" is a palindrome if spaces are ignored and not otherwise. "Step on no pets" is a palindrome whether spaces are ignored or not, but is not a palindrome if it is case sensitive since the ‘S’ and ‘s’ are not the same.

Background

Palindromes are character sequences that read the same forward or backward (e.g. the strings "mom" or "123 454 321"). Punctuation and spaces are frequently ignored so that phrases like "Dog, as a devil deified, lived as a god." are palindromes. Conversely, even if spaces are not ignored phrases like "Rats live on no evil star" are still palindromes.

Specifications

Command Line Parameters

The program name will be followed by a list of strings. The program will determine whether each string is a palindrome and output the results. Punctuation will always be ignored. An optional flag can precede a term that modifies how the palindrome is determined.

Strings

Each string will be separated by a space on the command line. *If you want to include a string that has spaces in it (e.g. "Rats live on no evil star"), then put quotes around it. The quote characters will not be part of the string that is read in.*

Flags

- Optional for the user (but your program has to handle them!)
- If present, flags always appear after the program name and before any strings are processed and apply to all subsequent strings processed.
- Must start with a minus (-) sign followed by one or two flag values that can be capital or lowercase. e.g. **`-c`**, **`-S`**, **`-Cs`**, **`-Sc`**, *etc*.
    - There are no spaces between starting minus (-) sign and flag(s).
    - Flags values are case-insensitive.
    - **`c`** or **`C`**: Indicates that comparisons should be case-sensitive for all input strings. *The default condition (i.e. if the flag is NOT included) is to ignore case-sensitivity.* So, for example:
        - `palindrome Mom` should evaluate as being a palindrome.
        - `palindrome -c Mom` should **not** evaluate as being a palindrome.
    - **`s`** or **`S`**: Indicates that comparisons should not ignore spaces for all input strings. *The default condition (i.e. if the flag is NOT included) is to ignore spaces.* So, for example:
        - `palindrome "A nut for a jar of tuna"` should evaluate as being a palindrome.
        - `palindrome -s "A nut for a jar of tuna"` should **not** evaluate as being a palindrome.
- Options can appear in different flags, e.g. you can use **`-Sc`** or **`-S -c`**
- Output printUsageInfo when you get an invalid flag, i.e. any value other s or c after the first dash (-)

Code Expectations

- Your program should only get user input from the command line. (i.e. "cin" should not be anywhere in your code).
- Required Functions:
    - A function that prints program usage message in case no input strings were found at the command line.
        - Name: printUsageInfo
        - Parameter(s): a string representing the name of the executable from the command line.
        - Return: void.
    - A function that determines whether a string is a character-unit palindrome.
        - Name: isPalindrome
        - Parameter(s): an input string, a boolean flag that considers case-sensitivity when true, and a boolean flag that ignores spaces when true.
        - Return: bool.
        - Calls helper function isPalindromeR to determine whether a string is a palindrome.
            - The string passed into isPalindromeR after dealing with flags.
                - If case in-sensitive, make all lower or upper case so that case does not matter.
                - If spaces are ignored, remove spaces from string.
    - Helper **recursive** function that determines whether a string is a character-unit palindrome. This does not deal with flags. So it will be case sensitive and include spaces as characters.

        - Name: isPalindromeR
        - Parameter(s): an input string
        - Return: bool
- All functions should be placed in a separate file following the code organization conventions we covered.

Program Flow

1. Your program will take arguments from the **command-line**.
2. Determine if you have enough arguments. If not, output a usage message and exit program normally (return 0 or exit with code 0).
3. If flags are present.
    1. Process and set values to use when processing a palindrome.
4. Loop through remaining arguments which are all input strings:
    1. Process each by calling isPalindrome function with flag values.
    2. Output results

Program Flow Notes:

- Any time you encounter a syntax error, you should print a usage message and exit the program immediately.

Hints

- Remember rules for a good recursive function.
- Recommended Functions to write:

  Note: You could combine these into a single function. e.g. "preprocessString"
    - tolower - convert each letter to the lowercase version for case-insensitive comparisons.
        - Parameter(s): a string to be converted to lowercase
        - Return: a string with all lowercase characters

    - removePunctuation - Remove all punctuation marks possibly including spaces depending on the flag value.
        - Parameter(s): a string and a boolean flag indicating whether to also remove spaces
        - Return: a string with punctuation/spaces removed
- Existing functions that might help:
    - tolower
    - substr
    - isalnum
    - erase (This can be used instead of substr, but is a bit more complicated.)
- Development Environment:
    - Since you will be using command line parameters, you will need to figure out how to add command line parameters in your IDE. In Eclipse, when you get to Run As, select “Run Configurations…” Instead of “1 Local C/C++ Application” In the resulting window, click on the “Arguments” tab. You can run the command line arguments there.

Example Output

Note: you will probably only see the `./` if running in a Unix environment.

```
./palindrome
Usage: ./palindrome [-c] [-s] string ...
  -c: turn on case sensitivity
  -s: turn off ignoring spaces

./palindrome -c
Usage: ./palindrome [-c] [-s] string ...
  -c: turn on case sensitivity
```

```
  -s: turn off ignoring spaces

./palindrome Kayak
"Kayak" is a palindrome

./palindrome Ka:yak?
"Ka:yak?" is a palindrome

./palindrome -c Kayak
"Kayak" is not a palindrome.

./palindrome -C Kayak
"Kayak" is not a palindrome.

./palindrome -c kayak
"kayak" is a palindrome.

./palindrome "Test Set"
"Test Set" is a palindrome.

./palindrome "T@est Se*t"
"T@est Se*t" is a palindrome.

./palindrome -sc "Test Set"
"Test Set" is not a palindrome.

./palindrome -s -c "Test Set"
```

```
"Test Set" is not a palindrome.

./palindrome -s -s "Test Set"
"Test Set" is not a palindrome.

./palindrome -scs "Test Set"
"Test Set" is not a palindrome.

./palindrome -cd
Usage: ./palindrome [-c] [-s] string ...
  -c: turn on case sensitivity
  -s: turn off ignoring spaces

./palindrome Kayak madam "Test Set" "Evil Olive" "test set"
"loop pool"
"Kayak" is a palindrome.
"madam" is a palindrome.
"Test Set" is a palindrome.
"Evil Olive" is a palindrome.
"test set" is a palindrome.
"loop pool" is a palindrome.
```

```
./palindrome -s Kayak madam "Test Set" "Evil Olive" "test set"
"loop pool"
"Kayak" is a palindrome.
"madam" is a palindrome.
"Test Set" is not a palindrome.
"Evil Olive" is not a palindrome.
"test set" is not a palindrome.
"loop pool" is a palindrome.

./palindrome -c Kayak madam "Test Set" "Evil Olive" "test set"
"loop pool"
"Kayak" is not a palindrome.
"madam" is a palindrome.
"Test Set" is not a palindrome.
"Evil Olive" is not a palindrome.
"test set" is a palindrome.
"loop pool" is a palindrome.

./palindrome -c -s Kayak madam "Test Set" "Evil Olive" "test
set" "loop pool"
"Kayak" is not a palindrome.
"madam" is a palindrome.
"Test Set" is not a palindrome.
"Evil Olive" is not a palindrome.
"test set" is not a palindrome.
"loop pool" is a palindrome.
```