Assignment

A palindrome is a string of letters that reads the same forward and backward (ignoring spaces, punctuation, and capitalization). For example:

A man, a plan, a canal: Panama!

Write two (2) C++ programs to determine if a string represents a palindrome. The first program is based on string objects while the second is based on c-strings.

Program #1: string Object version (named palindrome.cpp)

1. Prompt for the user to enter a potential palindrome
2. Read the user input in to a string object (the test input to your program may contain spaces but will not contain punctuation or mixed-case letters)
3. Write a message stating that the input was or was not a valid palindrome
4. Your program may assume that all characters are in one case and that the input does not contain any punctuation (aside from spaces)

Suggestions

It is probably easiest to first remove all of the spaces from the string and to then test the spaceless string to see if it is a palindrome.

The following code fragment (which you may use in your program) will remove the spaces from a string object:

```cpp
int x = s.find(' '); // see p. 307
while(x != -1 && x < (signed)s.size())
{
    s.erase(x, 1);
    x = s.find(' ');
}
```

See palnumber (written in class) for an example of doing a character-by-character comparison of string objects.

Although much of the code for this assignment is provided for you, you should understand what the code is doing (who knows what could be on the next mid-term :-)

Program #2: C-String Version (named cpalindrome.cpp)

1. The program accepts a potential palindrome on the command line (i.e., an array of cstrings) – This program does not prompt for input or read a string from cin and this program must be run from the command line. For example:
   c:\> cpalindrome a man a plan a canal panama
2. You can also provide "command line" input from within Visual Studio:
   a. On the Solution Explorer, click "Properties" (the button on the top left)
   b. This opens a new window. Select "Debugging" on the left panel
   c. Select "Command Line Arguements" on the right panel and enter the arguments (your test palindrome) and press OK
3. Write a message stating that the input was or was not a valid palindrome
4. Your program may assume that all characters are in one case and that the input does not contain any punctuation (aside from spaces)

Suggestions

Each word entered on the command line is a space-separated string of characters, which the system provides to your program as a cstring located in the argv array. (Note that the following is not code that you write but is an illustration that corresponds to the example shown in step 1 above.) In this example, argc = 8
   argv[0] = cpalindrome (the program name – often not of interest)
   argv[1] = a
   argv[2] = man
   argv[3] = a
   argv[4] = plan
   argv[5] = a
   argv[6] = canal
   argv[7] = panama

The test input will not contain punctuation marks or mixed case letters

Concatenate the command line arguments into a single cstring, which can then be checked as above
   char palindrome[1000] = ""; // empty string
   for (int i = 1; i < argc; i++)
       strcat(palindrome, argv[i]);

Submitting Your Code

Upload two files to Blackboard/WSU Online: palindrome.cpp and cpalindrome.cpp (please do not zip the files). Please be sure the files are named as specified. Please be sure that cpalindrome takes its input from the command line and does not prompt for input or attempt to read from cin.