Sean Conrad

CSCI 301 Section 1

Project 1 Due 1/19/17

Design Document: Palindrome Program

Problem:

Design a program that will take a string from the user and determine if that string is a palindrome. The program must ignore spaces, special characters, numbers, and only interpret the alpha characters. The program must also support up to 80 alpha characters and use arrays to handle all strings.

I decided to use a class to solve this problem.

CLASS: Palindrome.

Private:

-char Array[80]; this will hold the characters of the string

-bool Result; this will hold the final result

-int Length; this will hold the length of the array

Public:

-setPalindrome(string); this will process the string and insert it into the Array[] and set the Length

-setResult(); this will create a second array that contains all the elements of Array[] in reverse order. It will then compare each element from both arrays and set bool Result to true/false accordingly.

-getResult(); this will return the value of bool Result. Use this function to determine what statement will be printed.

Main();

Main function will create an object of the class defined above. Only one variable in main() is required.

Variable: String Str;

Print a statement for User that asks for a possible palindrome and assigns the input to variable Str;

Call the function setPalindrome() with the parameter Str.

Then call setResult() without any parameters.

Finally, use getResult(); in an if-statement to determine what print statement should be used and then return 0;

//Sean Conrad

//CSCI 301 Section 1

//Project 1

//Due 1/19/2017

//This programm will determine if a user entered string is an array.

//it supports space and non alpha character removal

//the program will only support up to 80 characters

#include <iostream>

#include <string>

#include <cctype>

#include <algorithm>

#define SIZE 80

using namespace std;

//class declaration

class Palindrome

{

private:

char array[SIZE];

bool result;

int length;

public:

void setPalindrome(string);//Done.

bool setResult(); //done.

bool getResult() const; //done.

};

//the setPalindrome function will check each character, remove non alpha and spaces, and insert characters into an array.

void Palindrome::setPalindrome(string str)

{

length = 0;

for(char c : str)

{

if(isalpha(c))

{

array[length] = c;

++length;

}

}

}

//the setResult function will create reverse[] and set it to the exact reverse of array[].

//then it will compare each array element by element, stopping only if it doesnt match or it has finished comparing

//then it will set the bool variable accordlingly

bool Palindrome::setResult()

{

char reverse[SIZE];

int i = 0;

bool isTrue = true;

for(int i = 0; i<length;++i)

{

reverse[length-1-i]=array[i];

}

i=0;

for(i = 0; i<length;++i)

{

if(reverse[i] == array[i] && i != length)

{

result = true;

}

else result = false;

}

}

//will return a bool value.

bool Palindrome::getResult() const

{

return result;

}

//main creates an object and call the appropriate functions from the Palindrome Class.

int main()

{

Palindrome String;

string str;

cout<<"Enter possible palindrome using only lower case letters.”<<endl;

getline(cin,str);

String.setPalindrome(str);

String.setResult();

if( String.getResult() )

{

cout<<”It is a Palindrome.”;

}

else

{

cout<<”It’s NOT a Palindrome.”;

}

return 0;

}

Test Cases:

1.

Enter palindrome.

If I had a hi-fi

It is a Palindrome.

2.

Enter palindrome.

Madam I’m adam

It is a Palindrome.

3.

Enter palindrome.

My name is Sean and I like science

It’s NOT a Palindrome.

User Document

This program will take a phrase you type and tell you if it is a palindrome. Follow the steps below to use the program.

1. Compile and run the program.
2. The program should read “Enter possible palindrome using only lower case letters.”
3. Type the phrase you want to check if it is a palindrome and press ‘ENTER’
4. The program will print the final result.

Summary:

I had a difficult time making this program at first. At first glance it seems straight forward but I eventually found it quite challenging. My biggest problem was separating alpha characters from spaces and non alpha characters.

Eventually I found a good way to use to separate sort the characters. In this program I decided to make a class. I knew a class wasn’t necessary for this particular program but I knew the importance of being able to create object oriented programs so that is why I did it this way.