Alexander Laudino

CSC-162-IN1

Dr. Farrett

Lab Assignment 3 – Palindrome Detector Reference Documents

**Requirements**

Requirements for PalindromeDetector

|  |  |
| --- | --- |
| Requirement | Priority |
| Two variables to hold String and char array | High |
| Constructor to set variables | High |
| Method for user input | High |
| Method to remove whitespace and symbol characters from String | High |
| Method for returning String variable | Medium |
| Empty method to be called from program that returns boolean value | High |
| Method to check whether array of characters is a palindrome | High |
| Method to recursively check subarray | High |

Requirements for TestPalindromeDetector

|  |  |
| --- | --- |
| Requirement | Priority |
| Create new PalindromeDetector object | High |
| Initialize and set boolean variable to hold return value of isPalindrome() method | High |
| Display result | High |

**Pseudo-code**

This program checks if a user input word, phrase, or sentence is a palindrome, meaning it reads the same forward and backward.

TestPalindromeDetector

Begin

1. Create new PalindromeDetector object as test
2. boolean c = test.isPalindrome()
3. Print “Is “ + test.getInput() + “ a palindrome: “ + c

End

The variables, constructors, and methods the program depends on.

PalindromeDetector

String s, user input word, phrase or sentence

char[] c, array of only letters and digits from String

Begin PalindromeDetector()

1. Set s to setInput()
2. Set c to normalizeInput(s)

End

Begin setInput()

1. Print “Enter a word or phrase: “
2. Prompt user for input
3. Return input

End

Begin normalizeInput(String s)

1. Create temporary array with size of length of string input
2. Initialize character counter, charCount = 0
3. Begin for loop to loop through characters in string
4. If character is letter or digit
5. Add lowercase of character to temporary array
6. Increment character counter
7. Create new character array with size of character counter
8. Begin for loop to copy non-empty elements from temporary array to character array
9. Return character array

End

Begin getInput()

1. Return s

End

Begin isPalindrome()

1. Return isPalindrome(c)

End

Begin isPalindrome(char[] c)

1. Return isPalindrome(c, 0, c.length – 1)

End

Begin inPalindrome(char[] c, int low, int high)

1. If high <= low
2. Return true
3. Else if c[low] != c[high]
4. Return false
5. Else
6. Return isPalindrome(c, low + 1, high – 1)

End

|  |
| --- |
| PalindromeDetector |
| -s : String  -c : char[] |
| +PalindromeDetector()  -setInput() : String  -normalizeInput(s : String) : char[]  +getInput() : String  +isPalindrome() : boolean  -isPalindrome(c : char[]) : boolean  -isPalindrome(c : char[], low : int, high : int) : boolean |

**UML**

|  |
| --- |
| TestPalindromeDetector |
|  |
|  |