Tyler Waltze

7/20/15

MET CS 342

Programming Assignment 2

Explanation

The palindrome algorithm is rather simple and straightforward. First thing required is to sanitize the string. Strip out any unimportant characters/attributes (punctuation, case, etc). Then, create a stack of chars. Step through the string, character by character, and add each character to the top of the stack. At the end, the stack holds the string in reverse order. The last letter of the original string is now the first to come off the stack. Next, step through the stack and pop off each char. Each time a char is popped, compare it to the char at the same index in the original string. If they match, continue on. If they don't match, that means the word is not a palindrome and we can stop right there. If we reach the end without any problems, then the word is a palindrome.

The N Queen problem is based on recursion and an array implementation of a stack holding *n* number of queens. The index of each spot in the array corresponds to a row on a chess board, while the actual integer value at said index corresponds to a column on a chess board. Starting at [0, 0], try placing a queen. After every successful queen placement, try placing a new queen in the next column. While in each column, move down the rows, from 0 to $n - 1$, trying to place a queen. After every queen placement, validate the board. This is done by stepping through the array and checking the rows, columns, and diagonals for any conflicts against the most recently added queen (that at the top of the stack). In order to check the diagonals, we use the line equation with a slope of 1. If a board is invalid, we pop the top queen off the stack and try a new row. If we reach the end of the row with no valid board configuration, we return null. The parent recursive call then gets a null return value and does the same thing as described above.

Output

Palindrome

```
Enter the string you want to check: racecar
RACECAR is a palindrome.

Enter the string you want to check: raC e car
RACECAR is a palindrome.

Enter the string you want to check: apple
APPLE is not a palindrome.
```

N Queens

```
[0, 4, 7, 5, 2, 6, 1, 3]
| 0 |   |   |   |   |   |   |   |
|   |   |   |   | 4 |   |   |   |
|   |   |   |   |   |   |   | 7 |
|   |   |   |   |   | 5 |   |   |
|   |   | 2 |   |   |   |   |   |
|   |   |   |   |   |   | 6 |   |
|   | 1 |   |   |   |   |   |   |
|   |   |   | 3 |   |   |   |   |

[0, 4, 7, 5, 2, 6, 1, 3]
| 0 |   |   |   |   |   |   |   |
|   |   |   |   | 4 |   |   |   |
|   |   |   |   |   |   |   | 7 |
|   |   |   |   | 5 |   |   |   |
|   |   | 2 |   |   |   |   |   |
|   |   |   |   |   |   | 6 |   |
|   | 1 |   |   |   |   |   |   |
|   |   |   | 3 |   |   |   |   |

[0, 4, 7, 5, 2, 6, 1, 3]
| 0 |   |   |   |   |   |   |   |
|   |   |   |   | 4 |   |   |   |
|   |   |   |   |   |   |   | 7 |
|   |   |   |   |   | 5 |   |   |
|   |   | 2 |   |   |   |   |   |
|   |   |   |   |   |   | 6 |   |
|   | 1 |   |   |   |   |   |   |
|   |   |   | 3 |   |   |   |   |
```