```
/* ************************************************** *
 * Assignment : 1 - Recursion
 * Name       : Lina Kang
 * Student ID : 1072568
 * CS1D       : MW 2:30 - 5:00
 * Due Date   : 08/26/20
 * ************************************************
 *                    DESCRIPTION
 *
 * This assignment checks for palindrome words using a
 * recursive function that takes a string as an argument
 * and returns a TRUE if the string is a palindrome
 * otherwise FALSE is returned.
 *
 * ************************************************
 *                      OUTPUT
 *
 * ** Palindrome Check **
 *
 * ----------------------------------------
 * Word : Saddleback
 * False - "saddleback" is NOT a Palindrome
 * ----------------------------------------
 * Word : A man a plan a canal Panama
 * True - "amanaplanacanalpanama" is a Palindrome
 * ----------------------------------------
 * Word : The rain in Spain
 * False - "therainінspain" is NOT a Palindrome
 * ----------------------------------------
 * Word : No lemon, no melon
 * True - "nolemonnomelon" is a Palindrome
 * ----------------------------------------
 * Word : radar
 * True - "radar" is a Palindrome
 * ----------------------------------------
 * Word : CS1D
 * False - "cs1d" is NOT a Palindrome
 * ----------------------------------------
 * Word : Was it cat I saw?
 * True - "wasitcatisaw" is a Palindrome
 * ----------------------------------------
 * Word : Racecar
 * True - "racecar" is a Palindrome
 * ----------------------------------------
 * Word : dad
 * True - "dad" is a Palindrome
 *
 * ************************************************/

#include <iostream>

using namespace std;

// recursion function
// - checks the match between first/last characters
// - if they match, delete first/last
// - continue checking the rest of the pairs
```

```cpp
bool recursion(string str)
{
    //if the string is only left with the middle character or no character(Base Case)
    if(str.size() == 1 || str.size() == 0)
        return true;
    //checks if first letter and last letter is the same (General Case)
    else if(str[0] == str[str.size()-1])
    {
        string newStr;
        for(int i = 1; i < str.size()-1; i++)
        {
            newStr.append(to_string(str[i]));
        }
        recursion(newStr);
    }
    else
        return false;

}

// removes space, changes to lowercase, removes punctuation from input words (helper function)
void clearSpaceLowercasePunctuation(string & str)
{
    for(int i = 0; i < str.size(); i++)
    {
        if(str[i] != ' ' && !ispunct(str[i]))
            str[i] = tolower(str[i]);
        else
        {
            str.erase(i,1);
            i--;
        }
    }
}

int main()
{
    string words[9]= {"Saddleback",
                      "A man a plan a canal Panama",
                      "The rain in Spain",
                      "No lemon, no melon",
                      "radar",
                      "CS1D",
                      "Was it cat I saw?",
                      "Racecar",
                      "dad"};

    string inputString;

    cout << "** Palindrome Check **" << endl << endl;

    //traverse through words array and checks if the word is a palindrome
    for(int i = 0; i < 9; i++)
    {
        cout << "----------------------------------------" << endl;

        inputString = words[i];
```

```
115
116         cout << "Word : " << inputString << endl;
117
118         clearSpaceLowercasePunctuation(inputString);
119
120         if(recursion(inputString))
121             cout << "True - \"" << inputString << "\" is a Palindrome" << endl;
122         else
123             cout << "False - \"" << inputString << "\" is NOT a Palindrome" << endl;
124     }
125 }
126
127
```