

```

/*****
*   Author: Jessica Schuler
*   Date Created: 11-2-13
*   Last Modified Date: 11-3-13
*   Filename: a4.cpp
*
*   Overview:
*       This program lets a user input a word or words. It then with
*       either produce the reverse of the word or tell the user if
*       the word is a palindrome based on user selection.
*   Input:
*       The user will input a word or words.
*   Output:
*       The output will be the word reversed or it will tell the user
*       if the input is a palindrome.
*****/

#include<iostream>//library standard input/output
#include<algorithm>//library to use reverse
#include<string>//library to use string
#include<cctype>//library to use ispunct and isspace

using namespace std;//declares standard use of all entities

void rec_reverse(string input);//declares recursive reverse function
void ir_reverse(string input);//declares iterative reverse function
bool rec_palindrome(string input);//declares recursive palindrome function
bool ir_palindrome(string input);//declares iterative palindrom function
string rev(string input);//declares reverse string function
string rem_punc(string input);//declares func to remove punctuation+whitespace
string makeLower(string input);//declares function to convert to lowercase

int main()//starts main function
{
    string input="";//declares string input variable
    int type1, type2;//declares variables for picking functions

    cout<<"Let's have fun with words!"<<endl;
    cout<<"Enter a word or words you would like to play with: ";
    getline(cin,input);//gets the word input from the user

    cout<<"Your input converted to all lowercase is: ";
    cout<<makeLower(input)<<endl;//calls function to convert to lowercase

    cout<<"Your input without punctuation or whitespace is: ";
    //calls function to take out punctuation and whitespace
    cout<<rem_punc(makeLower(input))<<endl;

    cout<<endl;//added to break up output for readability

    //The following gets user input if they want Iterative
    //or recursive functions
    cout<<"Now lets pick the type of fuction you would like: "<<endl;
    cout<<"Enter a 1 if you would like a Recursive function, "<<endl;
    cout<<"Or a 2 if you would like an Iterative function: "<<endl;
    cin>>type1;
    while(type1<1||type1>2)//checks for valid input
    {
        cout<<"That entry is invalid, Please enter a 1 or 2 now!"<<endl;
        cin>>type1;
    }
}

```

```

    }

    cout<<endl;//added to break up output for readability

    //The following gets user input if they want to reverse
    //the word or if they want to check if it is a palindrome
    cout<<"Now pick the game you would like: "<<endl;
    cout<<"Enter a 1 to Reverse the letters of the word you input, "<<endl;
    cout<<"Or a 2 to see if your word is a Palindrome: "<<endl;
    cin>>type2;
    while(type2<1||type2>2)//checks for valid input
    {
        cout<<"That entry is invalid, Please enter a 1 or 2 now!"<<endl;
        cin>>type2;
    }

    switch (type1)//switch statement to proceed based on user input
    {
        case 1://for if the user picked Recursive functions
            switch (type2)
            {
                case 1://for if the user picked to reverse the word
                    cout<<"Your input reversed recursively: ";
                    rec_reverse(rem_punc(makeLower(input)));
                    cout<<endl;
                    break;
                case 2://for if the user picked to check for a palindrome
                    if(rec_palindrome(rem_punc(makeLower(input))))
                    {
                        cout<<"Recursively, your input IS a palindrome!"<<endl;
                    }
                    else
                    {
                        cout<<"Recursively, your input IS NOT a palindrome!"<<endl;
                    }
                    break;
            }
            break;
        case 2: //for if the user picked iterative functions
            switch (type2)
            {
                case 1://for if the user picked to reverse the word
                    cout<<"Your input reversed iteratively: ";
                    ir_reverse(rem_punc(makeLower(input)));
                    cout<<endl;
                    break;
                case 2://for if the user picked to check for a palindrome
                    if(ir_palindrome(rem_punc(makeLower(input))))
                    {
                        cout<<"Iteratively, your input IS a palindrome!"<<endl;
                    }
                    else
                    {
                        cout<<"Iteratively, your input IS NOT a palindrome!"<<endl;
                    }
                    break;
            }
    }
    return 0;//ends the main function
}

```

```

string makeLower(string input)//defines function to convert all letters to lowercase
{
    int strlen = input.length();

    for(int i=0; i<strlen; i++)
        input[i]=tolower(input[i]);//converts to lower case letters

    return input;    //returns lowercase input
}

string rem_punc(string input)//defines function to remove punctuation+whitespace
{
    int strlen = input.length();

    for(int i=0; i<strlen; i++)
    {
        if(ispunct(input[i]))
            input.erase(i,1);//erases any punctuation
        if(isspace(input[i]))
            input.erase(i,1);//erases any whitespace
    }
    return input;    //returns input without punctuation or whitespace
}

void ir_reverse(string input)//defines iterative reverse function
{
    reverse(input.begin(),input.end());//reverses order of letters
    cout<<input;//produces input reversed
}

void rec_reverse(string input)//defines recursive reverse function
{
    if(input == "")//base case
    {
        return;
    }
    else
    {
        rec_reverse(input.substr(1));//calls function to keep repeating
        cout<<input.at(0);
    }
}

string rev(string input)//function to reverse a string input
{
    int start =0;//sets starting place to 0
    int end = input.length();//finds the length of the word
    string temp(input);//temp place for the reversed input

    while (start<end)//while statement to keep going until the end is reached
    {
        end--;//takes off the end
        swap(temp[start], temp[end]);//swaps letters
        start++;//adds to the start
    }
    return temp;//returns the temp input reversed
}

bool ir_palindrome(string input)//defines iterative palindrome function
{

```

```
    return(input == rev(input)); //checks to see if input is equal to reverse input
}

bool rec_palindrome(string input) //defines recursive palindrome function
{
    int strlen = input.size(); //defines size of string input

    if(input[0] != input[strlen-1]) //1st base case
    {
        return false;
    }
    else if(strlen <= 1) //2nd base case
    {
        return true;
    }
    else
        return rec_palindrome(input.substr(1, strlen-2)); //calls function to repeat
}
```