/\* LAB ASSIGNMENT 9

A palindrome is a string of character that's the same forward and backward. Typically, punctuation, capitalization,

and spaces are ignored. For example, "Poor Dan is in a droop" is a palindrome, as can be seen by examining the

characters "poor danisina droop" and observing that they are the same forward and backward. One way to check for a

palindrome is to reverse the characters in the string and then compare with them the original-in a palindrome, the

sequence will be identical. Write C++ program with functions-

a) To print original string followed by reversed string using stack

b) To check whether given string is palindrome or not\*/

#include<iostream>

#include<string.h>

#define max 50

using namespace std;

class STACK

{

private:

char a[max];

int top;

public:

STACK() //constructor

{

top=-1; //stack is empty

}

void push(char);

void reverse();

void convert(char[]);

void palindrome();

};

void STACK::push(char c) //push describes insertion of element in stack

{

top++; //pre increamentation i.e. value is increamented first nd then declared

a[top] = c;

a[top+1]='\0';

}

void STACK::reverse()

{

char str[max];

cout<<"\n\n The Reverse string is : ";

for(int i=top,j=0; i>=0; i--,j++) //i is assigned to top which is getting decreamented further

{

cout<<a[i];

str[j]=a[i];

}

cout<<endl;

}

void STACK::convert(char str[]) //for convertion of capital case to small and removal of spaces between strings

{

int j,k,len = strlen(str);

for(j=0, k=0; j<len; j++)

{

if(((int)str[j]>=97 && (int)str[j]<=122) || ((int)str[j]>=65 && (int)str[j]<=90))

// 97 to 122 represents small case of alphabet in ascii value

// 65 to 90 represents upper case of alpahabets in ascii value

{

if( (int)str[j] <=90 ) //if ascii is less than 90 then 32 is added in it to get small case letters

{

str[k] = (char)( (int)str[j] + 32 );

}else

{

str[k] = str[j];

}

k++;

}

}

str[k]='\0';

cout<<endl<<"The Converted String is : "<<str<<"\n";

}

void STACK::palindrome() // for checking palindrome in the string

{

char str[max];

int i,j;

for(i=top,j=0; i>=0; i--,j++)

{

str[j]=a[i];

}

str[j]='\0';

if(strcmp(str,a) == 0) //if the string and the reverse of string are equal then it is a palindrome

cout<<"\n\nString is palindrome";

else

cout<<"\n\nString is not palindrome";

}

int main()

{

STACK;

char str[max];

int i=0;

cout<<"\nEnter string to be reversed and check is it palindrome or not : \n\n";

cin.getline(str , 50);//takes input stream & string as parameters & reads line of text from stream into string

stack.convert(str); //converts from the type of its class to another specified type

while(str[i] != '\0')

{

stack.push(str[i]);

i++;

}

stack.reverse();

stack.palindrome();

}