/\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Assignment no. 6

In any language program mostly syntax error occurs due to unbalancing delimiter

such as (),{},[]. Write C++ program using stack to check whether given expression

is well parenthesized or not.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*/

#include <iostream>

#include <string.h>

using namespace std;

char stk[10];

int size = 8;

int top = -1;

int stk\_Empty()

{

if (top == -1)

return 1;

else

return 0;

}

int stk\_Full()

{

if (top == size - 1)

return 1;

else

return 0;

}

void push(char ch)

{

top++;

stk[top] = ch;

}

char pop()

{

char x = stk[top];

top--;

return x;

}

void Display()

{

for (int i = 0; i <= top; i++)

cout << " " << stk[i];

}

//To check whether given expression is well parenthesized or not?

void check\_expr()

{

char expr[10], ch;

int i;

cout << "\nEnter the expression: ";

cin >> expr;

for (i = 0; expr[i] != '\0'; i++)

{

if (expr[i] == '(' || expr[i] == '[' || expr[i] == '{')

{

if (!stk\_Full())

push(expr[i]);

else

cout << "\n can't push as stack full\_\_\_!!!";

}

else

{

ch = pop();

if (expr[i] == ')' && ch != '(')

{

cout << "\nExpression is not well parenthesized";

push(ch);

break;

}

if (expr[i] == ']' && ch != '[')

{

cout << "\nExpression is not well parenthesized";

push(ch);

break;

}

if (expr[i] == '}' && ch != '{')

{

cout << "\nExpression is not well parenthesized";

push(ch);

break;

}

}

}

if (!stk\_Empty())

cout << "\nExpression is not well parenthesized";

else

cout << "\nExpression is well parenthesized !!";

}

int main()

{

cout << "Checking whether expression is well parenthesized or not?";

check\_expr();

cout << "\n\n";

return 0;

}