Practical No. 09 (Group D)

Name : Atharva B. Iparkar

Roll no : S211045

Class : S.E.

Div : A

Batch : A-2

Problem Statement :

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.

Code :

```cpp
#include <iostream>
#include <cstring>  // Use for strlen
using namespace std;

class braces {
    char st[20]; // Stack array
    int top;

public:
```

```cpp
    braces() {

        top = -1; // Initialize stack as empty

    }

    void push(char a);

    void pop();

    bool isEmpty();

    void input();

};


void braces::push(char a) {

    if (top == 19) {  // Prevent overflow

        cout << "\nStack overflow, expression is too long.";

        return;

    }

    top++;

    st[top] = a;

}


void braces::pop() {

    if (top == -1) { // Prevent underflow

        cout << "\nStack underflow, no matching opening brace.";

        return;

    }

    top--;
```

```cpp
}

bool braces::isEmpty() {
    return top == -1; // Check if stack is empty
}

void braces::input() {
    char ch[20];  // Input expression
    int i = 0;
    cout << "\nEnter the expression: ";
    cin >> ch;

    while (i < strlen(ch)) {
        // Push opening braces onto stack
        if (ch[i] == '{' || ch[i] == '[' || ch[i] == '(') {
            push(ch[i]);
        }
        // Handle closing braces
        if (ch[i] == '}') {
            if (top != -1 && st[top] == '{')
                pop();
            else {
                cout << "\nMatching opening brace '{' is not found.";
                return;
```

```cpp
        }

    }

    if (ch[i] == ']') {

        if (top != -1 && st[top] == '[')

            pop();

        else {

            cout << "\nMatching opening brace '[' is not found.";

            return;

        }

    }

    if (ch[i] == ')') {

        if (top != -1 && st[top] == '(')

            pop();

        else {

            cout << "\nMatching opening brace '(' is not found.";

            return;

        }

    }

    i++;

}


// After processing the expression, check for unmatched opening braces

if (isEmpty()) {

    cout << "\nExpression is well-parenthesized.";
```

```cpp
        } else {

            while (!isEmpty()) {

                if (st[top] == '{')

                    cout << "\nMatching closing brace '}' is not found.";

                if (st[top] == '[')

                    cout << "\nMatching closing brace ']' is not found.";

                if (st[top] == '(')

                    cout << "\nMatching closing brace ')' is not found.";

                pop();

            }

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

        }

}


int main() {

    braces c;

    c.input();

    return 0;

}
```

Output :

```
user@user-VirtualBox:~/S211045_Atharva$ g++ Practical9.cpp -o p
user@user-VirtualBox:~/S211045_Atharva$ ./p

Enter the expression: (

Matching closing brace ')' is not found.
Expression is not well-parenthesized.user@user-VirtualBox:~/S211045_Atharva$ ./p

Enter the expression: (a+b-c)}

Matching opening brace '{' is not found.user@user-VirtualBox:~/S211045_Atharva$ ./p

Enter the expression: (a+b)

user@user-VirtualBox:~/S211045_Atharva$
```