

Question3.cpp

```
1  #include <iostream>
2  #include <string>
3
4  using namespace std;
5
6  class stack
7  {
8      char arr[20];
9      int top = -1;
10
11  public:
12      stack()
13      {
14          for (int i = 0; i < 20; i++)
15          {
16              arr[i] = 0;
17          }
18      }
19      bool full()
20      {
21          return top == 19;
22      }
23      bool empty()
24      {
25          return top == -1;
26      }
27
28      void push(char value)
29      {
30          if (!full())
31          {
32              top++;
33              arr[top] = value;
34          }
35          else
36          {
37              cout << "Array is Full.\n";
38              return;
39          }
40      }
41      void pop()
42      {
43          if (!empty())
44          {
45              top--;
46          }
47          else
48          {
49              cout << "Stack is Empty!" << endl;
50          }
51      }
52      char Top()
53      {
```

```
54     if (!empty())
55     {
56         return arr[top];
57     }
58     else
59     {
60         cout << "Array Empty, Returning 'X'...\n";
61         return 'X';
62     }
63 }
64 void display()
65 {
66     cout << "\n--";
67     for (int i = 0; i <= top; i++)
68     {
69         cout << arr[i];
70     }
71     cout << "--\n";
72 }
73 };
74
75 bool MatchDelim(string ex)
76 {
77     stack delims;
78     for (int i = 0; i < ex.length(); i++)
79     {
80         if (ex[i] == '{' || ex[i] == '(' || ex[i] == '[')
81         {
82             delims.push(ex[i]);
83         }
84         else if (ex[i] == '}' || ex[i] == ')' || ex[i] == ']')
85         {
86             if (ex[i] == ')' && delims.Top() == '(')
87             {
88                 delims.pop();
89             }
90             else if (ex[i] == ']' && delims.Top() == '[')
91             {
92                 delims.pop();
93             }
94             else if (ex[i] == '}' && delims.Top() == '{')
95             {
96                 delims.pop();
97             }
98             else
99             {
100                 cout << "Open and Close Brackets Mismatched at position " << i << endl;
101                 return false;
102             }
103         }
104     }
105     if (delims.empty())
106     {
107         cout << "Expression is valid\n";
108         return true;
109     }
```

```
110     else
111     {
112         cout << "Unclosed Delimiters Remain\n";
113         delims.display();
114         return false;
115     }
116 }
117
118 int main()
119 {
120     string exp;
121     cout << "Enter Expressions to match the Delimiters:\n";
122     getline(cin, exp);
123     MatchDelim(exp);
124 }
```