

NAME – RAJDEEP JAISWAL

DATE – 10 Oct 2021

BRANCH – BTECH CSE

SEC = 13 A

UID -20BCS2761

SUB- DS LAB Worksheet

Write a program to input two stacks and compare their contents.

Solution

Algorithm –

1. Take a flag variable and set it to true initially, *flag = true*. This variable will indicate whether the stacks are same or not.
2. First check if the size of given **stack1** and **stack2** are equal. If the size is not equal, set flag to false and return it.
3. If the size is same, then compare the top elements of both of the given stacks.
4. If the top of both stacks is **NOT** same, set flag to false and return it otherwise pop top elements of both stacks.
5. Repeat step 3 and 4 until all elements are popped out from both of the stacks.
6. If both stacks gets empty and the flag variable is still true, it means that the stacks are same.

Code In text Forms –

```
#include <bits/stdc++.h>
#include <iostream>
using namespace std;

bool isSameStack(stack<string> stack1, stack<string> stack2)
{
    // Create a flag variable
    bool flag = true;

    if (stack1.size() != stack2.size()) {
        flag = false;
        return flag;
    }

    while (stack1.empty() == false) {
        // If the top elements of both stacks
        // are same
        if (stack1.top() == stack2.top()) {
            // Pop top of both stacks
            stack1.pop();
        }
    }
}
```

```
        stack2.pop();
    }
    else {

        flag = false;
        break;
    }
}

// Return flag
return flag;
}

// Driver Code
int main()
{
    // Creating stacks
    stack<string> stack1;
    stack<string> stack2;

    // Inserting elements to stack1
    stack1.push("Rajdeep");
    stack1.push("18");
    stack1.push("Chandigarh University");
    stack1.push("Welcomes");
    stack1.push("You");

    // Inserting elements to stack2
    stack2.push("Rajdeep");
    stack2.push("18");
    stack2.push("Chandigarh University");
    stack2.push("Welcomes");
    stack2.push("You");

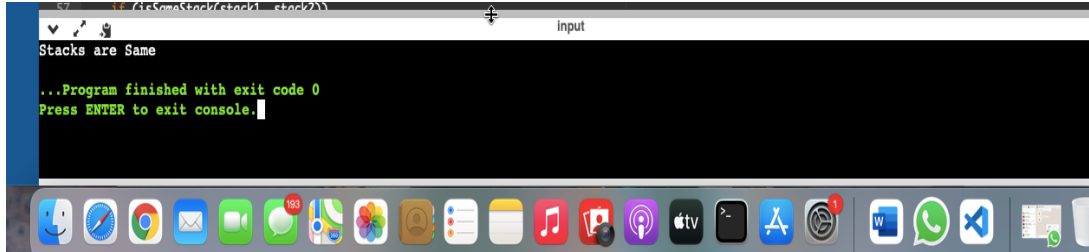
    if (isSameStack(stack1, stack2))
        cout << "Stacks are Same";
    else
        cout << "Stacks are not Same";

    return 0;
}
```

Code In Compiler –

```
main.cpp
1 #include <bits/stdc++.h>
2 #include <iostream>
3 using namespace std;
4
5 bool isSameStack(stack<string> stack1, stack<string> stack2)
6 {
7     // Create a flag variable
8     bool flag = true;
9
10
11     if (stack1.size() != stack2.size()) {
12         flag = false;
13         return flag;
14     }
15
16
17     while (stack1.empty() == false) {
18         // If the top elements of both stacks
19         // are same
20         if (stack1.top() == stack2.top()) {
21             // Pop top of both stacks
22             stack1.pop();
23             stack2.pop();
24         }
25         else {
26
27             flag = false;
28             break;
29         }
30     }
31
32     // Return flag
33     return flag;
34 }
35
36 // Driver Code
37 int main()
38 {
39     // Creating stacks
40     stack<string> stack1;
41     stack<string> stack2;
42
43     // Inserting elements to stack1
44     stack1.push("Rajdeep");
45     stack1.push("18");
46     stack1.push("Chandigarh University");
47     stack1.push("Welcomes");
48     stack1.push("You");
49
50     // Inserting elements to stack2
51     stack2.push("Rajdeep");
52     stack2.push("18");
53     stack2.push("Chandigarh University");
54     stack2.push("Welcomes");
55     stack2.push("You");
56
57     if (isSameStack(stack1, stack2))
58         cout << "Stacks are Same";
59     else
60         cout << "Stacks are not Same";
61
62     return 0;
63 }
64
65
```

OUTPUT



```
Stacks are Same  
...Program finished with exit code 0  
Press ENTER to exit console.
```

Learning outcomes (What I have learnt):

- 1.
- 2.
- 3.
- 4.
- 5.

I

Evaluation Grid (To be created as per the SOP and Assessment guidelines by the faculty):

Sr. No.	Parameters	Marks Obtained	Maximum Marks
1.			
2.			
3.			