Stack in C++ STL

Difficulty Level : • Last Updated : 06

adaptors with LIFO(Last In First Out) type of working, where a new element is added at one end and (top) an element is removed from that end only.

The functions associated with stack are:
empty() – Returns whether the stack is empty – Time Complexity : O(1)
size() – Returns the size of the stack – Time Complexity : O(1)
top() – Returns a reference to the top most element of the stack – Time Complexity : O(1)
push(g) – Adds the element 'g' at the top of the stack – Time Complexity : O(1)
pop() – Deletes the top most element of the stack – Time Complexity : O(1)

```cpp
// CPP program to demonstr
#include <bits/stdc++.h>
using namespace std;

void showstack(stack <int>
{
    while (!s.empty())
    {
        cout << '\t' << s.
        s.pop();
    }
    cout << '\n';
}

int main ()
{
    stack <int> s;
    s.push(10);
    s.push(30);
    s.push(20);
    s.push(5);
    s.push(1);

    cout << "The stack is
    showstack(s);

    cout << "\ns.size() :
    cout << "\ns.top() : "


    cout << "\ns.pop() : "
    s.pop();
    showstack(s);

    return 0;
}
```

Output:

```
The stack is :     1    5

s.size() : 5
s.top() : 1
s.pop() :     5    20    30
```

List of functions of Stack:

- stack::top() in C++ STL
- stack::empty() and stack::size() in C++ STL
- stack::push() and stack::pop() in C++ STL
- stack::swap() in C++ STL
- stack::emplace() in C++ STL
- Recent Articles on C++ Stack

Please write comments if you find anything incorrect, or you want to share more information about the topic discussed above

RECOMMENDED ARTICLES