**stack**

template <class T, class Container = deque<T> > class stack;

LIFO stack

Stacks are a type of container adaptor, specifically designed to operate in a LIFO context (last-in first-out), where elements are inserted and extracted only from one end of the container.  
  
**stack**s are implemented as *containers adaptors*, which are classes that use an encapsulated object of a specific container class as its *underlying container*, providing a specific set of member functions to access its elements. Elements are *pushed*/*popped* from the *"back"* of the specific container, which is known as the *top* of the stack.  
  
The underlying container may be any of the standard container class templates or some other specifically designed container class. The container shall support the following operations:

* empty
* size
* back
* push\_back
* pop\_back

The standard container classes [vector](http://www.cplusplus.com/vector), [deque](http://www.cplusplus.com/deque) and [list](http://www.cplusplus.com/list) fulfill these requirements. By default, if no container class is specified for a particular stack class instantiation, the standard container [deque](http://www.cplusplus.com/deque) is used.

**Template parameters**

T

Type of the elements.  
Aliased as member type stack::value\_type.

Container

Type of the internal *underlying container* object where the elements are stored.  
Its value\_type shall be T.  
Aliased as member type stack::container\_type.

**Member types**

* [C++98](javascript:switch1.select(1))
* [C++11](javascript:switch1.select(2))

|  |  |  |
| --- | --- | --- |
| **member type** | **definition** | **notes** |
| value\_type | The first template parameter (T) | Type of the elements |
| container\_type | The second template parameter (Container) | Type of the *underlying container* |
| size\_type | an unsigned integral type | usually the same as [size\_t](http://www.cplusplus.com/size_t) |

**Member functions**

[**(constructor)**](http://www.cplusplus.com/reference/stack/stack/stack/)

Construct stack (public member function )

[**empty**](http://www.cplusplus.com/reference/stack/stack/empty/)

Test whether container is empty (public member function )

[**size**](http://www.cplusplus.com/reference/stack/stack/size/)

Return size (public member function )

[**top**](http://www.cplusplus.com/reference/stack/stack/top/)

Access next element (public member function )

[**push**](http://www.cplusplus.com/reference/stack/stack/push/)

Insert element (public member function )

[**emplace**](http://www.cplusplus.com/reference/stack/stack/emplace/)

Construct and insert element (public member function )

[**pop**](http://www.cplusplus.com/reference/stack/stack/pop/)

Remove top element (public member function )

[**swap**](http://www.cplusplus.com/reference/stack/stack/swap/)

Swap contents (public member function )

**Non-member function overloads**

[**relational operators**](http://www.cplusplus.com/reference/stack/stack/operators/)

Relational operators for stack (function )

[**swap (stack)**](http://www.cplusplus.com/reference/stack/stack/swap-free/)

Exchange contents of stacks (public member function )

**Non-member class specializations**