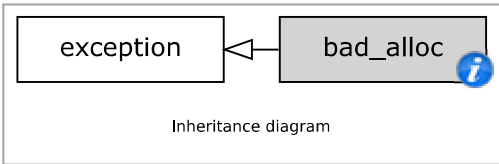# std::bad_alloc

```
class bad_alloc;
```

`std::bad_alloc` is the type of the object thrown as exceptions by the allocation functions to report failure to allocate storage.

Inheritance diagram

Member functions

(constructor)	constructs the bad_alloc object (public member function)
operator=	replaces a bad_alloc object (public member function)
what	returns explanatory string (public member function)

std::bad_alloc::bad_alloc

```
bad_alloc() throw();          (until C++11)
bad_alloc() noexcept;         (since C++11)
```

Constructs new `bad_alloc` object with an implementation-defined null-terminated byte string which is accessible through `what()`.

Parameters

(none)

std::bad_alloc::operator=

```
bad_alloc& operator=( const bad_alloc& other ) throw();      (until C++11)
bad_alloc& operator=( const bad_alloc& other ) noexcept;     (since C++11)
```

Assigns the contents of `other`.

Parameters

other - another exception object to assign

Return value

`*this`

---

std::bad_alloc::what

`virtual const char* what() const throw();`	(until C++11)
`virtual const char* what() const noexcept;`	(since C++11)

Returns the explanatory string.

Parameters

(none)

Return value

Pointer to a null-terminated string with explanatory information.

Inherited from std::exception

Member functions

(destructor) [virtual]	destroys the exception object (virtual public member function of `std::exception`)
what [virtual]	returns an explanatory string (virtual public member function of `std::exception`)

Example

Run this code

```cpp
#include <iostream>
#include <new>

int main()
{
    try {
        while (true) {
            new int[100000000ul];
        }
    } catch (const std::bad_alloc& e) {
        std::cout << "Allocation failed: " << e.what() << '\n';
    }
}
```

Possible output:

```
Allocation failed: std::bad_alloc
```

See also

operator new **operator new[]**	allocation functions (function)