

std::unordered_set

在标头 <unordered_set> 定义

```
template<
    class Key,
    class Hash = std::hash<Key>,
    class KeyEqual = std::equal_to<Key>,
    class Allocator = std::allocator<Key>
> class unordered_set;

namespace pmr {
    template <class Key,
        class Hash = std::hash<Key>,
        class Pred = std::equal_to<Key>>
        using unordered_set = std::unordered_set<Key, Hash, Pred,
            std::pmr::polymorphic_allocator<Key>>;
}
```

unordered_set 是含有 Key 类型唯一对象集合的关联容器。搜索、插入和移除拥有平均常数时间复杂度。

在内部，元素并不以任何特别顺序排序，而是组织进桶中。元素被放进哪个桶完全依赖其值的哈希。这允许对单独元素的快速访问，因为哈希一旦确定，就准确指代元素被放入的桶。

不可修改容器元素（即使通过非 const 迭代器），因为修改可能更改元素的哈希，并破坏容器。

std::unordered_set 满足容器 (Container)、知分配器容器 (AllocatorAwareContainer)、无序关联容器 (UnorderedAssociativeContainer) 的要求。

成员类型

成员类型	定义
key_type	Key
value_type	Key
size_type	无符号整数类型（通常是 std::size_t）
difference_type	有符号整数类型（通常是 std::ptrdiff_t）
hasher	Hash
key_equal	KeyEqual
allocator_type	Allocator
reference	value_type&
const_reference	const value_type&
pointer	std::allocator_traits<Allocator>::pointer
const_pointer	std::allocator_traits<Allocator>::const_pointer
iterator	指向 value_type 的常老式向前迭代器 (LegacyForwardIterator)
const_iterator	指向 const value_type 的老式向前迭代器 (LegacyForwardIterator)

local_iterator	类别、值、差、指针和引用类型都与 iterator 相同的迭代器类型。 能用此迭代器在单个桶迭代，但不能跨桶。
const_local_iterator	类别、值、差、指针和引用类型都与 const_iterator 相同的迭代器类型。 能用此迭代器在单个桶迭代，但不能跨桶。
node_type(C++17 起)	表示容器结点的结点把柄特化
描述插入 node_type 结果的类型，下列类型的特化	
insert_return_type(C++17 起)	<div><pre>template<class Iter, class NodeType> struct /*未指定*/ { Iter position; bool inserted; NodeType node; };</pre></div> 以模板实参 iterator 和 node_type 实例化。

成员函数

(构造函数) (C++11)	构造 unordered_set (公开成员函数)
(析构函数) (C++11)	析构 unordered_set (公开成员函数)
operator= (C++11)	赋值给容器 (公开成员函数)
get_allocator (C++11)	返回相关的分配器 (公开成员函数)

迭代器

begin cbegin (C++11)	返回指向起始的迭代器 (公开成员函数)
end cend (C++11)	返回指向末尾的迭代器 (公开成员函数)

容量

empty (C++11)	检查容器是否为空 (公开成员函数)
size (C++11)	返回容纳的元素数 (公开成员函数)
max_size (C++11)	返回可容纳的最大元素数 (公开成员函数)

修改器

clear (C++11)	清除内容 (公开成员函数)
---------------	------------------

insert (C++11)	插入元素或结点 (C++17 起) (公开成员函数)
emplace (C++11)	原位构造元素 (公开成员函数)
emplace_hint (C++11)	使用提示原位构造元素 (公开成员函数)
erase (C++11)	擦除元素 (公开成员函数)
swap (C++11)	交换内容 (公开成员函数)
extract (C++17)	从另一容器释出结点 (公开成员函数)
merge (C++17)	从另一容器接合结点 (公开成员函数)

查找

count (C++11)	返回匹配特定键的元素数量 (公开成员函数)
find (C++11)	寻找带有特定键的元素 (公开成员函数)
contains (C++20)	检查容器是否含有带特定键的元素 (公开成员函数)
equal_range (C++11)	返回匹配特定键的元素范围 (公开成员函数)

桶接口

begin (size_type) (C++11) cbegin (size_type)	返回一个迭代器，指向指定的桶的开始 (公开成员函数)
end (size_type) (C++11) cend (size_type)	返回一个迭代器，指向指定的桶的末尾 (公开成员函数)
bucket_count (C++11)	返回桶数 (公开成员函数)
max_bucket_count (C++11)	返回桶的最大数量 (公开成员函数)
bucket_size (C++11)	返回在特定的桶中的元素数量 (公开成员函数)
bucket (C++11)	返回带有特定键的桶 (公开成员函数)

哈希策略

load_factor (C++11)	返回每个桶的平均元素数量 (公开成员函数)
max_load_factor (C++11)	管理每个桶的平均元素数量的最大值 (公开成员函数)
rehash (C++11)	为至少为指定数量的桶预留存储空间并重新生成散列表 (公开成员函数)

reserve (C++11)	为至少为指定数量的元素预留存储空间并重新生成哈希表。 (公开成员函数)
------------------------	----------------------------------------

观察器

hash_function (C++11)	返回用于对键散列的函数 (公开成员函数)
key_eq (C++11)	返回用于比较键的相等性的函数 (公开成员函数)

非成员函数

operator== operator!= (C++20 中移除)	比较 unordered_set 中的值 (函数模板)
std::swap (std::unordered_set) (C++11)	特化 std::swap 算法 (函数模板)
erase_if (std::unordered_set) (C++20)	擦除所有满足特定判别标准的元素 (函数模板)

推导指引(C++17 起)

注意

成员类型 `iterator` 与 `const_iterator` 可能是同一类型的别名。这表明以这两个类型为参数类型的一对函数重载可能违背单一定义规则。因为 `iterator` 可转换到 `const_iterator`，所以可以改成只提供一个以 `const_iterator` 为参数类型的函数。
来自“https://zh.cppreference.com/mwiki/index.php?title=cpp/container/unordered_set&oldid=67006”