Multiset Built-in Functions in C++ STL

1. Constructor

Name	Details	Time Complexity
multiset<type> ms;	Construct an empty multiset.	O(1)
multiset<type> ms(cmp);	Construct an empty multiset with a custom comparison function.	O(1)
multiset<type> ms(it1, it2);	Construct a multiset with elements from the range [it1, it2).	O(N log N)
multiset<type> ms(ms2);	Construct a multiset by copying another multiset ms2.	O(N)

2. Capacity

Name	Details	Time Complexity
ms.size()	Returns the number of elements in the multiset.	O(1)
ms.max_size()	Returns the maximum size the multiset can hold.	O(1)
ms.empty()	Returns true if the multiset is empty, otherwise false.	O(1)

3. Modifiers

Name	Details	Time Complexity
ms.insert(val)	Adds an element val to the multiset.	O(log N)
ms.erase(val)	Removes all elements equal to val from the multiset.	O(log N)
ms.erase(it)	Removes the element at iterator it.	O(1)
ms.clear()	Clears all elements from the multiset.	O(N)
ms.swap(ms2)	Swaps contents with another multiset ms2.	O(1)
ms.emplace(args...)	Inserts a new element in place constructed with args.	O(log N)
ms.insert(it1, it2)	Inserts elements from the range [it1, it2).	O(N log N)

4. Element Access

Name	Details	Time Complexity
ms.find(val)	Finds an element with the value val.	O(log N)
ms.count(val)	Counts elements with the value val.	O(log N)
ms.lower_bound(val)	Returns an iterator to the first element that is not less than val.	O(log N)
ms.upper_bound(val)	Returns an iterator to the first element that is greater than val.	O(log N)
ms.equal_range(val)	Returns a pair of iterators of the range that includes all the elements equiva	O(log N)

5. Iterators

Name	Details	Time Complexity
ms.begin()	Returns an iterator to the first element.	O(1)
ms.end()	Returns an iterator to the element following the last element.	O(1)

ms.rbegin()	Returns a reverse iterator to the last element.	O(1)
ms.rend()	Returns a reverse iterator to the element preceding the first element.	O(1)