1. Map: map<int,int>mp;

Function	Definition
map::insert()	Insert elements with a particular key in the map container -> O(log n)
map:: count()	Returns the number of matches to element with key-value 'g' in the map. -> O(log n)
map equal_range()	Returns an iterator of pairs. The pair refers to the bounds of a range that includes all the elements in the container which have a key equivalent to k.
map erase()	Used to erase elements from the container -> O(log n)
map rend()	Returns a reverse iterator pointing to the theoretical element right before the first key-value pair in the map(which is considered its reverse end).
map rbegin()	Returns a reverse iterator which points to the last element of the map.
map find()	Returns an iterator to the element with key-value 'g' in the map if found, else returns the iterator to end.
map crbegin() and crend()	crbegin() returns a constant reverse iterator referring to the last element in the map container. crend() returns a constant reverse iterator pointing to the theoretical element before the first element in the map.
map cbegin() and cend()	cbegin() returns a constant iterator referring to the first element in the map container. cend() returns a constant iterator pointing to the theoretical element that follows the last element in the multimap.
map emplace()	Inserts the key and its element in the map container.
map max_size()	Returns the maximum number of elements a map container can hold -> O(1)
map upper_bound()	Returns an iterator to the first element that is equivalent to mapped value with key-value 'g' or definitely will go after the element with key-value 'g' in the map
map operator=	Assigns contents of a container to a different container, replacing its current content.
map lower_bound()	Returns an iterator to the first element that is equivalent to the mapped value with key-value 'g' or definitely will not go before the element with key-value 'g' in the map -> O(log n)
map emplace_hint()	Inserts the key and its element in the map container with a given hint.
map value_comp()	Returns the object that determines how the elements in the map are ordered ('<' by default).
map key_comp()	Returns the object that determines how the elements in the map are ordered ('<' by default).
map::size()	Returns the number of elements in the map.
map::empty()	Returns whether the map is empty
map::begin() and end()	begin() returns an iterator to the first element in the map. end() returns an iterator to the theoretical element that follows the last element in the map
map::operator[]	This operator is used to reference the element present at the position given inside the operator.
map::clear()	Removes all the elements from the map.
map::at() and map::swap()	at() function is used to return the reference to the element associated with the key k. swap() function is used to exchange the contents of two maps but the maps must be of the same type, although sizes may differ.