Vector:

<https://www.geeksforgeeks.org/vector-in-cpp-stl/>

push\_back(): Inserts a new element at the end of the vector. Its time complexity is **O(1)**.

size(): Returns the number of elements in the vector. Its time complexity is **O(1)**.

Vector capacity: <https://www.geeksforgeeks.org/vector-capacity-function-in-c-stl/>

Note:

The time complexity to find an element in `std::vector` by linear search is **O(N)**. It is O(log N) for `std::map` and O(1) for `std::unordered\_map`. However, the complexity notation ignores constant factors. Different containers have various traversal overheads to find an element.

Set:

insert(): insert a new element. Its time complexity is **O(logN)** where N is the size of the set. size(): Returns the size of the set or the number of elements in the set. Its time complexity is O(1)

setname. erase(startingposition, endingposition) – **O(n)**

The time complexity of set\_name. find( key ) is **O( log N )**

**Unordered set:**

<https://www.geeksforgeeks.org/unordered_set-in-cpp-stl/>

many ways to create vector

<https://www.geeksforgeeks.org/initialize-a-vector-in-cpp-different-ways/>

{{1,3}, {2,4}, {5,7}, {6,8}} store in vector<vector>ans

1 3

2 4

5 7

6 8

Can use it in ans[0][0]=1

Ans[0][1]=3

Ans[2][1]=7