

**NOTRE DAME** UNIVERSITY

BANGLADESH

**Computer Club**

**Competitive Programming Class Documentation**

**Level : Intermediate**

**Instructor :** [**MD. Tanvir Rahman Tareq**](https://bd.linkedin.com/in/tanvirtareq)

**Github :** [**Tanvir Tareq**](https://github.com/tanvirtareq)

**Schedule : Saturday, 10am.**

**Room No. : Computer Lab-418**

**Lecture-01**

**Date : 12th April, 2025**

**Topic : Standard Template Library (STL)**

* **Discussed About :-**

1. What is STL
2. Uses of Vector
3. Uses of Set
4. Uses of Multiset
5. Order\_set
6. Pair
7. Template for Order\_set → [Link](https://github.com/tanvirtareq/sublime_text_snippet/blob/main/ordered_set.sublime-snippet)

* **Discussed Problems :-**

1. Stack – Leetcode [678. Valid Parenthesis String](https://leetcode.com/problems/valid-parenthesis-string/)
2. Inverse Number counter using odered\_set [Link](https://afteracademy.com/blog/inversion-count-in-an-array/)

* **Next Discussion Topic :-**

1. Map
2. Stack
3. Queue
4. Priority Queue
5. Dequeue

*Possible Issues : Code::Blocks Version was not updated for STL. For ignoring Compiler issues we have used* [*Usaco IDE*](https://ide.usaco.guide/)

**Class-Code :**

#include<bits/stdc++.h>

using namespace std;

#include <ext/pb\_ds/assoc\_container.hpp>

#include <ext/pb\_ds/tree\_policy.hpp>

using namespace std;

using namespace \_\_gnu\_pbds;

*// Define the ordered\_set template with a customizable comparator*

template<typename T, typename Compare = less<T>>

using ordered\_set = tree<T, null\_type, Compare, rb\_tree\_tag, tree\_order\_statistics\_node\_update>;

int main() {

ios\_base::sync\_with\_stdio(false);

cin.tie(NULL);

ordered\_set<int> st;

st.insert(0);

st.insert(1);

st.insert(2);

st.insert(5);

st.insert(7);

st.insert(8);

cout<<\*st.find\_by\_order(3)<<endl;

*// cout<<st.order\_of\_key(6)<<endl;*

*// cout<<\*st.lower\_bound(6)<<'\n';*

*// cout<<distance(st.begin(), st.lower\_bound(6))<<endl;*

return 0;

}

**Inverse Number Counter :-**

#include<bits/stdc++.h>

using namespace std;

#include <ext/pb\_ds/assoc\_container.hpp>

#include <ext/pb\_ds/tree\_policy.hpp>

using namespace std;

using namespace \_\_gnu\_pbds;

*// Define the ordered\_set template with a customizable comparator*

template<typename T, typename Compare = less<T>>

using ordered\_set = tree<T, null\_type, Compare, rb\_tree\_tag, tree\_order\_statistics\_node\_update>;

int main ()

{

ios\_base::sync\_with\_stdio(false);

cin.tie(NULL);

int n;

cin >> n;

vector<int> v;

for (int i=0; i<n; i++){

int x;

cin >> x;

v.push\_back(x);

}

ordered\_set<pair<int, int> > ost;

int ans = 0;

for (int i=v.size()-1; i>=0; i++){

ans+=ost.order\_of\_key({v[i],-1});

ost.insert({v[i],i});

}

cout << ans << "\n";

return 0;

}