# STL Pairs, sort() and structure in C++

## pop\_back() in vector

* Removes the last element of a vector

| vector<int> vec={1,2,3,4,5}; // (size=5) vec.pop\_back(); |
| --- |

After pop\_back(),

vec becomes {1,2,3,4}; (size=4)

**Time Complexity = O(1)**

## sort()

**Time Complexity = O(n log n )**

vec:={5,1,7,3,4,9}

**syntax->** **sort(starting\_iterator, ending\_iterator)**

**starting\_iterator->** vec.begin();

**ending\_iterator->** vec.end();

**Example**

| sort(vec.begin(),vec.end()); |
| --- |

## swap()

**Syntax: swap(a, b);**

**swap() is used to interchange the values of any 2 variables**

**Example:**

| int a=5; int b=6; swap(a,b); cout<<a<<endl; cout<<b<<endl; |
| --- |

**Example for sort() :**

Let n = size of vector

vec:= {1,2,3,4,5,6};

Staring\_itr -> vec.begin();

Second\_itr -> vec.begin()+1;

Third\_itr -> vec.begin()+2;

..

Last\_itr -> vec.begin()+n-1;

vec.begin()+n == vec.end();

sort(vec.begin(), vec.begin()+n);

sort(vec.begin(), vec.end());

L---r sort(l,r+1);

## reverse()

**Time complexity: O(n)**

vec-> {4,1,2,8,3};

**How will you sort in decreasing order ?**

| sort(vec.begin(),vec.end()); // {1,2,3,4,8} reverse(vec.begin(),vec.end()); // {8,4,3,2,1} |
| --- |

**How do we sort in case of array?**

| int arr[5] = {4,1,2,8,3}; |
| --- |

n = size of array (Here, it is 5)

Staring\_itr -> arr;

Second\_itr -> arr+1;

Third\_itr -> arr+2;

..

Last\_itr -> arr+n-1;

Ending\_itr -> last\_itr+1 == arr+n-1+1 == arr+n;

| sort(arr, arr + n); |
| --- |

**Code-1 (To reverse an array without using reverse() )**

| #include <bits/stdc++.h> using namespace std; int main(){   int arr[5]={4,1,2,8,3};   int n=sizeof(arr)/sizeof(int);   int l=0,r=n-1;  while(l<=r){  swap(arr[l],arr[r]);  l++;  r--;  }  for(int i=0;i<n;i++) cout<<arr[i]<<" ";   return 0; } |
| --- |

## Struct

**Code-1**

| #include <bits/stdc++.h> using namespace std;  struct Freshers{  string name;  string AdmNo;  int age;  double height; };  // structure definition ends with a semicolon (;) int main(){    Freshers fresher;  fresher.name = "Manyank";  fresher.AdmNo = "20JE0655";  fresher.age = 18;  fresher.height = 6.1;    cout<<"Info of freshers is :"<<endl;  cout<<fresher.name<<endl;  cout<<fresher.AdmNo<<endl;  cout<<fresher.age<<endl;  cout<<fresher.height<<endl;  return 0; } |
| --- |

**Code-2**

| #include <bits/stdc++.h> using namespace std;  struct Point{  int x;  int y; }; int main(){    Point point[n];  for(int i=0;i<n;i++) cin>>point[i].x>>point[i].y;    (x1,y1);  (x2,y2);  (x3,y3);  ...  (xn,yn);    int X[n];  int Y[n];  for(int i=0;i<n;i++) cin>>X[i]>>Y[i];    (xi,yi);    cout<<X[i]<<" "<<Y[i]<<endl;      return 0; } |
| --- |
|  |

**Pair in C++ STL**

| #include <bits/stdc++.h>  using namespace std;  int main(){    pair<int,int> point;    cin>>point.first>>point.second;  cout<<point.first<<" "<<point.second<<endl;    pair<string,double> Fresher;    cin>>Fresher.first;  cin>>Fresher.second;    cout<<"The name of the student is: " << Fresher.first << endl;  cout<<"The height of the student is: " << Fresher.second << endl;   return 0; } |
| --- |
|  |

**Point in 3D->**

| #include <bits/stdc++.h> using namespace std; int main(){    pair<int,pair<int,int>> point3D;  // x-> point3D.first;  // y-> point3D.second.first;  // z-> point3D.second.second;   return 0; } |
| --- |

**Hackerrank Problem: Equalize the array**

**Link:** [**https://www.hackerrank.com/challenges/equality-in-a-array/problem**](https://www.hackerrank.com/challenges/equality-in-a-array/problem)

**(First, try it by yourself, then only look at the solution below)**

#include<bits/stdc++.h>

using namespace std;

int main()

{

int n;

cin>>n;

int a[n]; // 3 1 1 2 2 2 3 3 3

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

cin>>a[i];

}

sort(a,a+n); // 1 1 2 2 2 3 3 3 3

int ans = 10000;

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

int freq=0;

int com = a[i];

while(i<n&&a[i]==com){

freq++;

i++;

}

i--;

ans = min(ans,n-freq);

}

cout<<ans;

}

**Try the below problem, it would be discussed in next class:**

<https://atcoder.jp/contests/abc187/tasks/abc187_d>