题意：数组里的数字可以任意更换位置

问以最优方式替换

最多有多少个数字在替换后比替换前大

考虑贪心：自然是大的数放在比它小的数的最大数 位置上最优

这样，先给数组排序，之后用双指针扫描

#include <bits/stdc++.h>

using namespace std;

#define INF 1000000007

typedef long long ll;

int a[100010];

bool b[100010];

int main()

{

// #ifndef ONLINE\_JUDGE

// freopen("input.txt","r",stdin);

// #endif // ONLINE\_JUDGE

//

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

cin.tie(0);

int n;

cin>>n;

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

cin>>a[i];

sort(a,a+n);

//memset(b,0,sizeof(b));

int ans(0);

int j=1;

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

while(a[i]==a[j] && j<n)j++;

if(j==n)break;

ans++;

j++;

}

cout<<ans<<"\n";

return 0;

}

或者用二分查找

#include <bits/stdc++.h>

using namespace std;

#define INF 1000000007

typedef long long ll;

vector<int> a,b;

int main()

{

// #ifndef ONLINE\_JUDGE

// freopen("input.txt","r",stdin);

// #endif // ONLINE\_JUDGE

//

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

cin.tie(0);

int n,x;

cin>>n;

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

cin>>x;

a.push\_back(x);

b.push\_back(x);

}

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

vector<int>::iterator it;

int ans(0);

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

it=lower\_bound(b.begin(),b.end(),a[i]+1);

if(it!=b.end()){

ans++;

b.erase(it);

}

}

cout<<ans<<"\n";

return 0;

}