Name: Sucheta Pal UID: 20BCS7901

Q1. Candies

```
Code:
```

```
#include <bits/stdc++.h>
using namespace std;
// INT HERE MEANS LONG LONG
#define int long long
#define endl '\n'
bool solve(int *arr, int n, int ele)
  int s = 0, e = n-1;
  while(s \le e)
     int mid = (s+e)/2;
     if(arr[mid] == ele)
     return true;
     else if(arr[mid]<ele)
       s = mid+1;
     else
     e = mid-1;
  return false;
int32_t main(){
  ios base::sync with stdio(false);
  cin.tie(NULL);
  int t;
  cin >> t;
  while (t--){
     int n;
     cin>>n;
     int * candies = new int[n];
     for(int i = 0; i < n; i++)
```

```
{
    cin>>candies[i];
}

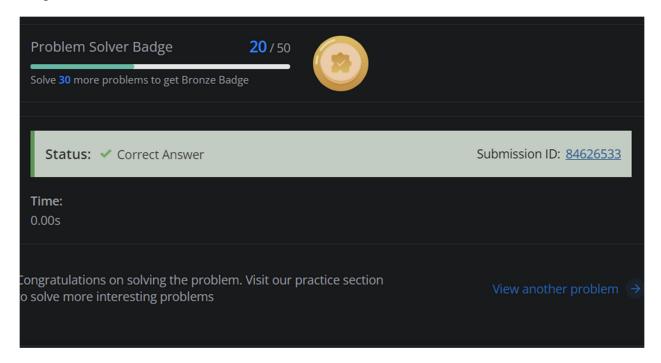
int desired_level = *max_element(candies,candies+n)-*min_element(candies,candies+n);
bool f = false;

for(int i = 0; i<n; i++)
{
    if(candies[i] == desired_level)
    {
        cout<< desired_level<<endl;
        f = true;
        break;
    }
}

if(!f)
    cout<< "NO"<<endl;
}

return 0;
}
</pre>
```

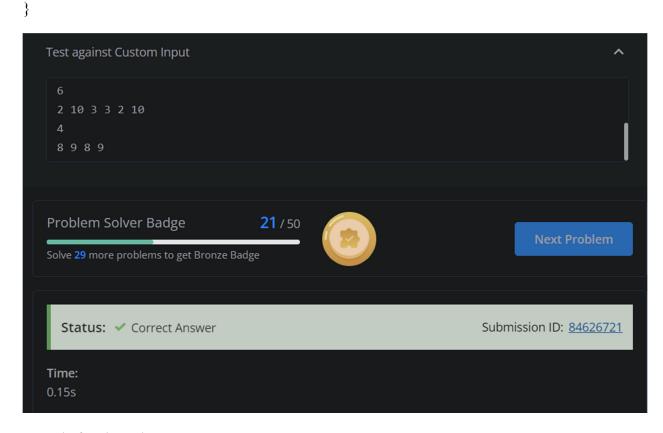
Output:



Q2. Chef and patients

```
Code:
#include <bits/stdc++.h>
using namespace std;
void solve() {
  int n; cin >> n;
  vector<pair<int, int>> a(n);
  int mx = 0;
  for(int i=0; i<n; i++) {
     a[i].second = i;
    cin >> a[i].first;
    mx = max(mx, a[i].first);
  }
  for(int i=0; i<n; i++) a[i].first = mx - a[i].first;
  sort(a.begin(), a.end());
  vector<int> ans(n);
  for(int i=0; i < n; i++) ans[a[i].second] = i+1;
  for(int& e : ans) cout << e << " "; cout << endl;
}
int main() {
  int ti; cin >> ti;
```

```
while(ti--) solve();
return 0;
```



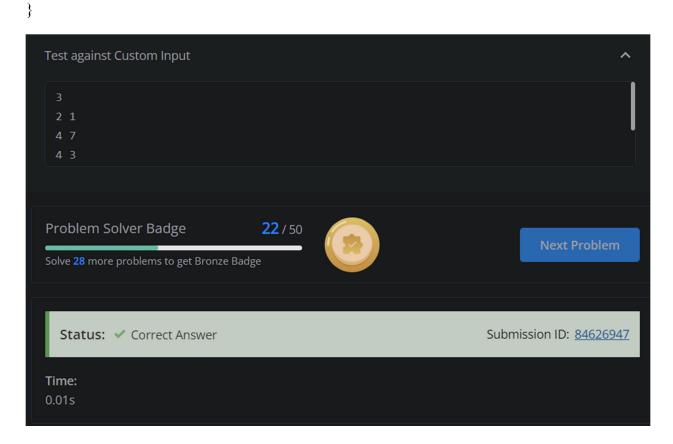
Q3. Chef and employment test

```
Code:
#include <bits/stdc++.h>
using namespace std;

int main() {
    int t;
    cin>>t;
    while(t--){
    int n,k;
```

cin>>n>>k;

```
int arr[n];
for(int i=0;i<n;i++){
    cin>>arr[i];
}
sort(arr,arr+n);
cout<<arr[(n+k)/2]<<endl;
}
return 0;</pre>
```



Q4. chef and party

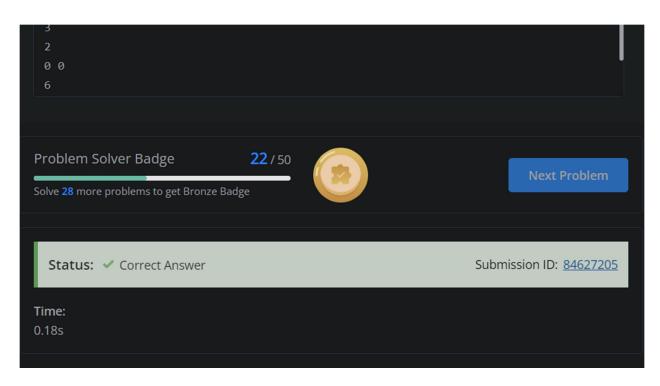
Code:

Output:

#include <bits/stdc++.h>

```
int main() {
       // your code goes here
        int t;
        cin>>t;
        while(t--){
          int n;
          cin>>n;
          int arr[n];
          for(int i=0;i<n;i++){
             int a;
             cin>>arr[i];
          int count=0;
          sort(arr,arr+n);
           for(int i=0;i<n;i++){
             if(arr[i] == 0 \parallel arr[i] \leq= count)\{
                count++;
          cout<<count<<endl;</pre>
        return 0;
}
```

using namespace std;



Q5. Frog sort

```
Code:
#include <iostream>
#include<vector>
#include<algorithm>
#include<math.h>
using namespace std;

bool compare(vector<int> &v1,vector<int> &v2){
  return v1[1]<v2[1];
}

int main() {
  int t;
  cin>>t;
```

```
while(t--){
      int n;
      cin>>n;
      vector<vector<int>>> v(n,vector<int>(3));
      for(int i=0;i<n;i++)
 {
    v[i][0]=i;
     cin>>v[i][1];
 }
for(int j=0;j<n;j++){
      cin>>v[j][2];
}
sort(v.begin(),v.end(),compare);
// for(int i=0;i< n;i++)
// {
      cout \!\!<\!\! v[i][0] \!\!<\!\! " " \!\!<\!\! v[i][1] \!\!<\!\! " " \!\!<\!\! v[i][2] \!\!<\!\! endl;
// }
int ans=0;
for(int i=1;i< n;i++){}
  if(v[i][0]\!\!<\!\!=\!\!v[i\text{-}1][0])\{
```

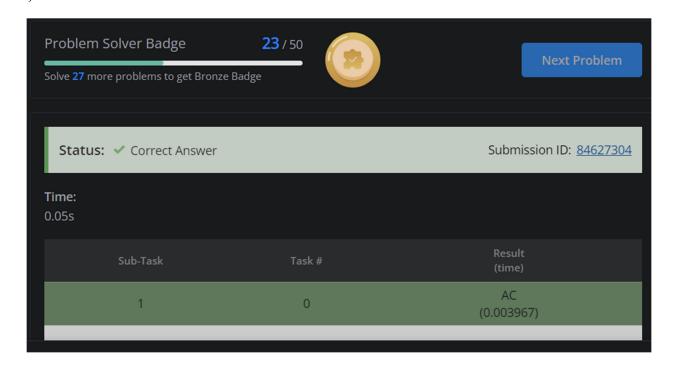
```
int inc=v[i-1][0]-v[i][0]+1;
  int count=ceil((inc*1.0)/(v[i][2]*1.0));

// cout<<i<<" "<<count<<" ";
  ans+=count;
  v[i][0]+=v[i][2]*count;

// cout<<<v[i][0]<<endl;

}

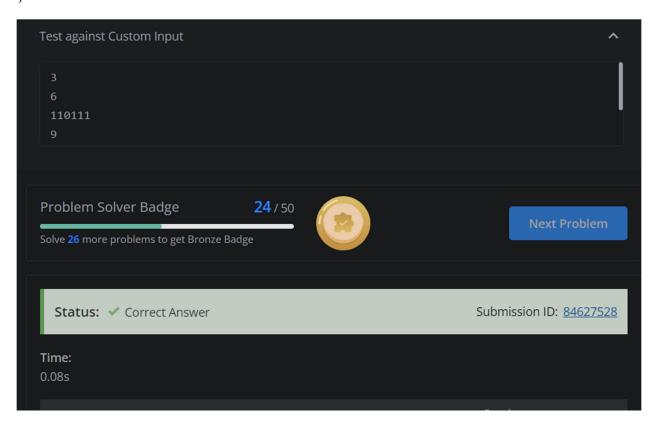
cout<<ans<<endl;
}
return 0;
}</pre>
```



Q6. Flip Sorting

Code:

```
#include <bits/stdc++.h>
using namespace std;
#define ll long long int
int main()
{
11 t;
cin>>t;
while(t--)
{
int n;
cin>>n;
string s;
cin>>s;
string k=s;
sort(k.begin(),k.end());
if(k==s)
cout<<"0"<<endl;
else
   int count=0;
   for(int i=0;i<n-1;i++)
   \{ if(s[i]!=s[i+1]) \}
   count++;
   }
   cout<<count<<endl;</pre>
```



Q7. Chef and sorting

#include <bits/stdc++.h>

using namespace std;

typedef long long ll;

#define maxN 1010

#define maxK 60

```
#define mod 100000007
int a[maxN];
int main(){
int tc, n, i, j, k, l = (1 << 30) - 1;
scanf("%d", &tc);
while (tc--) {
scanf("%d", &n);
for (i = 0; i < n; i++) {
scanf("%d", &a[i]);
j = 0, k = 1;
for (i = 1; i < n; i++) {
if (a[i - 1] < a[i]) {
k++;
}
else if (a[i - 1] > a[i]) {
j++;
}
printf("%d\n", min(j, k));
if (k \le j) {
printf("%d %d %d\n", 3, n, 1);
for (i = 0; i < n; i++) {
a[i] \stackrel{\wedge}{=} 1;
}
```

```
\label{eq:for_continuous_section} $$ for (i=1; i < n; i++) $$ if (a[i-1] > a[i]) $$ printf("%d %d %d\n", 2, i+1, a[i-1] - a[i]); $$ $$ $$ $$ return 0; $$ $$ $$
```

