

Day 3-Week 2- 7th April

1. Minimum difference pair

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

// } Driver Code Ends
class Solution{
    public:
        int minimum_difference(vector<int>nums)
        {
            sort(nums.begin(),nums.end());
            int min=INT_MAX;
            for(int i=0;i<nums.size()-1;i++)
            {
                if(nums[i+1]-nums[i]<min)
                {
                    min=nums[i+1]-nums[i];
                }
            }
        }
    };
};
```

```
    }  
}  
return min;  
}  
  
};  
  
// { Driver Code Starts.  
int main(){  
    int tc;  
    cin >> tc;  
    while(tc--){  
        int n;  
        cin >> n;  
        vector<int>nums(n);  
        for(int i = 0; i < nums.size(); i++)cin >> nums[i];  
        Solution ob;  
        int ans = ob.minimum_difference(nums);  
        cout << ans <<"\n";  
    }
```

```
        return 0;
    } // } Driver Code Ends
```

2. Rotate Array

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

```
// } Driver Code Ends
class Solution{
public:
```

//Function to rotate an array by d elements in counter-clockwise direction.

```
void rotateArr(int arr[], int d, int n)
{
    d=d%n;
    reverse(arr,arr+d);
    reverse(arr+d,arr+n);
    reverse(arr,arr+n);
```

```
}  
};
```

```
// { Driver Code Starts.
```

```
int main() {
```

```
    int t;
```

```
    //taking testcases
```

```
    cin >> t;
```

```
    while(t--){
```

```
        int n, d;
```

```
        //input n and d
```

```
        cin >> n >> d;
```

```
        int arr[n];
```

```
        //inserting elements in the array
```

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

```
        cin >> arr[i];
    }
    Solution ob;
    //calling rotateArr() function
    ob.rotateArr(arr, d,n);

    //printing the elements of the array
    for(int i =0;i<n;i++){
        cout << arr[i] << " ";
    }
    cout << endl;
}
return 0;
} // } Driver Code Ends
```

3. First and last occurrences of x

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

```
// } Driver Code Ends  
vector<int> find(int arr[], int n , int x )  
{  
    int first=-1;  
    int last=-1;  
    for(int i=0;i<n;i++)  
    {  
        if(arr[i]==x)  
        {  
            first=i;  
            break;  
        }  
    }  
    for(int i=n-1;i>=0;i--)  
    {  
        if(arr[i]==x)  
        {  
            last=i;  
            break;  
        }  
    }
```

```
    }  
    vector<int>v;  
    v.push_back(first);  
    v.push_back(last);  
    return v;  
}
```

```
// { Driver Code Starts.
```

```
int main()  
{  
    int t;  
    cin>>t;  
    while(t--)  
    {  
        int n,x;  
        cin>>n>>x;  
        int arr[n],i;  
        for(i=0;i<n;i++)  
            cin>>arr[i];
```

```
vector<int> ans;  
ans=find(arr,n,x);  
cout<<ans[0]<<" "<<ans[1]<<endl;  
}  
return 0;  
}
```

```
// } Driver Code Ends
```

4. Find Missing And Repeating

```
// { Driver Code Starts
```

```
#include <bits/stdc++.h>
```

```
using namespace std;
```

```
// } Driver Code Ends
```

```
class Solution{
```

```
public:
```

```
int *findTwoElement(int *arr, int n) {
```



```
int *ans=new int[2];
```

```
int freq[n]={0};
```

```
int missing=0;
```

```
int repeating=0;
```

```
for(int i=0;i<n;i++){  
    freq[arr[i]-1]++;  
}
```

```
for(int i=0;i<n;i++){  
    if(freq[i]==0){  
        missing=i+1;  
    }  
    if(freq[i]==2){  
        repeating=i+1;  
    }  
}
```

```
    ans[0]=repeating;
    ans[1]=missing;

    return ans;
}
};
```

```
// { Driver Code Starts.
```

```
int main() {
    int t;
    cin >> t;
    while (t--) {
        int n;
        cin >> n;
        int a[n];
        for (int i = 0; i < n; i++) {
            cin >> a[i];
        }
    }
}
```

```
Solution ob;  
auto ans = ob.findTwoElement(a, n);  
cout << ans[0] << " " << ans[1] << "\n";  
}  
return 0;  
} // } Driver Code Ends
```