Unique Subarrays (G23)

A contiguous subarray is defined as unique if all the integers contained within it occur exactly once. There is a unique weight associated with each of the subarray. Unique weight for any subarray equals it's length if it's unique, 0 otherwise. Your task is to calculate the sum of unique weights of all the contiguous subarrays contained within a given array.

Input : arr[] = {1, 2, 3}

Output : 10

{1, 2, 3} is a subarray of length 3 with distinct elements. Total length of length three = 3.

{1, 2}, {2, 3} are 2 subarray of length 2 with distinct elements. Total length of lengths two = 2 + 2 = 4

{1}, {2}, {3} are 3 subarrays of length 1 with distinct element. Total lengths of length one = 1 + 1 + 1 = 3

Sum of lengths = 3 + 4 + 3 = 10

Input : arr[] = {1, 2, 1}

Output : 7

Input : arr[] = {1, 2, 3, 4} Output : 20

Input Format

- First line of the input contains an integer T, denoting the number of testcases
- T lines follow, where first line of each testcase contains an integer **N** denoting array size.
- Last line of each testcase then contains N single space separated integers

Constraints

- $1 \leq T,N \leq 10^5$
- $0 \leq A_i \leq 10^9$
- Summation of N for all T does not exceed 10^5

Output Format

1. Print the summation of unique weights of all the subarrays for each testcase in a separate line.

Sample Input 0

```
1
5
1 2 3 4 5
```

Sample Output 0

```
35
```

Explanation 0

Since, all integers are distinct within any contiguous subarray, therefore the unique weight will be the summation of lengths of all subarrays. Hence, this sums upto (5*1) + (4*2) + (3*3) + (2*4) + (1*5) = 35

Sample Input 1

```
1
8
1 2 3 4 4 6 7 7
```

Sample Output 1

```
31
```

Contest ends in an hour

Submissions: 0
Max Score: 20
Difficulty: Hard

Rate This Challenge:

More

Current Buffer (saved locally, editable) C++

```cpp
#include <cmath>
#include <cstdio>
#include <vector>
#include <iostream>
#include <algorithm>
#include<bits/stdc++.h>
using namespace std;
int subarray(int a[],int n)
{
    unordered_set<int> s;
    int j=0,r=0;
    for(int i=0;i<n;i++)
    {
        while(j<n && s.find(a[j])==s.end())
        {
            s.insert(a[j]);
            j++;

        }
        r+=((j-i)*(j-i+1))/2;
        s.erase(a[i]);
    }
    return r;
}

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

```
36          cout<<subarray(a,n)<<"\n";
37      }
38      return 0;
39  }
40
```

Line: 34 Col: 29

Upload Code as File Test against custom input Run Code Submit Code

Testcase 0 ✔ Testcase 1 ✔

Congratulations, you passed the sample test case.

Click the **Submit Code** button to run your code against all the test cases.

Input (stdin)

```
1
5
1 2 3 4 5
```

Your Output (stdout)

```
35
```

Expected Output

```
35
```