https://course.acciojob.com/idle?question=1ca1c62e-19c2-4ef6-9f62-0fb652b756dc

- MEDIUM
- Max Score: 30 Points
-

Longest Subsequence With Difference One

You are given an array `nums` of size `N`.

Find the length of the longest subsequence of array `nums` such that the absolute difference between every adjacent element in the subsequence is one.

Input Format

First line contains the size of array `N`.

Second line contains n-spaced integers represeting array `nums`.

Output Format

Print an integer denoting the length of the longest subsequence of array `nums` such that the difference between adjacent elements is one.

Example 1

Input

```
5
4 2 1 5 6
```

Output

```
3
```

Explanation

The longest subsequence satisfying the condition is {4, 5, 6}.

Example 2

Input

```
6
-2 2 -1 1 0 -1
```

Output

```
4
```

Explanation

The longest subsequence satisfying the condition is {-2, -1, 0, -1}. There is another possible subsequence of the same length, i.e., {2, 1, 0, -1}. The length of both the subsequences is 4.

Constraints

1 <= N <= 10^5

-10^9 <= nums[i] <= 10^9

Topic Tags

- **Hashing**

My code

```
// n java
import java.util.*;
import java.lang.*;
import java.io.*;
```

```
public class Main
{
        static int longestSubsequence(int n, int a[])
    {
        // code here

        HashMap<Integer,Integer> hm=new HashMap<>();
        //hm stores the curr num as index and value as the len of the
subseq in whcih key is the last num
        int max=1;//len of the longest subseq having diff 1
        for(int i=0;i<n;i++){
            //will check if any of find 1 and find 2 exist in hm
            int find1=a[i]-1;
            int find2=a[i]+1;
            if(hm.containsKey(find1) && hm.containsKey(find2)){
                int len=Math.max(hm.get(find1),hm.get(find2));
                //curr num will join with the subseq which is longer
                hm.put(a[i],len+1);
            }
            else if(hm.containsKey(find1)){
                hm.put(a[i],hm.get(find1)+1);//curr num will join with the
subseq of which find1 is last num
            }
            else if(hm.containsKey(find2)){
                hm.put(a[i],hm.get(find2)+1);//curr num will join with the
subseq of which find2 is last num
            }
            else{
                hm.put(a[i],1);
            }
```

```
            max=Math.max(max,hm.get(a[i]));
        }
        return max;
    }
        public static void main (String[] args) throws
java.lang.Exception
        {
                //your code here
                Scanner sc=new Scanner(System.in);
                int n=sc.nextInt();
                int a[]=new int[n];
                for(int i=0;i<n;i++){
                        a[i]=sc.nextInt();
                }
                System.out.println(longestSubsequence(n,  a));
        }
}
```