LIS:

Givne an array of number I have to find the largest common increasing Subsequence in both sides of the array;

Input :

10 1 2 3 4 5 4 3 2 1 10

19

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

Output:

9

9

Code:

```
///10534 Wavio Sequence
#include<bits/stdc++.h>
using namespace std;
#define MX 10000
#define ll long long
#define inf 999999999
ll sequence1[MX + 5];
ll sequence2[MX + 5];
ll arr2[MX + 5];
ll L1[MX + 5];
ll L2[MX + 5];
ll I1[MX + 5];
ll I2[MX + 5];
ll n;

ll LIS()
{
    I1[0] = -inf;
    for(int i=1; i<=n; i++)
        I1[i]= inf;
    ll lo,hi,mid;
    ll length = 0;
    for(int i=0; i<n; i++)
    {
        lo = 0,hi = length;
        while(lo<=hi)
        {
            mid = (lo + hi)/2;
            if(I1[mid]<sequence1[i])
                lo =mid + 1;
            else
                hi = mid - 1;
        }
        I1[lo] = sequence1[i];
        L1[i] = lo;
        if(length<lo)
            length = lo;
    }
    return length;
}

ll LIS2()
{
    I2[0] = -inf;
    for(int i=1; i<=n; i++)
        I2[i]= inf;
    ll lo,hi,mid;
    ll length = 0;
    for(int i=0; i<n; i++)
    {
        lo = 0,hi = length;
        while(lo<=hi)
        {
            mid = (lo + hi)/2;
            if(I2[mid]<sequence2[i])
                lo =mid + 1;
            else
                hi = mid - 1;
        }
        I2[lo] = sequence2[i];
        L2[i] = lo;
        if(length<lo)
            length = lo;
    }
    return length;
}
int main()
{
    while(scanf("%lld",&n) == 1)
    {
        for(int i=0; i<n; i++)
        {
            scanf("%lld",&sequence1[i]);
        }
        int k=0;
        for(int i=n-1; i>=0; i--)
        {
            sequence2[k++] = sequence1[i];
        }
        ll ans1 = LIS();
        ll ans2 = LIS2();
        ll cnt1 = 0;
        ll cnt2 = 0;

        for(int i=0; i<n/2; i++)
            swap(L2[i],L2[n-i-1]);
        ll c = 0;
        for(ll i=0; i<n; i++)
        {
            c = max(c,L1[i]);
            L1[i] = c;
        }
        c = 0;
        for(ll i=n-1; i>=0; i--)
        {
            c = max(c,L2[i]);
            L2[i] = c;
        }
for(ll i=0; i<n; i++)
        {
            if(L1[i] == L2[i])
            {
                cnt1 =max(cnt1,L1[i]);
            }
        }
        cout<<cnt1 * 2 - 1<<endl;
    }
    return 0;
}
```