

Secuencias

Autores del problema: Félix Moreno Peñarrubia y Maria Prat Colomer

Sea d_0, \dots, d_{n-2} la secuencia definida por $d_i = d_{i+1} - d_i$. Notemos que una operación lo que hace es cambiar exactamente uno de los números de la secuencia d , y para que la secuencia a sea VomitA (unimodal) la secuencia d tiene que haber un índice $0 \leq j \leq n-1$ tal que $d_i \leq 0$ para todo $i < j$ y $d_i \geq 0$ para todo $i \geq j$. Podemos probar todos los índices j , contar cuántos valores de d hay que cambiar para que se satisfaga la condición (el número de índices $i < j$ tales que $d_i > 0$ más el número de índices $i \geq j$ tal que $d_i < 0$) y retornar menor número que obtengamos. Esto implementado directamente es $O(n^2)$, podemos mejorarlo a $O(n)$ guardando en dos arrays la frecuencia acumulativa de los índices i tales que $d_i > 0$ y $d_i < 0$.

Solución en C++

```
#include<bits/stdc++.h>
using namespace std;
typedef vector<int> vi;
int main() {
    int T;
    cin >> T;
    while(T--) {
        int n;
        cin >> n;
        vi a(n);
        for(int i=0; i < n; ++i) cin >> a[i];
        int pos = 0;
        int neg = 0;
        vi sp(n);
        vi sn(n);
        for(int i=0; i < n-1; ++i) {
            int d= a[i+1]-a[i];
            sp[i+1] = sp[i];
            sn[i+1] = sn[i];
            if(d > 0) {
                pos++;
                sp[i+1]++;
            }
            else if(d < 0) {
                neg++;
                sn[i+1]++;
            }
        }
        int ans = n+1;
        for(int i=0; i < n; ++i) {
            ans = min(ans, min(sn[i]+pos-sp[i], sp[i]+neg-sn[i]));
        }
        cout << ans << endl;
    }
}
```

Autor de la solución: Félix Moreno Peñarrubia (Miembro del comité organizador)