slope trick

算法简介

一种维护凸包的技巧。

算法实现

假定一个函数 $f(x)$ 是连续函数，可以被划分为多条直线，直线的斜率单调递增/递减，则可以用最终直线 $g(x)$ 和可重集 S 来表示 $f(x)$□

例如 $f(x)=|x-1|$ 可以表示为 $(x-1,\{1,1\})$□$f(x)=|2x-4|+|3x|$ 可以表示为 $(5x-4,\{0,0,0,0,0,0,2,2,2,2\})$□

不难发现，设 $f(x)=(k_1x+b_1,S_1),g(x)=(k_2x+b_2,S_2)$ 且 $f(x),g(x)$ 的斜率都是单调性相同时，有

$$ f(x)+g(x)=((k_1+k_2)x+b_1+b_2,S1\cup S_2) $$

算法例题

例题一

CF713C

题意

给定序列 A□每次操作可以选中一个位置 i□令 $a_i\gets a_i+1$ 或 $a_i\gets a_i-1$□问使得序列严格单增的最小操作次数。

题解

令 $a_i\gets a_i-i$□于是问题转化求使得序列非递减的最小操作次数。

设 $f_i(x)$ 表示使得 $a[1\sim i]$ 非递减且 $a_i\le x$ 的最小操作次数□$g_i(x)$ 表示使得 $a[1\sim i]$ 非递减且 $a_i=x$ 的最小操作次数。

于是有 $g_i(x)=f_{i-1}(x)+|x-a_i|$□由于 f_i,f_{i-1} 斜率最大值为 0□g_i 斜率最大值为 1，因此为了得到 f_i□需要删除 g_i 的 S 的最大元素。

设 g_i 的 S 中的最大值为 k□则该操作相当于把 g_i 的最后一段改成水平线，则

$$ f_i(x)=(y=g_i(k),S-\{k\})=(y=f_{i-1}(k)+|k-a_i|,S-\{k\}) $$

不难发现 k 一定位于 $f_{i-1}(x)$ 的最后水平线那一段，于是只需要维护所有 $f_i(x)$ 的最后一段的截

距即可，同时这也是 $f_i(x)$ 的最小值。

```
int main()
{
    int n=read_int();
    priority_queue<int> q;
    LL ans=0;
    _for(i,0,n){
        int a=read_int()-i;
        q.push(a);
        if(a<q.top()){
            ans+=q.top()-a;
            q.pop();
            q.push(a);
        }
    }
    enter(ans);
    return 0;
}
```