

知识精炼（二）



主讲人：邓哲也

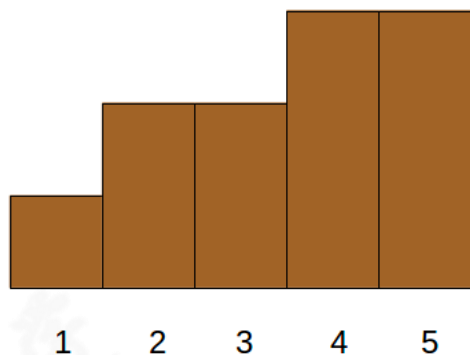


Codeforces 484E. Sign on Fence

有 n 个篱笆排成一列，第 i 个篱笆的高度为 $h[i]$.

每次询问给定 l, r, w ，问区间 $[l, r]$ 中连续的 w 个篱笆的最小高度的最大值。

$n, m \leq 100000, h[i] \leq 10^9$



input output

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

```
2
3
1
```

Codeforces 484E. Sign on Fence

注意到答案是可以二分的。

二分最小高度 h_{\min} ，只要在区间 $[1, r]$ 中检查是否有连续 w 个数都大于等于 h_{\min} 。

我们把 $h[i] \geq h_{\min}$ 的位置填上 1， $h[i] < h_{\min}$ 的位置填上 0。

问题就变成了检查区间 $[1, r]$ 中是否有连续 w 个 1。

这个问题可以用线段树来做。

Codeforces 484E. Sign on Fence

维护区间内连续 1 的最大长度即可。

$lmax[x]$ 表示 x 代表的区间从左端点开始连续的 1 的个数

$rmax[x]$ 表示 x 代表的区间从右端点开始连续的 1 的个数

$f[x]$ 表示 x 代表的区间中连续的 1 的个数的最大值

Codeforces 484E. Sign on Fence

```
void update(int l, int r, int x) {  
    int mid = (l + r) >> 1;  
    lmax[x] = lmax[ls];  
    if (lmax[ls] == mid - 1 + 1) lmax[x] += lmax[rs];  
    rmax[x] = rmax[rs];  
    if (rmax[rs] == r - mid) rmax[x] += rmax[ls];  
    f[x] = max(max(f[ls], f[rs]), lmax[rs] + rmax[ls]);  
}
```

Codeforces 484E. Sign on Fence

但是上面做的前提是在固定 h_{\min} 的情况下。

因为总共有 n 个 $h[i]$ ，所以 h_{\min} 最多可能有 n 种取值。

最多可能有 n 颗线段树。

使用可持久化线段树！

Codeforces 484E. Sign on Fence

我们对 h 从大到小排序。

依次往线段树中插入 1，得到一棵新的线段树。

二分 h_{\min} 的时候，在 h_{\min} 对应的那颗线段树上做查询即可。

时间复杂度： $O(n \log n + m \log^2 n)$

空间复杂度： $O(n \log n)$

下节课再见