这道题经历了3个版本，第一个版本阶梯高度是1,第二个版本高度差变成了k,被嘲笑太简单后，改成了现在的版本。

先说第二个版本的做法吧，有一个非常漂亮的O(n)的做法：

因为要最长，所以如果h(i)为阶梯的起点，那么把h(i)加工成高度1肯定是最优的。

dp(i):表示从以h(i)为阶梯的起点，所能加工出的最长阶梯的最右端的位置，那么有转移为：

Int tt=1;

for (int i=1, p=1; i<=n; i++, p+=k) {

while (p>h[i]) {

dp[tt ++] = i;

p -= k;

}

}

for (int i=tt; i<=n; i++) dp[tt] = n;

然后泥for一遍版本2的就做完了。

现在考虑询问区间[pl ,pr]的最大值：

很明显，对于pl <= i <= pr, 如果dp[i]>=pr, 那么dp[i-1]必定>=pr,

于是，我们可以用线段树维护两个值，(max{dp[i]-i+1}, max{dp[i]}) (注意这两个是分开维护的)

对于询问的区间pl, pr, 我们分成两个部分去询问：

1. 先在线段树上二分找到一个最左的dp[i]>=pr的位置l;
2. 然后在[pl, l-1]上询问一个最大的dp[i]-i+1即可;

然后答案即为两部分取最大即可，所以每次复杂度为log(n)的。