登山

题目就是修改最少的下标使得形成一个等差数列，枚举公差 d，那么 d 最多枚举到 $O(\frac{m}{n})$。

接下来就是考虑计算最少修改多少位置，对于 h_i，只有当 $h_1 = h_i - (i-1)d$ 时这个位置不用修改，这样我们可以对于每个 i 求出 $h_1 = i$ 时有多少位置不用改，取最多的地方作为 h_1 即可，时间复杂度 $O(\frac{m}{n} \times n) = O(m)$。

水属性魔法使

对于一条边 (u_i, v_i, d_i) 上的隔板，其限制为要么两边的高度一样，要么都小于等于 d_i，前者是容易处理的。

对于后者，设从 u_i 到 v_i 最大边权最小的路径上的最大边是 l，那么两者的高度也一定要 $\le l$，那么这就是最小瓶颈路，只有最小生成树上的边有用。

考虑 Kruskal 的过程，用 f_u 维护出 u 所在连通块的方案数，合并两个连通块的时候，两边是独立的直接乘起来就行，复杂度 $O(m \log m)$。

跳石头

$O(n^4)$ 的做法就是记录 i, j 枚举 k, l 暴力转移，而一个事实是我们可以每次强制先动左脚再动右脚，记录 $g_{0/1,i,j}$ 表示当前该动左脚/右脚，两个脚分别在哪块石头上的最小代价，转移就只需要枚举一个脚移动到了哪里，时间是 $O(n^3)$。

我们的转移形如

$g_{0,i,j} = \min\limits_{k<j} g_{1,i,k} + (j-k)^2$，这是一个经典的斜率优化柿子，可以维护凸壳或者李超树，时间复杂度 $O(n^2)$ 或者 $O(n^2 \log n)$。

花园魔法

考虑固定 l 的时候，随着 r 的移动怎么维护答案。

一个想法是如果把 x 变成 y 就连一条从 x 到 y 的边，但是这样如果有 1变成 2，2 变成 1 就完了。

我们考虑每次新建一个虚点 p 作为 x,y 的父亲，然后 p 的颜色为 y 的颜色，这样子就不会出问题了，那么当 r 移动到 n 之后我们就得到了一颗森林，考虑怎么处理查询。颜色段数只需要知道相邻两个位置的颜色是否相同，而对于两个位置 $i,i+1$，第一次变得相同的时刻就是它们的 LCA，用树状数组维护一个后缀加即可。

现在不固定 l，从大到小移动 l，考虑在前面新建一个操作有什么影响，其实只需要新建一个节点 p 作为 x,y 的父亲，然后 y 原本的父亲作为 p 的父亲即可，可以发现只有 x 的祖先关系会更改，重构 $x-1,x,x+1$ 之间的贡献即可。

综上我们需要加叶子求 LCA 和树状数组，前者可以用倍增动态维护。

时间复杂度 $O(n \log n)$。