暴雨

考虑 DP，f[i][j][p][0/1]表示前 i 列，最大值是 j，且要求在后面存在一个高度至少为 j 的土地，已经铲平了 p 块，当前积水体积为奇数/偶数的方案数。由于至多会去掉 K 块土地，所以对于任意一个 i，j 的值只有 K+1 种。

前缀后缀 DP 都做出来之后，枚举最终最大值所在列合并答案即可，注意处理多个最大值的情况。

时间复杂度 O(NK^2)

AVL 树

按照中序遍历和先序遍历贪心均可。

每一次检查当前点是否可以加入最终的树中，从当前点向上，每次遇到自己是左子树时，根据目前的情况计算右子树至少需要留下多少的点。

计算得到一个点留下整棵树至少多大，如果不超过 K，则显然可以。

由于读入的树是 AVL 树，树高为$\log N$。

时间复杂度 O($N\, log\, N$)

挖掘机

显然，最终的挖的方案一定可以按照深度从小到大的顺序操作，那么在操作深度 D 时，小于 D 的所有深度相应区间的土地一定都消失了。由于一次操作只可能影响同一个深度的土地，所以我们可以将每一行看成一个独立的问题，最后将答案相加即可。

剩下的问题就是如何处理一行。一个显然的贪心是区间中最靠前的一块土地肯定要被覆盖，最优的方式就是将其作为长度为 K 的区间的第一个。不断的做这样的操作即可得到最少步数。我们预处理出每个位置开始的倍增数组，即可在$log\, W$的时间内回答询问。

时间复杂度 O（$H(W+Q)\, log\, W$）

游戏

由于每个长度 K 对应的总方案数是确定的，所以我们就是要计算三种情况的数量。

我们考虑计算$X \leq Y$的方案数，考虑对串 A#B 构建后缀数组，对于每一个属于 A 的后缀，排在它后面的属于 B 的后缀和它产生的贡献是 1-min(SA,SB)长度的答案均+1，SA，SB 分别表示两个后缀的长度。考虑暴力，从后往前，遇到 B 后缀时，将 1-SB 的贡献均+1，遇到 A 后缀时，将 1-SA 的贡献都加进 1-SA 的答案中去。

对于 A 后缀，在它前面的 B 后缀的贡献是 1-lcp(SA,SB)的答案+1。考虑类似的暴力，从前往后，在遇到后缀 B 时，将 1-K（或者 1-SB，是一样的）的贡献+1，遇到每一个 height 都将大于 height 的贡献清零，遇到一个后缀 A，就将 1-heightA 的贡献加进答案。

于是我们需要维护一个结构，能够将 1-x 的贡献+1，1-x 的贡献加进答案，x-K 的贡献清零。考虑这样一个数组，每一项是$(1 \quad delta \quad answer)$，delta 表示当前位置贡献和，answer 表示当前位置累计答案，则对于增加贡献的操作，相当于对区间乘上一个矩阵$\begin{pmatrix}1 & 1 & 0\\0 & 1 & 0\\0 & 0 & 1\end{pmatrix}$，加进答案的操作相当于对区间乘上一个矩阵$\begin{pmatrix}1 & 0 & 0\\0 & 1 & 1\\0 & 0 & 1\end{pmatrix}$，清零操作相当于对区间乘上一个矩阵$\begin{pmatrix}1 & 0 & 0\\0 & 0 & 0\\0 & 0 & 1\end{pmatrix}$。于是我们只需要一个结构能够维护不停的给区间乘上一个矩阵即可，这可以用线段树来实现。

由于$\begin{pmatrix}1 & a & b\\0 & c & d\\0 & 0 & 1\end{pmatrix}\begin{pmatrix}1 & e & f\\0 & g & h\\0 & 0 & 1\end{pmatrix} = \begin{pmatrix}1 & e+ag & f+ah+b\\0 & cg & ch+d\\0 & 0 & 1\end{pmatrix}$，所以只需要维护 4 个值即可。

$X \geq Y$的情况同理。我们用这两种方案数减去总的方案数就得到了相等的方案数。然后我们就能计算得到小于和大于的方案数。

时间复杂度 O（$(N+M)\,log(N+M)$）