

Subway

记一下当前已经用了哪些地铁，然后跑 Dijkstra。当然也可以理解为建分层图。注意不需要真的把最后用的是哪个地铁记状态里。 $O(2^k km \log)$ 。

Distanced

先把正的绑成联通块，那么我们就是可以选择用负的连起来一些或者断掉一些。相当于要跑一个最小生成树状物，所以可以直接sort完一条一条边考虑。我们还会发现有的边会改变：并起来两块之后断掉的代价会增大（两个断掉合二为一），断掉一块之后如果还想连起来两头代价也会增大。但是代价仍然是越来越大的，所以kruskal还是对的（只要不会发现一条代价更小的边就行）。

Color

首先我们可以直接DP，我们的操作大概是一开始初始化每个f[i]为要么只有一个1要么全是1，然后对于每个子树的f把它取反（ $x \rightarrow \text{sum}-x$ ）然后相乘。

我们发现这样搞完DP数组大部分位置基本都一样。记一下哪些位置比较不一样，我们需要支持的操作就是把这样的数组取反（ $x \rightarrow \text{sum}-x$ ）和对乘。对乘我们实际上可以启发式，所以只需要小的往大的修改，小的里没有的就是乘常数。可以在线段树合并的时候打tag，也可以直接发现可以维护原始值，然后所有原始值上多一个 $kx+b$ 的tag，然后拿个map或者unordered_map维护原始值。

Grid

分治。假设当前区间是权值 $[l, r]$ ，对于所有 $> r$ 的格子算出来它开头往后越来越大的路径数， $< l$ 的算出来它结尾的路径数，然后 $> r$ 和 $< l$ 也可以连起来计入答案。