

映射 (xiz)

直接 *hash* 就可以了...

优美一点的做法是 KMP，记每个字符的前一个字符距离自己的位置为 w_i ，如果两个串的 w 数组相等，则可以匹配。预处理出 w 就可以用 KMP 了

路径 (path)

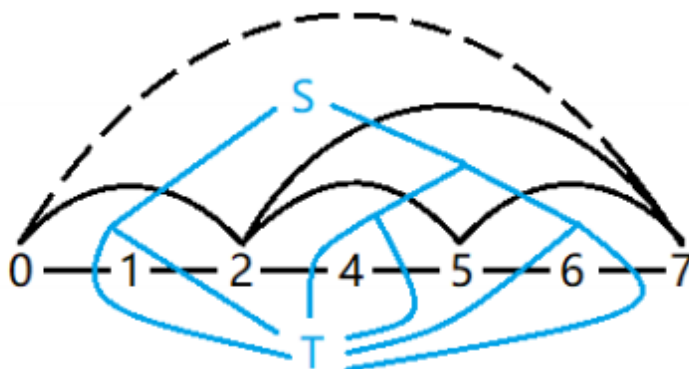
解法一

暴力枚举经过了哪些颜色，之后路径其实就已经确定了，判一下即可。

时间复杂度 $O(2^k(n + m))$ ，其中 k 为不同的颜色个数。期望得分 40 分。

解法二

可以发现给定的图是一个平面图，并且其对偶图主要部分形成了一棵树。平面图上的一个环对应偶图上的一个割。因此我们可以额外加入一条从结点 0 到结点 n 的边，把这条边下方的区域视为源点 S ，外侧的区域视为汇点 T ，那么图中的最短路就转化为了最小环，即对应 S 与 T 之间的最小割。做的时候可以先把无用的点和边删去，那么原图的最底层一定就是一条链了。例如，样例二的图大概长这样：



考虑对颜色的限制怎么放到对偶图上。假设对偶图上 x 和父亲之间的边对应的是原图的边 (u, v) ，如果 x 被划分到 T 割的话，也就意味着跳过了 $u + 1..v - 1$ 这些结点。如果对偶图上另一个区域 y 对应的边也跳过了某个与 $u + 1..v - 1$ 的其中一个结点同色的结点，那么 y 也必须同时划分到 T 割。具体建图时可以给每种颜色添加一个辅助点，然后从 x 向 $u + 1..v - 1$ 的颜色的辅助点连容量为正无穷的双向边。不过这样边数会变成平方级别的，实际上 x 在 T 割时其子树肯定都在 T 割，因此只需要向未被更底层的边覆盖的结点连就可以了。

时间复杂度 $O(\max flow(O(n), O(m)))$ ，期望得分 100 分。

序列 (seq)

考虑转化问题.

对所有数取 \log_{10} , 于是乘法变成了加法, 并且最高位可以直接计算. 于是问题转化为区间排序并维护区间和.

一个显然的想法是对每个点开一个值域线段树, 再在全局用一个大线段树维护每个值域线段树之间的关系. 排序操作相当于将两端未完全覆盖的值域线段树分裂, 同时将中间的值域线段树合并. 询问时只要全局线段树中维护一下和就可以了.

显然只有 $O(n)$ 次分裂与合并.