

J 因为字典序最小所以贪心取最小，对于第  $i$  位只可能是三种答案  $s1[i]$ ,  $s2[i]$ , 和小写字母里面非  $s1[i]$ ,  $s2[i]$  的最小字母，如果存在某些位  $s1[j] == s2[j]$ ，那么这一位不管选哪一种与答案串的汉明距离都不会变化，因此当然直接取最小的 'a'，然后对于剩下的位从左到右贪心，依次尝试取最小，如果能在这一位取最小就取即可，判定条件就是后面的位能否补偿回来

K 对于左端点排序，从小到大扫，对于某个左端点，当然是贪心取已经扫过的右端点最远的  $K$  个，那么贪心取就好了，可以用很多种数据结构维护，如优先队列，set，map

L 树分治一下，然后统计每个根的贡献，首先 BFS 出所有的从这个根出去的路径，然后每个子树的扫，统计非当前子树里面与当前子树里面枚举到的这个路径的状态互补的有多少个，可以 map 实现（状态指算术分解之后指数模 3 的状态压后的状态）