

# 咒语

算法: 由于最终所求字符串  $S$  的各个位之间互不影响, 所以我们可以逐位确定每一位取 0 还是取 1。因为我们需要相异度之和最小, 且在这一前提下 0 的个数尽量多, 所以  $S$  的第  $i$  位取 0, 当且仅当输入的  $n$  个字符串中, 第  $i$  位为 0 的字符串个数大于等于第  $i$  位为 1 的字符串个数。

时间复杂度:  $O(n*L)$

# 神光

算法：首先我们注意到，当  $R$  和  $G$  的大小超过了  $N$  时， $L$  的最小值就是 1，因此，我们只需要考虑  $R$  和  $G$  小于  $N$  的情况，于是  $R, G$  的规模就降到了 2000 以内。

显然要采用二分答案的方法。那么问题转化为，判断给定的  $L$  能否摧毁所有法坛，我们采用动态规划方法。首先将法坛的位置按照从小到大进行排序。令  $dp[i][j]$  表示，在用了  $i$  次红光， $j$  次绿光的情况下，最多从第一座法坛开始，一直摧毁到第几座法坛。那么状态转移方程即为  $dp[i][j] = \max(P[dp[i-1][j] + 1], Q[dp[i][j-1] + 1])$ 。其中  $P[k]$  表示使用一次红光，能从第  $k$  座法坛向右（正向为右）连续摧毁到第几座， $Q[k]$  表示使用一次绿光，能从第  $k$  座法坛向右连续摧毁到第几座。 $P$  和  $Q$  数组可以通过预处理得到。最终，我们只要判断  $dp[R][G]$  的值是否为  $N$  即可。

时间复杂度： $O(N^2)$

期望得分：100

# 迷宫

算法：题目要求次短路，我们可以先用 Dijkstra 算法求出最短路。设最短路长为  $s1$ ，与结点  $n$  相连的最短边的长度为  $c$ ，设  $s2=s1+2*c$ ，则  $s2$  就是次短路长度的上界。接下来我们只要进行 DFS 深搜即可，在搜索过程中利用上界  $s2$  进行剪枝，并不断更新  $s2$ ，就可以在题目规定的时间内得到结果。

期望得分：100

