

1 相遇

测试点 1。和样例一样。

测试点 2。答案是 0。

测试点 3。终点在起点的左侧，只能倒着走，答案是 N 。

测试点 4。向右走一格，答案是 1。

测试点 5。每次移动有 3 种方式，由于答案不超过 8，爆搜即可，搜索次数不超过 3^8 。

测试点 1 ~ 10。将爆搜改为 BFS 即可，注意对状态的处理，由于限制了 N 和 K 都在 0 到 10^5 之间，因此在 BFS 时候可以只保留一个区间 $[L, R]$ 内的状态，比如 $L = -10^6$ ， $R = 10^6$ 。当然，存在更紧的上下界 L 、 R ，取 $\pm 10^6$ 已经足够了。

2 秘密邮件

算法一：当 $n=2$ 时，只需要判断 A 和 B 是否一样然后输出 0 或者 1 即可。期望得分 10 分。

算法二：把 A 串的每个状态看成一个节点，共 $n!$ 个节点。然后进行 BFS，计算答案。期望得分 20 分。

算法三：对于 40% 的数据，每种大写字母最多只出现一次。我们考虑一个例子：当 A 串为 ACBD， B 串为 CDBA 的时候，我们把 A 串的字母顺次标记为 1,2,3,4。对于 B 串的每个字母我们将它标记为它在 A 串中的位置，比如字母 C 在 A 串中是第 2 个字母，因此把 C 标记为 2。这样标记以后我们得到的 B 串就为 2,4,3,1。

我们考虑这个问题的另一种形式：每次可以交换 B 串相邻的两个字母问最少交换多少次可以变成 A 串，易知这个问题和原问题答案相等。我们可以发现每次交换 B 串相邻的两个字母，要么导致标号序列逆序对减一，要么导致标号序列逆序对加一，而最终的 A 序列是不含逆序对的。因此我们只要每次都让 B 的逆序对减一即可得到最优策略，因此 B 序列中的逆序对对数就是答案。期望得分 40 分。

算法四：当只出现两种字母的时候，我们不妨设其中一种为字母 x ，我们记 x 在 A 串的位置依次为 a_1, a_2, \dots, a_m ；在 B 串中的位置依次为 b_1, b_2, \dots, b_m 。根据贪心，我们需要让每个 a_i 在结束时对应到位置 b_i 。因此答案为 $\sum_{i=1}^m |a_i - b_i|$ 。期望得分 30 分，结合算法二后期望得分 70 分。

算法五：我们只需要在算法三的基础上进行改进即可，同样是将 A 串顺次标记为 $1, 2, 3, 4, \dots, n$ 。接下来我们对每种字母分别在 B 串中按照它们在 B 串中的出现位置从小到大进行标号。如 $A = \text{CDDGF}$, $B = \text{DGDGF}$ 。则我们将 B 串标记为 $2, 4, 3, 5, 1$ 。接下来只要统计 B 串标号的逆序对数即可，使用 $O(n \log n)$ 复杂度的算法统计，期望得分 100 分。

3 小乔

测试点 1 ~ 2。简单计算就可以知道答案，期望得分 10 分。

测试点 1 ~ 6。我们可以把坐标系划分成 $m \times r$ 块，对于一个扇形，将它覆盖到的每一块的覆盖次数加 1，最后统计有哪些块的覆盖次数不小于 k ，计算面积即可。复杂度 $O(nmr)$ ，期望得分 30 分。

测试点 7 ~ 12。这部分测试点所有扇形的半径都相同，这样我们可以把扇形“拉直”成线段，覆盖的区域就成了数轴上的区间。那么问题就转化成了经典的区间覆盖问题。我们把每个区间拆成左端点和右端点两个位置，然后把所有位置按照坐标排序。从左到右扫，遇到左端点则将覆盖次数加 1，遇到右端点则减 1。当覆盖次数不小于 k 时，把这个区间的面积加入答案。复杂度 $O(n \log n)$ ，结合前面的算法可以得到 60 分。

测试点 1 ~ 20。我们把每个扇形的左右段对应的角度都提出来排序，将圆周划分成 $O(n)$ 个部分。我们考虑一个区间，取出所有覆盖了这个区间的扇形。在这个区间内，半径较大的扇形一定完全覆盖了半径较小的扇形。那么这里被覆盖了至少 k 次的面积就是半径第 k 大的扇形在这个区间内的面积。

如果我们按照角度顺序依次处理每个区间，那么我们就需要一种数据结构来帮助我们快速查找第 k 大的元素，还需要支持插入和删除一个数。一个可行的，也是标程采用的数据结构是平衡树。当然这不是唯一的做法。另外，还有 20 分的数据 $k = 1$ ，可以直接用堆维护。使用平衡树的算法复杂度 $O(n \log n)$ ，期望得分 100 分。