

CSP2023模拟赛

题目名称	AK神	桌球游戏	烽火石	弟中弟
题目类型	传统型	传统型	传统型	传统型
可执行文件名	ak	game	tower	loser
输入文件名	ak.in	game.in	tower.in	loser.in
输出文件名	ak.out	gamek.out	tower.out	loser.out
每个测试点时限	1.0 秒	2.0 秒	1.0 秒	1.0 秒
内存限	512 MiB	512 MiB	512 MiB	512 MiB
子任务/测试点数目	5	4	10	5
是否等分	否	否	是	否

提交源文件程序名

对于C++语言	ak.cpp	game.cpp	tower.cpp	loser.cpp
---------	--------	----------	-----------	-----------

编译选项

对于C++语言	-lm -O2 -std=c++17
---------	--------------------

注意事项(请仔细阅读)

1. 文件名（程序名和输入输出文件名）必须使用英文小写。
2. C/C++中函数main()的返回类型必须是int，程序正常结束时返回值必须是0。
3. 选手提交的程序代码文件请在**个人目录下以及子文件夹内各放一份**。
4. 若无特殊说明，结果的比较方式为全文比较（过滤行末空格及文末回车）。
5. 选手提交的程序源文件必须不大于100KB。
6. 程序可使用的栈空间内存限制于题目的内存限制一直。
7. 使用std::deque等STL容器时，请注意其内存空间消耗。
8. 评测时采用的机器配置为 AMD Ryzen 7 5800H with Radeon Graphics，内存16GiB。上述时限以此配置为准。
9. 评测在Windows 10下进行，使用LemonLine进行评测。

AK 神 (ak)

【题目描述】

Alice 和 Bob 正在玩一个双人游戏。

他们面前有 n 堆石头排成一排，其中第 i 堆石头有 a_i 颗石头。

两个人轮流操作，每次可以选择**连续的 k 堆石头**，将它们拿走，剩下的石头合并成一个新的石头序列，直到剩下唯一一堆石头时结束。

Alice 先手，他的目标是使两人拿走的石头数量总和最小，Bob 的目标则是使这个值最大。

若两人都使用最优策略，那么他们最后拿走的石头数量总和为多少？

【输入格式】

第一行两个正整数 n, k ，表示初始石头序列长度以及每次拿走的石头堆数量。

第二行 n 个正整数，第 i 个正整数 a_i 表示第 i 堆石头的数量。

【输出格式】

输出一个整数表示最后拿走的石头数量总和。

【输入输出样例1】

ak.in	ak.out
5 2 3 5 3 2 1	11

【输入输出样例1说明】

初始序列为 $\{3, 5, 3, 2, 1\}$ 。

Alice 的最优策略是拿走最右侧的 $\{2, 1\}$ ，序列变为 $\{3, 5, 3\}$ 。然后 Bob 拿走最左侧的 $\{3, 5\}$ ，序列变为 $\{3\}$ 。两人总计拿走 11 颗石头。

【数据规模与约定】

本题采用捆绑测试。对于所有的测试数据，保证 $1 \leq n, k \leq 10^6, n \equiv 1(\text{ mod } k), 1 \leq a_i \leq 10^9$ 。

子任务编号	$n \leq$	$k \leq$	$a_i \leq$	分值
1	10	3	10^9	20
2	10^6	1	10^9	10
3	10^6	2	10^9	30
4	10^6	10^6	2	30
5	10^6	10^6	10^9	10

桌球游戏 (game)

【题目描述】

Hikari 酱喜欢玩桌球游戏。桌球游戏在一个 $R \times C$ 的矩形桌面上进行，其中 R, C 为正整数，桌面的四个角的顶点各有一个洞。桌球可以认为是桌面上的一个动点。当桌球从某点沿一个方向射出后，桌球将一直沿该方向运动，直到桌球碰到矩形桌面的边界：

1. 当桌球碰到了矩形桌面的四个角的顶点之一时，球将落入洞中，运动停止；
2. 当桌球碰到了矩形桌面的四个边界之一，但不是顶点，球将沿该边反弹，反弹时满足反射角等于入射角。

Hikari 总是喜欢将球从桌面的左下角沿右上 45° 方向发射，然后观察球的碰撞，直到球落入洞中。她认为碰撞次数越多的桌面游戏体验越高，因此定义该 $R \times C$ 桌面的价值为发射后球的碰撞次数，记为 $f(R, C)$ 。例如， $f(5, 2) = 5$ 。

Hikari 的问题是，对于所有长、宽分别不超过 M, N 的矩形桌面，它们的价值分别是多少。为了方便起见，你只需求出所有这些矩形桌面的价值之和，即 $\sum_{R=1}^M \sum_{C=1}^N f(R, C)$

【输入格式】

文件的第一行包含一个整数 T 表示测试用例个数。接下来 T 个测试用例的描述如下：每个测试用例一行，包含两个整数 M 和 N ，用一个空格隔开。

【输出格式】

对于每个测试用例输出一行答案 mod 1000000007。

【输入输出样例1】

game.in	game.out
4	52
5 4	30
4 4	709032898
1000 981	283206800
5872502 5382176	

【数据规模与约定】

本题采用捆绑测试。

子任务编号	$N, M \leq$	$T \leq$	分值
1	10	100	30
2	10^3	10000	30
3	10^5	100	20
4	10^7	10000	20

烽火石 (tower)

【题目描述】

击败第二天王之后，小 A 遇见了一个与众不同的敌人——擅长谋略，手里有着大量士兵了第三天王——小 B。

小 A 决定主动出击，利用自己手中士兵数量的优势冲击小 B 驻守的整条防线。为此，小 A 特地从北方魔法学院购来了新的发明——烽火石。

烽火石是一种特殊的石头，可以起到类似烽火台的信息交流功能。小 A 需要驻守的防线上一共有 N 座城市，第 i 座城市内的烽火石颜色为 b_i 。 b_i 是一个不大于 K 的正整数。从城市 i 传递消息到城市 j 需要 $|i - j|$ 个单位时间。然而，并不是任意两个烽火石之间都可以传递消息的。我们可以用一个 $K \times K$ 的 01 矩阵 S 来表示不同颜色的烽火石传递消息的可行性。 $S_{i,j} = 1$ 表示颜色为 i 的烽火石可以将消息传递给颜色为 j 的烽火石， $S_{i,j} = 0$ 则表示不能。

不保证 $S_{i,j} = S_{j,i}$ ，也不保证 $S_{i,i} = 1$ 。

你需要求出，将一条信息从第 1 座城市传递到第 N 座城市所需要的最短时间

【输入格式】

第一行输入两个用空格隔开的正整数 N, K ，含义见题面。

输入的第二行包括 N 个用空格隔开的正整数 b_1, b_2, \dots, b_N 。

接下来的 K 行描述矩阵 S ，每一行包括一个长度为 K 的由 01 组成的字符串。第 i 行的第 j 个字符即为 $S_{i,j}$ 。

【输出格式】

输出一行一个正整数，表示将一条信息从第一座城市传递到第 N 座城市所需要的最短时间，如果不能传达，请输出 -1。

【输入输出样例1】

tower.in	tower.out
5 4 6 1 4 2 3 4 1010 0001 0110 0100	6

【输入输出样例1说明】

最优的信息传输序列为 $1 \rightarrow 4 \rightarrow 3 \rightarrow 5$ ，花费的总时间为 $1 - 4 + 4 - 3 + 3 - 5 = 6$ 。

【数据规模与约定】

- 对于前 20% 的测试数据，保证 $N \leq 10$ 。
- 对于前 40% 的测试数据，保证 $N \leq 1000$ 。

对于全部的数据， $1 \leq N \leq 5 \times 10^4, 1 \leq K \leq 50$ 。

弟中弟 (loser)

【题目描述】

给定一棵 n 个点的树，其中每个点上有一个左括号或右括号。

有 q 次操作，每次操作给定一个参数 op ，以及一条有向简单路径 $x \rightarrow y$ 。

- 若 $op = 1$ ，则将路径 $x \rightarrow y$ 上的所有左括号变为右括号，右括号变为左括号。
- 若 $op = 2$ ，则记 S 为路径 $x \rightarrow y$ 上经过的所有点上面的括号依次拼接，输出 S 的最长合法括号子序列长度。

【输入格式】

第一行两个正整数 n, q ，表示树的大小和操作次数。

接下来 $n - 1$ 行，每行两个正整数 u, v ，描述一条树边 (u, v) 。

再往下一行 n 个 01 整数，第 i 个整数表示点 i 上的括号类型，其中 0 表示左括号，1 表示右括号。

接下来 q 行，每行三个正整数 op, x, y ，描述操作参数和有向简单路径 $x \rightarrow y$ 。

【输出格式】

对于每个 $op = 2$ 的操作输出一行一个正整数，表示答案。

【输入输出样例1】

loser.in	loser.out
7 3 1 2 2 3 3 4 4 5 2 6 4 7 0 1 0 1 1 0 1 2 1 5 1 6 7 2 5 2	4 2

【数据规模与约定】

本题采用捆绑测试。对于所有的测试数据，保证 $1 \leq n, q \leq 10^5, 1 \leq u, v, x, y \leq n, op \in \{1, 2\}$ 。

子任务编号	\mathfrak{L}	\mathfrak{R}	\mathfrak{K}	特殊性质	分值
1	500	500	$\{1, 2\}$		12
2	4000	4000	$\{1, 2\}$		28
3	10^5	10^5	$\{2\}$		24

子任务编号	$n \leq$	$q \leq$	$op \in$	特殊性质	分值
4	10^5	10^5	$\{1, 2\}$	A	24
5	10^5	10^5	$\{1, 2\}$		12

- 特殊性质 A : 保证给出的第 i 条树边为 $(i, i + 1)$ 。