

1、纪念品

(souvenir.cpp/c/pas)

【问题描述】

小伟突然获得一种超能力，他知道未来 T 天 N 种纪念品每天的价格。某个纪念品的价格是指购买一个该纪念品所需的金币数量，以及卖出一个该纪念品换回的金币数量。

每天，小伟可以进行以下两种交易**无限次**：

- 任选一个纪念品，若手上有足够金币，以当日价格购买该纪念品；
- 卖出持有的任意一个纪念品，以当日价格换回金币。

每天卖出纪念品换回的金币可以**立即**用于购买纪念品，当日购买的纪念品也可以**当日卖出**换回金币。当然，一直持有纪念品也是可以的。

T 天之后，小伟的超能力消失。因此他一定会在第 T 天卖出**所有**纪念品换回金币。小伟现在有 M 枚金币，他想要在超能力消失后拥有尽可能多的金币。

【输入格式】

输入文件名为souvenir.in。

第一行包含三个正整数 T, N, M ，相邻两数之间以一个空格分开，分别代表未来天数 T ，纪念品数量 N ，小伟现在拥有的金币数量 M 。

接下来 T 行，每行包含 N 个正整数，相邻两数之间以一个空格分隔。第 i 行的 N 个正整数分别为 $P_{i,1}, P_{i,2}, \dots, P_{i,N}$ ，其中 $P_{i,j}$ 表示第 i 天第 j 种纪念品的价格。

【输出格式】

输出文件名为souvenir.out。

输出仅一行，包含一个正整数，表示小伟在超能力消失后最多能拥有的金币数量。

【输入输出样例 1】

souvenir.in	souvenir.out
6 1 100 50 20 25 20 25 50	305

见选手目录下的 souvenir/souvenir1.in和 souvenir/souvenir1.ans。

【输入输出样例 1 说明】

最佳策略是：

第二天花光所有 100 枚金币买入 5 个纪念品 1；

第三天卖出 5 个纪念品 1，获得金币 125 枚；

第四天买入 6 个纪念品 1，剩余 5 枚金币；

第六天必须卖出所有纪念品换回 300 枚金币，第四天剩余 5 枚金币,共 305 枚金币。

超能力消失后，小伟最多拥有 305 枚金币。

【输入输出样例 2】

souvenir.in	souvenir.out
3 3 100 10 20 15 15 17 13 15 25 16	217

见选手目录下的 souvenir/souvenir2.in和souvenir/souvenir2.ans。

【输入输出样例 2 说明】最佳

策略是：

第一天花光所有金币买入 10 个纪念品 1；

第二天卖出全部纪念品 1 得到 150 枚金币并买入 8 个纪念品 2 和 1 个纪念品 3，剩余 1 枚金币；

第三天必须卖出所有纪念品换回 216 枚金币，第二天剩余1 枚金币，共217 枚金币。

超能力消失后，小伟最多拥有 217 枚金币。

【输入输出样例 3】

见选手目录下的 souvenir/souvenir3.in和souvenir/souvenir3.ans。

【数据规模与约定】

对于 10% 的数据， $T = 1$ 。

对于 30% 的数据， $T \leq 4$, $N \leq 4$, $M \leq 100$ ，所有价格 $10 \leq P_{i,j} \leq 100$ 。

另有 15% 的数据， $T \leq 100$, $N = 1$ 。

另有 15% 的数据， $T = 2$, $N \leq 100$ 。

对于 100% 的数据， $T \leq 100$, $N \leq 100$, $M \leq 10^3$ ，所有价格 $1 \leq P_{i,j} \leq 10^4$ ，数据保证任意时刻，小明手上的金币数不可能超过 10^4 。

2、摆渡车

(bus.cpp/c/pas)

【问题描述】

有 n 名同学要乘坐摆渡车从人大附中前往人民大学，第 i 位同学在第 t_i 分钟去等车。只有一辆摆渡车在工作，但摆渡车容量可以视为无限大。摆渡车从人大附中出发、把车上的同学送到人民大学、再回到人大附中（去接其他同学），这样往返一趟总共花费 m 分钟（同学上下车时间忽略不计）。摆渡车要将所有同学都送到人民大学。

凯凯很好奇，如果他能任意安排摆渡车出发的时间，那么这些同学的等车时间之和最小为多少呢？

注意：摆渡车回到人大附中后可以即刻出发。

【输入格式】

输入文件名为 bus.in。

第一行包含两个正整数 n, m ，以一个空格分开，分别代表等车人数和摆渡车往返一趟的时间。

第二行包含 n 个正整数，相邻两数之间以一个空格分隔，第 i 个非负整数 t_i 代表第 i 个同学到达车站的时刻。

【输出格式】

输出文件名为 bus.out。

输出一行，一个整数，表示所有同学等车时间之和的最小值（单位：分钟）。

【输入输出样例 1】

bus.in	bus.out
5 1 3 4 4 3 5	0

见选手目录下的 bus/bus1.in和bus/bus1.ans。

【输入输出样例 1 说明】

同学 1 和同学 4 在第 3 分钟开始等车，等待 0 分钟，在第 3 分钟乘坐摆渡车出发。摆渡车在第 4 分钟回到人大附中。

同学 2 和同学 3 在第 4 分钟开始等车，等待 0 分钟，在第 4 分钟乘坐摆渡车出发。摆渡车在第 5 分钟回到人大附中。

同学 5 在第 5 分钟开始等车，等待 0 分钟，在第 5 分钟乘坐摆渡车出发。自此所有同学都被送到人民大学。总等待时间为 0。

【输入输出样例 2】

bus.in	bus.out
5 5 11 13 1 5 5	4

见选手目录下的 bus/bus2.in和bus/bus2.ans。

【输入输出样例 2 说明】

同学 3 在第 1 分钟开始等车，等待 0 分钟，在第 1 分钟乘坐摆渡车出发。摆渡车在第 6 分钟回到人大附中。

同学 4 和同学 5 在第 5 分钟开始等车，等待 1 分钟，在第 6 分钟乘坐摆渡车出发。摆渡车在第 11 分钟回到人大附中。

同学 1 在第 11 分钟开始等车，等待 2 分钟；同学 2 在第 13 分钟开始等车，等待 0 分钟。他/她们在第 13 分钟乘坐摆渡车出发。自此所有同学都被送到人民大学。总等待时间为 4。可以证明，没有总等待时间小于 4 的方案。

【输入输出样例 3】

见选手目录下的 `bus/bus3.in` 和 `bus/bus3.ans`。

【数据规模与约定】

对于 10% 的数据， $n \leq 10, m = 1, 0 \leq t_i \leq 100$ 。

对于 30% 的数据， $n \leq 20, m \leq 2, 0 \leq t_i \leq 100$ 。

对于 50% 的数据， $n \leq 500, m \leq 100, 0 \leq t_i \leq 10^4$ 。

另有 20% 的数据， $n \leq 500, m \leq 10, 0 \leq t_i \leq 4 \times 10^6$ 。

对于 100% 的数据， $n \leq 500, m \leq 100, 0 \leq t_i \leq 4 \times 10^6$ 。

3、棋盘

(chess.cpp/c/pas)

【问题描述】

有一个 $m \times m$ 的棋盘，棋盘上每一个格子可能是红色、黄色或没有任何颜色的。你现在要从棋盘的最左上角走到棋盘的最右下角。

任何一个时刻，你所站在的位置必须是有颜色的（不能是无色的），你只能向上、下、左、右四个方向前进。当你从一个格子走向另一个格子时，如果两个格子的颜色相同，那你不需要花费金币；如果不同，则你需要花费 1 个金币。

另外，你可以花费 2 个金币施展魔法让下一个无色格子暂时变为你指定的颜色。但这个魔法不能连续使用，而且这个魔法的持续时间很短，也就是说，如果你使用了这个魔法，走到了这个暂时有颜色的格子上，你就不能继续使用魔法；只有当你离开这个位置，走到一个本来就有颜色的格子上的时候，你才能继续使用这个魔法，而当你离开了这个位置（施展魔法使得变为有颜色的格子）时，这个格子恢复为无色。

现在你要从棋盘的最左上角，走到棋盘的最右下角，求花费的最少金币是多少？

【输入格式】

输入文件名为 chess.in。

数据的第一行包含两个正整数 m, n ，以一个空格分开，分别代表棋盘的大小，棋盘上有颜色的格子的数量。

接下来的 n 行，每行三个正整数 x, y, c ，分别表示坐标为 (x, y) 的格子有颜色 c 。

其中 $c=1$ 代表黄色， $c=0$ 代表红色。相邻两个数之间用一个空格隔开。棋盘左上角的坐标为 $(1, 1)$ ，右下角的坐标为 (m, m) 。

棋盘上其余的格子都是无色。保证棋盘的左上角，也就是 $(1, 1)$ 一定是有颜色的。

【输出格式】

输出文件名为 chess.out。

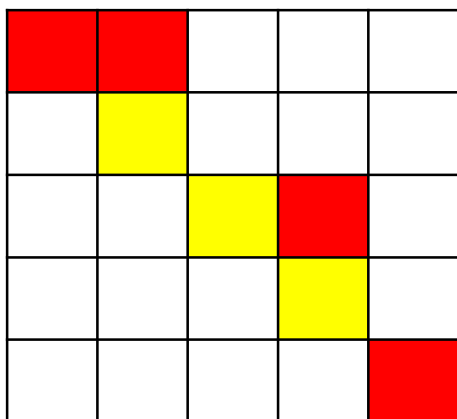
输出一行，一个整数，表示花费的金币的最小值，如果无法到达，输出-1。

【输入输出样例 1】

chess.in	chess.out
5 7 1 1 0 1 2 0 2 2 1 3 3 1 3 4 0 4 4 1 5 5 0	8

见选手目录下的 chess/chess1.in和chess/chess1.ans。

【输入输出样例 1 说明】



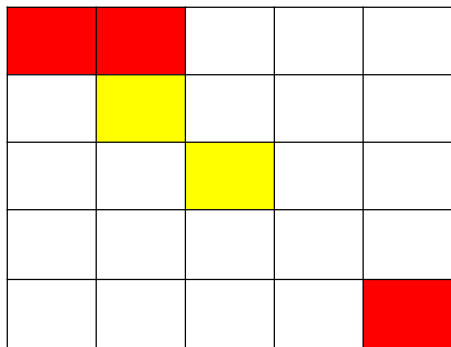
从 (1, 1) 开始, 走到 (1, 2) 不花费金币
 从 (1, 2) 向下走到 (2, 2) 花费 1 枚金币
 从 (2, 2) 施展魔法, 将 (2, 3) 变为黄色, 花费 2 枚金币
 从 (2, 2) 走到 (2, 3) 不花费金币
 从 (2, 3) 走到 (3, 3) 不花费金币
 从 (3, 3) 走到 (3, 4) 花费 1 枚金币
 从 (3, 4) 走到 (4, 4) 花费 1 枚金币
 从 (4, 4) 施展魔法, 将 (4, 5) 变为黄色, 花费 2 枚金币,
 从 (4, 4) 走到 (4, 5) 不花费金币
 从 (4, 5) 走到 (5, 5) 花费 1 枚金币
 共花费 8 枚金币

【输入输出样例 2】

chess.in	chess.out
5 5 1 1 0 1 2 0 2 2 1 3 3 1 5 5 0	-1

见选手目录下的 chess/chess2.in 和 chess/chess2.ans。

【输入输出样例 2 说明】



从 (1, 1) 走到 (1, 2), 不花费金币
从 (1, 2) 走到 (2, 2), 花费 1 金币
施展魔法将 (2, 3) 变为黄色, 并从 (2, 2) 走到 (2, 3) 花费 2 金币
从 (2, 3) 走到 (3, 3) 不花费金币
从 (3, 3) 只能施展魔法到达 (3, 2), (2, 3), (3, 4), (4, 3)
而从以上四点均无法到达 (5, 5), 故无法到达终点, 输出 -1

【输入输出样例 3】

见选手目录下的 chess/chess3.in 和 chess/chess3.ans。

【数据规模与约定】

对于 30% 的数据, $1 \leq m \leq 5, 1 \leq n \leq 10$ 。

对于 60% 的数据, $1 \leq m \leq 20, 1 \leq n \leq 200$ 。

对于 100% 的数据, $1 \leq m \leq 100, 1 \leq n \leq 1,000$ 。

4、海港 (port)

【问题描述】

小K是一个海港的海关工作人员，每天都有许多船只到达海港，船上通常有很多来自不同国家的乘客。

小K对这些到达海港的船只非常感兴趣，他按照时间记录下了到达海港的每一艘船只情况；对于第 i 艘到达的船，他记录了这艘船到达的时间 t_i (单位：秒)，船上的乘客数量 k_i ，以及每名乘客的国籍 $X_{i,1}, X_{i,2}, \dots, X_{i,k_i}$ 。

小K统计了 n 艘船的信息，希望你帮忙计算出以每一艘船到达时间为止的24小时 (24小时=86400秒) 内所有乘船到达的乘客来自多少个不同的国家。

形式化地讲，你需要计算 n 条信息。对于输出的第 i 条信息，你需要统计满足 $t_i - 86400 < t_p \leq t_i$ 的船只 p ，在所有的 $X_{p,j}$ 中，总共有多少个不同的数。

【输入格式】

从文件 `port.in` 中读入数据。

第一行输入一个正整数 n ，表示小K统计了 n 艘船的信息。

接下来 n 行，每行描述一艘船的信息：前两个整数 t_i 和 k_i 分别表示这艘船到达海港的时间和船上的乘客数量，接下来 k_i 个整数 $X_{i,j}$ 表示船上乘客的国籍。

保证输入的 t_i 是递增的，单位是秒；表示从小K第一次上班开始计时，这艘船在第 t_i 秒到达海港。

保证 $1 \leq n \leq 10^5$ ， $k_i \geq 1$ ， $\sum k_i \leq 3 \times 10^5$ ， $1 \leq x_{i,j} \leq 10^5$ ， $1 \leq t_{i-1} < t_i \leq 10^9$ 。

其中 $\sum k_i$ 表示所有的 k_i 的和， $\sum k_i = k_1 + k_2 + \dots + k_n$ 。

【输出格式】

输出到文件 `port.out` 中。

输出 n 行，第 i 行输出一个整数表示第 i 艘船到达后的统计信息

【样例1输入】

```
3
14 4 1 2 2
2 2 2 3
10 1 3
```

【样例1说明】

第一艘船在第1秒到达海港，最近24小时到达的船是第一艘船，共有4个乘客，分别是来自国家4,1,2,2，共来自3个不同的国家；

【样例1输出】

```
3
4
4
```


第二艘船在第2秒到达海港，最近24小时到达的船是第一艘船和第二艘船，共有 $4 + 2 = 6$ 个乘客，分别是来自国家4,1,2,2,2,3，共来自4个不同的国家；

第三艘船在第10秒到达海港，最近24小时到达的船是第一艘船、第二艘船和第三艘船，共有 $4 + 2 + 1 = 7$ 个乘客，分别是来自国家4,1,2,2,2,3,3，共来自4个不同的国家。

【样例2输入】

4
1 4 1 2 2 3
3 2 2 3
86401 2 3 4
86402 1 5

【样例2输出】

3
3
3
4

【样例2说明】

第一艘船在第1秒到达海港，最近24小时到达的船是第一艘船，共有4个乘客，分别是来自国家1,2,2,3，共来自3个不同的国家；

第二艘船在第3秒到达海港，最近24小时到达的船是第一艘船和第二艘船，共有 $4 + 2 = 6$ 个乘客，分别是来自国家1,2,2,3,2,3，共来自3个不同的国家；

第三艘船在第86401秒到达海港，最近24小时到达的船是第二艘船和第三艘船，共有 $2 + 2 = 4$ 个乘客，分别是来自国家2,3,3,4，共来自3个不同的国家；

第四艘船在第86403秒到达海港，最近24小时到达的船是第二艘船、第三艘船和第四艘船，共有 $2 + 2 + 1 = 5$ 个乘客，分别是来自国家2,3,3,4,5，共来自4个不同的国家。

【子任务】

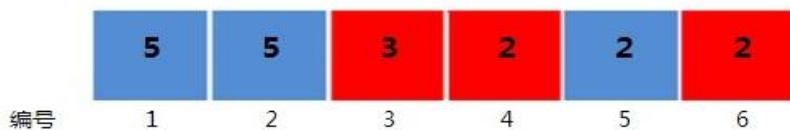
- 对于 10% 的测试点， $n = 1, \sum k_i \leq 10, 1 \leq x_{i,j} \leq 10, 1 \leq t_i \leq 10$ ；
- 对于 20% 的测试点， $1 \leq n \leq 10, \sum k_i \leq 100, 1 \leq x_{i,j} \leq 100, 1 \leq t_i \leq 32767$ ；
- 对于 40% 的测试点， $1 \leq n \leq 100, \sum k_i \leq 100, 1 \leq x_{i,j} \leq 100, 1 \leq t_i \leq 86400$ ；
- 对于 70% 的测试点， $1 \leq n \leq 1000, \sum k_i \leq 3000, 1 \leq x_{i,j} \leq 1000, 1 \leq t_i \leq 10^9$ ；
- 对于 100% 的测试点， $1 \leq n \leq 10^5, \sum k_i \leq 3 \times 10^5, 1 \leq x_{i,j} \leq 10^5, 1 \leq t_i \leq 10^9$ 。

3. 求和

(sum.cpp/c/pas)

【问题描述】

一条狭长的纸带被均匀划分出了 n 个格子，格子编号从 1 到 n 。每个格子上都染了一种颜色 $color_i$ （用 $[1, m]$ 其中的一个整数表示），并且写了一个数字 $number_i$ 。



定义一种特殊的三元组： (x, y, z) ，其中 x, y, z 都代表纸带上格子的编号，这里的三元组要求满足以下两个条件：

1. x, y, z 都是整数, $x < y < z, y - x = z - y$
2. $color_x = color_z$

满足上述条件的三元组的分数规定为 $(x + z) * (number_x + number_z)$ 。整个纸带的分数规定为所有满足条件的三元组的分数的和。这个分数可能会很大，你只要输出整个纸带的分数除以 10,007 所得的余数即可。

【输入格式】

输入文件名为 sum.in。

第一行是用一个空格隔开的两个正整数 n 和 m ， n 代表纸带上格子的个数， m 代表纸带上第二行有 n 个用空格隔开的正整数，第 i 个数字 $number_i$ 代表纸带上编号为 i 的格子上面写的数字。

第三行有 n 个用空格隔开的正整数，第 i 个数字 $color_i$ 代表纸带上编号为 i 的格子染的颜色。

【输出格式】

输出文件名为 sum.out。

共一行，一个整数，表示所求的纸带分数除以 10,007 所得的余数。

【输入输出样例 1】

sum.in	sum.out
6 2	82
5 5 3 2 2 2	
2 2 1 1 2 1	

见选手目录下的 sum/sum1.in 和 sum/sum1.ans。

【输入输出样例 1 说明】

纸带如题目描述中的图所示。

所有满足条件的三元组为： $(1, 3, 5), (4, 5, 6)$ 。

【输入输出样例 2】

sum.in	sum.out
15 4	1388
5 10 8 2 2 2 9 9 7 7 5 6 4 2 4	
2 2 3 3 4 3 3 2 4 4 4 4 1 1 1	

见选手目录下的 sum/sum2.in 和 sum/sum2.ans。

【输入输出样例 3】

见选手目录下的 sum/sum3.in 和 sum/sum3.ans。

【数据说明】

对于第 1 组至第 2 组数据， $1 \leq n \leq 100, 1 \leq m \leq 5$ ；

对于第 3 组至第 4 组数据， $1 \leq n \leq 3000, 1 \leq m \leq 100$ ；

对于第 5 组至第 6 组数据， $1 \leq n \leq 100000, 1 \leq m \leq 100000$ ，且不存在出现次数超过 20 的颜色；

对于全部 10 组数据， $1 \leq n \leq 100000, 1 \leq m \leq 100000, 1 \leq color_i \leq m, 1 \leq number_i \leq 100000$ 。