

省队选拔模拟测试

考试时间：5 小时

题目名称	图书馆	决战圣诞树	抽象画
题目类型	传统型	传统型	提交答案型
目录	library	xmas	paint
可执行文件名	library	xmas	N/A
输入文件名	library.in	xmas.in	N/A
输出文件名	library.out	xmas.out	N/A
每个测试点时限	1.5 秒	9 秒	N/A
内存限制	512 MiB	1024 MiB	N/A
子任务数目	6	25	10
测试点是否等分	否	是	是

提交源程序文件名

对于 C++ 语言	library.cpp	xmas.cpp	N/A
对于 C 语言	library.c	xmas.c	N/A
对于 Pascal 语言	library.pas	xmas.pas	N/A

编译选项

对于 C++ 语言	-O2 -lm -std=c++11	N/A
对于 C 语言	-O2 -lm	N/A
对于 Pascal 语言	-O2	N/A

注意事项

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() 的返回值类型必须是 int，程序正常结束时的返回值必须是 0。
3. 若无特殊说明，结果比较方式为忽略行末空格、文末回车后的全文比较。

4. 评测时采用的机器配置为：Intel(R) Core(TM) i5 – 9500 CPU @ 3.00GHz，内存 8GB。上述时限以此配置为准。
5. 只提供 Linux 格式附加样例文件。
6. 选手应在选手文件夹内建立各个题目对应的子文件夹，各题的源程序应放在相应的子文件夹里。

图书馆（library）

【题目描述】

S 省最近新建了一座图书馆。

fy 和 scx 分别是 S 省 Θ 中学的高一 (e) 班和高一 (π) 班的学生，她们经常去图书馆阅览书籍。

你知道，fy 和 scx 去图书馆的时间都是周期性的。换句话说，fy 每 A 天去一次图书馆 (即每隔 $A - 1$ 天去一次)，scx 每 B 天去一次图书馆 (即每隔 $B - 1$ 天去一次)。

你今天在图书馆中恰好看到了 fy 和 scx。据图书管理员回忆，今天距离她们上次同时来图书馆恰好 D 天。

你已经不记得具体的 A, B, D 的值了。你只知道 D 是一个完全平方数。

偶然间，你从图书管理员的桌上得到了一张登记表，其中记录了 Θ 中学高一 (e) 班和高一 (π) 班的所有人的阅览周期 (即多少天去一次图书馆)。

你想知道，根据你所知道的信息，从高一 (e) 班和高一 (π) 班随机抽取一个人，有多少的概率可能是 fy 和 scx。

当然，图书管理员统计的信息可能会有误，因此，如果不存在合法的情况，则认为概率为 0。

设高一 (e) 班和高一 (π) 班分别有 n, m 个人，你只需要输出概率乘 $n \times m$ 的值即可，可以证明，这个值一定是一个整数。

【输入格式】

从文件 *library.in* 中读入数据。第一行包含两个正整数 n, m ，分别表示高一 (e) 班和高一 (π) 班的人数。

第二行包含 n 个正整数 a_i ，表示高一 (e) 班所有人的阅览周期。

第三行包含 m 个正整数 b_i ，表示高一 (π) 班所有人的阅览周期。

【输出格式】

输出到文件 *library.out* 中。输出一行一个整数，表示概率乘 $n \times m$ 的值。

【样例 1 输入】

```
3 3
1 1 2
1 3 4
```

【样例 1 输出】

```
5
```

【样例 1 解释】

我们定义每个人的学号为输入中的编号。则一共有 $3 \times 3 = 9$ 种可能性，其中如下 5 种可能性是可能的：

- 1. fy 是高一 (e) 班的 1 号，scx 是高一 (π) 班的 1 号。
- 2. fy 是高一 (e) 班的 1 号，scx 是高一 (π) 班的 3 号。
- 3. fy 是高一 (e) 班的 2 号，scx 是高一 (π) 班的 1 号。
- 4. fy 是高一 (e) 班的 2 号，scx 是高一 (π) 班的 3 号。
- 5. fy 是高一 (e) 班的 3 号，scx 是高一 (π) 班的 3 号。

所以概率为 $\frac{5}{9}$ ，输出 $\frac{5}{9} \times 3 \times 3 = 5$ 。

【样例 2 输入】

```
7 9
1 2 2 4 6 7 9
1 2 3 4 6 7 8 9 12
```

【样例 2 输出】

```
14
```

【样例 2 解释】

定义如上，则 14 种可能性如下：((a, b) 表示 fy 是高一 (e) 班的 a 号，scx 是高一 (π) 班的 b 号)
(1, 1), (1, 4), (1, 8), (2, 4), (3, 4), (4, 1), (4, 2), (4, 4), (4, 8), (7, 1), (7, 3), (7, 4), (7, 8), (7, 9)。
所以概率为 $\frac{2}{9}$ ，输出 $\frac{2}{9} \times 7 \times 9 = 14$ 。

【样例 3】

见选手目录下的 *library/library3.in* 与 *library/library3.ans*。

【数据范围】

对于所有的测试点，均有 $1 \leq n, m \leq 5 \times 10^6; 1 \leq a_i, b_i \leq 10^6$ 。
由于某些特殊的原因，本题采用捆绑测试。为了拿到子任务的分数，你必须通过那个子任务中的所有测试点。
具体的子任务的数据规模见下表：

子任务	分值	n, m	a_i, b_i	其他性质
1	9	$n, m \leq 2000$	$a_i, b_i \leq 10^6$	无
2	16	$n, m \leq 10^6$	$a_i, b_i \leq 2000$	
3	13		$a_i = b_i = i$	$n = m$
4	20			无
5	24	$n, m \leq 5 \times 10^6$	$a_i, b_i \leq 2 \times 10^5$	
6	18		$a_i, b_i \leq 10^6$	

决战圣诞树 (xmas)

【题目描述】

还有？天就到圣诞节了，回想起上次圣诞节，仿佛就像是在几星期前一样。fy 和 scx 还沉浸在去年圣诞晚会的回忆之中。

上星期，S 省 Θ 中学的高一段举行了圣诞晚会。其中有一个游戏叫做「决战圣诞树」，参与者要求有敏捷的计算能力。

根据 Θ 中学的传统，每年的圣诞晚会都会举办一场「决战圣诞树」，在今年，经过重重选拔，高一 (e) 班的 fy 和高一 (π) 班的 scx 选入了总决赛，她们将要在 2019 年 12 月 25 日进行最后的拼搏。「决战圣诞树」的游戏规则如下：

在她们的前面，有一棵漂亮的圣诞树，它可以抽象成 n 个节点的无根树。初始时，每个节点上没有任何礼物，且 fy 和 scx 手上各有一张魔法卡片 (可以使用多次)。

我们定义每个人都有一个**满意指数**，它的取值为 $[0, K - 1]$ 的一个整数，初始时为 0，其中 K 为主持人决定的一个整数。如果某一时刻有一个人的满意指数 $\geq K$ ，则会自动对 K 取模。

接下来将会有四种不同的事件发生：

- 主持人在节点 v 处放入了一个礼物。礼物分为两种：
 - 棒棒糖：有两种选择：不吃和吃。如果吃了，则她的**满意指数**会增加 a (然后对 K 取模，下略)；如果不吃则**满意指数**不变。 a 的值由棒棒糖决定。
 - 巧克力魔方：有三种选择：不要、吃了和当魔方玩。如果吃了，则**满意指数**会增加 b ；如果你把它当魔方玩，则**满意指数**会增加 c ，不过你就不能吃它了；如果不要则**满意指数**不变。 b, c 的值由巧克力魔方决定。
- fy 使用了她手上的魔法卡片。她会指定树上的一个节点 v ，然后 v 上的所有礼物都会**复制**到与 v 相邻的每个点上。
- scx 使用了她手上的魔法卡片。她会指定树上的一条边 $e = (u, v)$ ，然后将 u 上的所有礼物**复制**到“将 e 去掉后 v 所在的子树”中的每个点上。
- 主持人提了一个问题。他会指定树上的一条边 $e = (u, v)$ ，然后将边 (u, v) 断开。显然，这棵圣诞树会分为两棵子树。

然后，fy 会得到 u 所在的子树中的所有礼物，scx 会得到 v 所在的子树中的所有礼物。主持人还会指定一个数 $Sat \in [0, K - 1]$ ，每个人需要求出，有多少种方案使得合理利用自己的礼物，使得最终的**满意指数**为 Sat 。答得快的人在该轮中获胜。

考虑到答案可能过大，因此你只需要报出方案数模素数 P 的值即可。

当然，回答完毕后，边 (u, v) 将会被重新连上。

fy: “根据前几年的「决战圣诞树」, 我发现这个数 P 有特殊的性质诶! 对于每场比赛, $P \bmod K$ 的值都等于 $1!$ ”

scx: “是诶, 不过, 有这个性质, 怎样才能快速计算呢?”

fy 和 scx 都想在这场比赛中获胜, 于是她们找到了神机妙算的你, 希望你在尽量短的时间内告诉她们答案。你只需要输出方案数模素数 P 的值, 你可以假设一定有 $P \bmod K = 1$ 。

【输入格式】

从文件 `xmas.in` 中读入数据。第一行包含三个正整数 T, K, P , 其中 T 表示数据组数, P 为模数, K (满意指数上限) 的具体含义见题目描述。

接着描述 T 组数据。对于每组数据, 第一行包含三个非负整数 $n, q, \text{dataType}$, 分别表示圣诞树的节点个数、事件的个数以及数据类型。数据类型详见「限制与约定」。

接下来的 $n - 1$ 行, 每行两个正整数 u, v , 表示圣诞树中 u 与 v 之间有一条边。

接下来的 q 行, 每行描述一个事件, 格式如下:

1. `L v a` 表示主持人在节点 v 处放入了一个参数为 a 的棒棒糖。
2. `C v b c` 表示主持人在节点 v 处放入了一个参数分别为 (b, c) 的巧克力魔方。
3. `F v` 表示 fy 使用了她手上的魔法卡片, 且指定了节点 v 。
4. `S u v` 表示 scx 使用了她手上的魔法卡片, 且指定了边 (u, v) 。需要注意 u 和 v 是有顺序的。
5. `Q u v Sat` 表示主持人进行了一次询问, 且指定了边 (u, v) , 且指定了参考满意指数 Sat 。同样需要注意 u 和 v 是有顺序的。

【输出格式】

输出到文件 `xmas.out` 中。对于每次 `Q` 操作, 输出一行, 包含两个整数, 分别表示 fy 和 scx 利用礼物使得满意指数为 Sat 的方案数模 p 的值, 注意顺序。

【样例 1 输入】

```
1 119 974849
5 17 31
1 2
1 3
2 4
2 5
Q 1 3 0
Q 1 2 54
L 2 24
```

```
Q 1 2 24
C 1 20 30
Q 2 4 54
Q 1 2 54
Q 5 2 54
F 2
Q 1 2 24
Q 1 2 48
Q 1 2 72
S 1 3
Q 2 4 24
Q 2 4 48
Q 2 4 54
Q 2 4 72
```

【样例 1 输出】

```
1 1
0 0
0 1
1 0
0 0
0 1
1 3
0 3
0 1
4 1
6 0
8 0
4 0
```

【样例 1 解释】

我有一个绝妙的解释，可惜空间太小，写不下，如果你对样例有疑问，请提问。

【样例 2 输入】

```
1 119 974849
2 13 23
1 2
```



```
C 1 17 17
Q 1 2 17
Q 2 1 17
F 1
Q 1 2 17
F 1
Q 1 2 17
F 2
Q 1 2 17
S 1 2
Q 1 2 17
S 2 1
Q 1 2 17
```

【样例 2 输出】

```
2 0
0 2
2 2
2 4
6 4
6 10
272 10
```

【样例 2 解释】

最后一个时刻，1 号节点上有 8 个参数分别为 (17, 17) 的巧克力魔方。而想让满意指数为 17 有两种方案：

1. 使用 (包括吃了和当魔方玩) 1 个巧克力魔方，则满意指数为 $17 \times 1 \bmod 119 = 17$ ，共有 $\binom{8}{1} \times 2^1 = 16$ 种方案。
2. 使用 8 个巧克力魔方，则满意指数为 $17 \times 8 \bmod 119 = 17$ ，共有 $\binom{8}{8} \times 2^8 = 256$ 种方案。

综上，一共有 $16 + 256 = 272$ 种方案。

【样例 3】

见选手目录下的 *xmas/xmas3.in* 与 *xmas/xmas3.ans*。

这个样例，无疑是善良的出题人无私的馈赠。大量精心构造的 $n \leq 50; q \leq 100$ 的数据，涵盖了 dataType 从 0 到 31 所有的组合。你可以利用这个测试点，对自己的程序进行全面的检查。足量的数据组数、不大的数据范围和多种多样的数据类型，能让程序中的错误无处遁形。出题人相信，这个美妙的样例，可以给拼搏于 AC 这道题的逐梦之路上的你，提供一个有力的援助。

【数据范围】

对于所有的测试点，均有 $2 \leq n \leq 5000; 0 \leq q \leq 5000; 2 \leq K \leq 200; 10^5 \leq P \leq 2 \times 10^6; P \bmod K = 1$ ，且 P 是素数。

对于所有的事件，保证 $1 \leq u, v \leq n; 0 \leq a, b, c, Sat \leq K - 1$ 。

本题共有 25 个测试点，每个测试点的分值为 4 分。在每个测试点中，所有 n 的和不会超过 23333，所有 q 的和不会超过 66666。

测试点编号	n	q	T	K	dataType	
1	≤ 5000	$= 0$	无限制	≤ 200	$\preceq 31 = (11111)_2$	
2	≤ 6	≤ 6	≤ 6	≤ 6		
3	≤ 20	≤ 25	≤ 20	≤ 25		
4	≤ 100	≤ 100	≤ 50			
5	≤ 5000	≤ 5000	无限制		≤ 25	$\preceq 16 = (10000)_2$
6						$\preceq 19 = (10011)_2$
7				$\preceq 23 = (10111)_2$		
8	$\preceq 24 = (11000)_2$					
9	≤ 100	≤ 100		$= 2$	$\preceq 27 = (11011)_2$	
10					$\preceq 31 = (11111)_2$	
11					$\preceq 24 = (11000)_2$	
12	≤ 5000	≤ 5000			$\preceq 27 = (11011)_2$	
13			$\preceq 31 = (11111)_2$			
14	≤ 2000	≤ 2000	≤ 10		≤ 200	$\preceq 8 = (01000)_2$
15						$\preceq 9 = (01001)_2$
16	≤ 5000	≤ 5000	无限制			$\preceq 11 = (01011)_2$
17				$\preceq 13 = (01101)_2$		
18	≤ 2000	≤ 2000	≤ 10	$\preceq 15 = (01111)_2$		
19	≤ 5000	≤ 5000	无限制	$\preceq 24 = (11000)_2$		
20	≤ 2000	≤ 2000	≤ 10			
21						$\preceq 25 = (11001)_2$
22	≤ 5000	≤ 5000	无限制			$\preceq 27 = (11011)_2$
23				$\preceq 29 = (11101)_2$		
24	≤ 2000	≤ 2000	≤ 10	$\preceq 31 = (11111)_2$		
25	≤ 5000	≤ 5000	无限制			

数据类型的含义：数据类型为一个 $0 \sim 31$ 的整数，我们可以将其看作一个长度为 5 的二进制串。其中位权为 1, 2, 4, 8, 16 的位的值分别表示：

- 位权为 1 的位如果是 0，则表示没有 C 事件，否则**可能有** C 事件；
- 位权为 2 的位如果是 0，则表示没有 F 事件，否则**可能有** F 事件；
- 位权为 4 的位如果是 0，则表示没有 S 事件，否则**可能有** S 事件；
- 位权为 8 的位如果是 0，则一定保证原树是链，且第 i 条边的 $u_i = i \wedge v_i = i + 1$ ，否则**不保证** 树的任何性质；
- 位权为 16 的位如果是 0，则保证 $a = b = c = Sat = 0$ ，否则**不保证** a, b, c, Sat 之间的性质。

对于两个表示**数据类型**的 (二进制) 整数 a, b ， $a \preceq b$ 当且仅当 $(a \& b) = a$ ，其中 $\&$ 表示**按位与** 运算。

抽象画 (paint)

这是一道提交答案题

【题目描述】

fy 和 scx 在 S 省 Θ 中学的艺术教室上美术课。

这节课上，美术老师教同学们如何画抽象画。

具体地，每个人会拿到一张白纸，纸上有 $2n$ 个点，保证没有三点共线。你需要画 n 条线段，将它们两两配对。

当然，由于传统的审美观念，如果有两条线段相交，这张画就显得非常难看。因此，需要保证所画的 n 条线段两两不相交。

美术老师定义一幅画的抽象程度为所有 n 条线段的长度之和。整个班里抽象程度最大的画的创作者会得到他的神秘奖励。

fy 和 scx 想得到这份神秘奖励，然而她们并不擅长美术，因此她们找到了擅长美术的你，希望你画一张抽象程度尽可能大的画。

【输入格式】

所有输入数据 paint1.in ~ paint10.in 见下发文件，分别对应 10 个子任务。

对于每组数据，第一行包含一个正整数 n ($n \leq 2000$)，表示点的个数 (的一半)。

接下来的 $2n$ 行，每行两个整数 x_i, y_i ，表示第 i 个点的坐标为 (x_i, y_i) 。

【输出格式】

输出文件为 paint1.out ~ paint10.out，分别对应相应的输入文件。

对于每组数据，需要输出 $2n$ 个正整数 (可以在一行内，也可以输出 $2n$ 行)，第 i 个整数 a_i 表示第 i 个点与第 a_i 个点相连。你需要保证 $a_i \neq i$ 且 $a_{a_i} = i$ 。

你需要使 n 条线段的长度之和尽可能大。

【样例 1 输入】

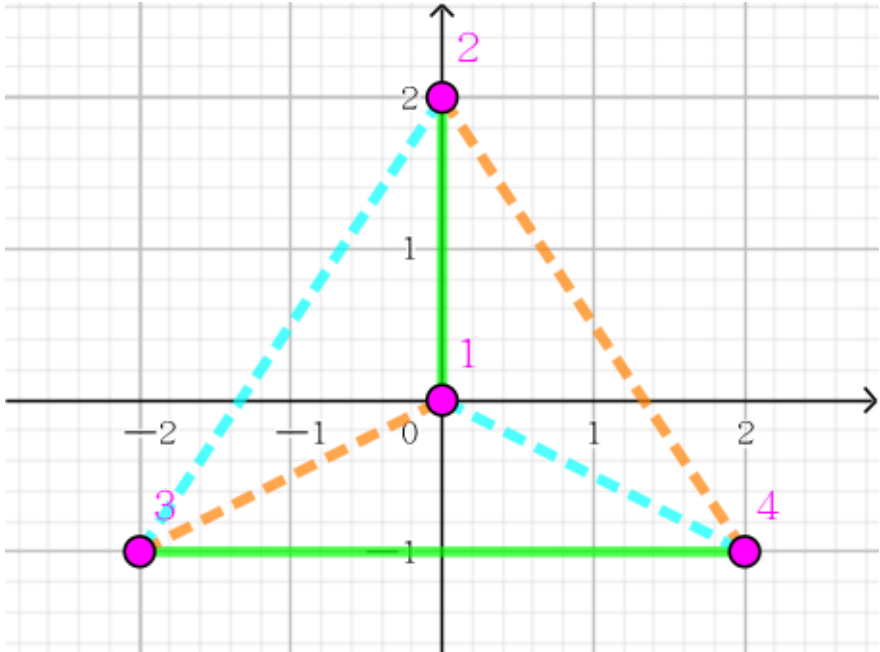
```
2
0 0
0 2
-2 -1
2 -1
```

【样例 1 输出】

2 1 4 3

【样例 1 解释】

4 个点的位置如下图所示：



共有 3 种配对方式，如果我们将 1 号点与 2 号点配对，则线段总长度为 6，如果将 1 号点与 3 (或 4) 号点配对，则线段总长度为 $\sqrt{5} + \sqrt{13} \approx 5.842 < 6$ 。
因此样例输出的方案是最优的。(当然，你并不一定要求出最优解)

【样例 2 输入】

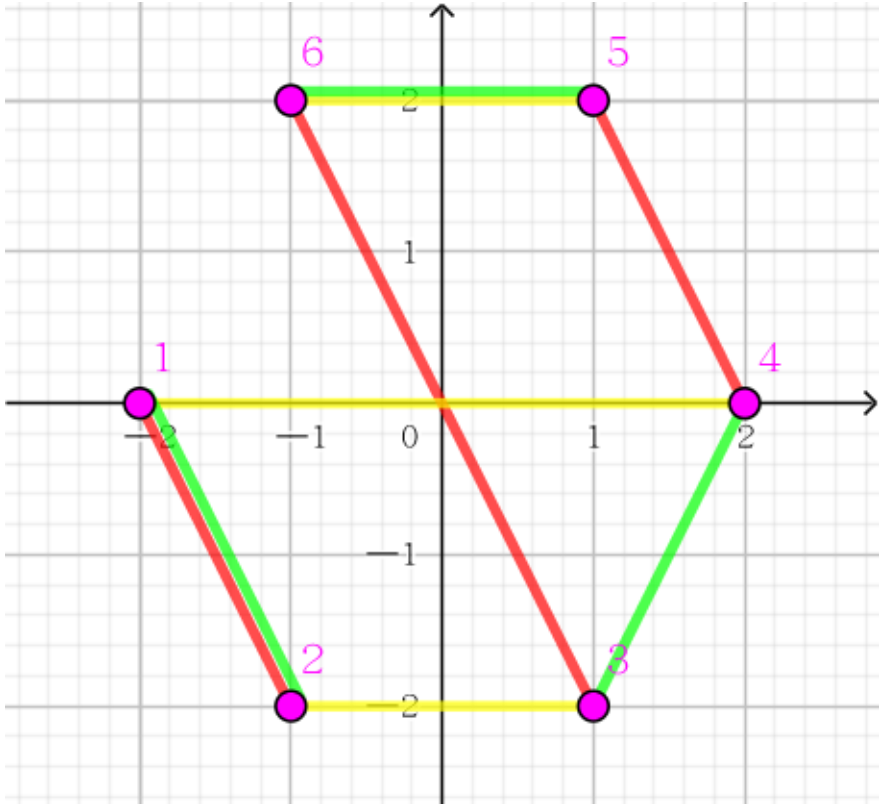
3
-2 0
-1 -2
1 -2
2 0
1 2
-1 2

【样例 2 输出】

2 1 4 3 6 5

【样例 2 解释】

6 个点的位置如下图所示：



由上图，红色、黄色、绿色的配对方式所对应的线段总长度分别为 $4\sqrt{5} \approx 8.944$, $8 = 8.000$, $2 + 2\sqrt{5} \approx 6.472$ ，因此红色的配对方式是最优的。

当然，你可以输出其它的解（只要合法），比如此样例输出了绿色的配对方案，可以获得一定的部分分（具体参见「子任务与部分分」）。

但是，配对 $\{(1,4),(2,5),(3,6)\}$ 是不合法的，因为线段 $(1,4)$ 和线段 $(2,5)$ 相交了。

【子任务及部分分】

我们提供了十个评分文件 `paint1.ans` ~ `paint10.ans`，分别对应每个测试点。每个评分文件共 10 行，第 i 行一个评分参数 w_i ，具体意义将在下面给出。本题中，每个测试点单独进行评分，每个测试点 10 分。如果选手的输出格式不合法或者参数不符合题目约定，则得 0 分。

否则，测评器会按照你的输出，执行对应的操作，并按照以下规则评分。

对于每个测试点，我们设置了 10 个评分参数 $w_1, w_2, w_3, \cdots, w_9, w_{10}$ 。

假设最终 n 条线段的总长度为 w ，则你的分数将会由下表给出：

得分	条件	得分	条件
10	$w \geq w_{10}$	5	$w \geq w_5$
9	$w \geq w_9$	4	$w \geq w_4$
8	$w \geq w_8$	3	$w \geq w_3$
7	$w \geq w_7$	2	$w \geq w_2$
6	$w \geq w_6$	1	$w \geq w_1$

若不符合表中所有条件，得 0 分；若符合表中的多个条件，则取分数最高的。

【如何测试你的输出】

~~相信你们是自己会写 checker 滴。~~

~~如果要高精度，建议使用 Python3 的 Decimal 库，最终测试的 checker 就会使用此库。~~

~~关于判断两条线段是否相交，这是计算几何基础，这里就不再赘述了。~~

如果选手所在机器上装有 Python3，则可以使用下发的 checker，使用方法如下：

```
./checker <input_file> <output_file> <answer_file>
(windows 用户为 checker <input_file> <output_file> <answer_file>)
```

如果无法运行,你需要 `chmod +x checker` 来提升权限或使用 `python3 checker <input_file> <output_file> <answer_file>`。