

T1 duck

输入格式：duck.in

输出格式：duck.out

时间限制：1s

空间限制：512MB

数据点个数：20

题目背景

duck 老是喜欢剥削你，不过他也有自己的原则

题目描述

duck 会告诉你他喜欢的数为 m 。

你手上有 $2n$ 堆糖果，第 i 堆糖果数量 a_i ，由于在 puck 的剥削下，糖果数量满足 $0 \leq a_i < m$ 。

你需要把这 $2n$ 堆糖果重新两两组成一堆。若你将两堆数量分别为 x 和 y 的糖果组成一堆，由于 duck 的剥削，新的一堆糖果的数量只会剩下 $(x + y) \% m$ 。

duck 最后会在你重组成的 n 堆新糖果中抢走数量最多的一堆。

你对 duck 一直怀恨在心，但苦于无法明面上反抗，你只能依靠暗中操作，使得 puck 拿到的糖果数量最少。

求 duck 最后能拿到的最少糖果数。

输入格式

第一行输入两个数 n, m ，分别表示你有 $2n$ 堆糖果和 puck 喜欢的数

第二行输入 $2n$ 个数，分别表示 a_i

输出格式

一行输出，表示答案

输入样例

输入 #1

```
1 | 3 10
2 | 0 2 3 4 5 9
```

输出 #1

```
1 | 5
```

输入 #2

1	2 10
2	1 9 1 9

输出 #2

1	0
---	---

说明/提示

样例解释

对于 #1, 选择 $(0, 5), (2, 3), (4, 9)$ 分别组成一堆

对于 #2, 选择 $(1, 9), (1, 9)$ 分别组成一堆

数据范围

对于 5% 的数据, 保证 $m = 1$

对于另外 5% 的数据, 保证 $a_i = 0$

对于另外 5% 的数据, 保证 $n = 1$

对于 30% 的数据, 保证 $n \leq 10^2$

对于 60% 的数据, 保证 $n \leq 5 \times 10^3$

对于 100% 的数据, 保证 $1 \leq n \leq 10^5, 1 \leq m \leq 10^9, 0 \leq a_i < m$

T2 最短路(graph.cpp, 3s/512MB)

题目描述

你有一个 n 个点的有向完全图和一个数 m 。

每个点有两个属性 a_i 和 b_i 。 $u \rightarrow v$ 的边的权值是 $(a_u + b_v) \bmod m$ 。

求 1 到 n 的最短路。

输入格式

本题的测试点包含有多组测试数据。

输入的第一行包含两个整数 c 和 t , 分别表示测试点编号和测试数据组数。对于样例, $c = 0$ 。

接下来, 对于每组测试数据:

第一行两个整数 n, m 。

第二行 n 个整数表示 $\{a_i\}$ 。

第三行 n 个整数表示 $\{b_i\}$ 。

输出格式

对于每组数据，一行一个整数表示 1 到 n 的最短路。

输入输出样例

输入 #1

1	0	1
2	4	12
3	10	11 6 0
4	8	7 4 1

输出 #1

1	3
---	---

输入 #2

1	0 1
2	10 1000
3	785 934 671 520 794 168 586 667 411 332
4	363 763 40 425 524 311 139 875 548 198

输出 #2

1	462
---	-----

提示

对于所有数据数据，保证 $1 \leq t \leq 4$, $2 \leq n \leq 2 \times 10^5$, $2 \leq m \leq 10^9$, $0 \leq a_i, b_i < m$ 。

测试点编号	$n \leq$	特殊性质
1, 2	10^3	无
3	2×10^5	A
4	10^5	B
5, 6, 7	2×10^5	C
8	5×10^4	无
9, 10	2×10^5	无

特殊性质 A：保证 a_i, b_i 在 $[0, m - 1]$ 中随机均匀生成。

特殊性质 B：保证 $m \leq 100$ 。

特殊性质 C：保证 $\max a_i + b_i < m$ 。

T3回文串(substr)

时间限制 2500ms

空间限制 512MB

题目背景

题目背景没有意义，可以跳过。

回来吧 ❤️ 回文题目 🤖 🤖 🤖

我最骄傲的信仰 🤖

历历在目的哈希 🤖

眼泪莫名在流淌 🤖

依稀记得manacher 🤖

还有回文自动机 🤖

把数据都给通过 🤖

就算通宵也不累 🤖

回文串，回文串，我们喜欢你

我们喜欢 自动机，manacher，哈希

回文串，回文串，我们喜欢你

真的真的有意思，有意思

开大范围前，我从 10^5 踏上征途，开始了第二次验题，本题各种暴力遂归于一统。

本题原范围所到之处，选手竭诚欢迎，真可谓占尽天时，那种勃勃生机、万物竞发的境界，犹在眼前。

区区开大范围之后，这里竟至于一变而成为出题人的葬身之地了么？

无论怎么样，数据范围是二十五万对十万，优势在我！

题目描述

对于一个字符串 S ，下标从 1 到 $|S|$ ，定义 S 的格式数组为 rk

若 $i \leq \frac{S}{2}$ ， $rk_i = \sum_{j=i+1}^{\lfloor \frac{S}{2} \rfloor} [S_j \leq S_i]$ 。

否则 $rk_i = \sum_{j=\lceil \frac{S}{2} \rceil + 1}^{i-1} [S_j \leq S_i]$ 。

对于一个字符串 S ，称 S 是回文的，当且仅当 $|S|$ 是偶数且 $\forall i \in [1, |S|], rk_i = rk_{|S|-i+1}$ 。

现在给你一个字符串 T ，请你求出其最长回文子串长度。

输入格式

第一行一个整数 $|T|$

第二行 $|T|$ 个在 $[1, |T|]$ 内的整数，表示 T 。

输出格式

一行一个整数，表示最长回文子串长度。

输入输出样例

样例输入 #1

```
1 | 8
2 | 3 3 3 3 1 2 3 4
```

样例输出 #1

```
1 | 8
```

样例解释 #1

对于 T 的子串 T ，其 rk 数组为 $\{3, 2, 1, 0, 0, 1, 2, 3\}$

满足 $\forall i \in [1, |T|], rk_i = rk_{|T|-i+1}$ ，所以 T 回文，所以答案为 $|T| = 8$ 。

提示说明

对于 20% 的数据， $|T| \leq 10^3$

对于 40% 的数据， $|T| \leq 4 \times 10^3$

对于 70% 的数据， $|T| \leq 10^5$

对于另外 10% 的数据，保证 $\forall i \in [1, |T|), T_i \leq T_{i+1}$

对于 100% 的数据， $|T| \leq 2.5 \times 10^5$

T4德扑 (poker)

时限见数据范围，空间限制 512MiB

题目背景

lfxxx 邀请你和他一起玩德州扑克。

之后又来了一群人玩德州扑克，随着人数的增多，手动计算筹码和胜率太麻烦，所以学 OI 的你需要写一个程序帮助他们自动化处理。

题目描述

德州扑克一共有 52 张牌，没有王牌，lfxxx 一共找了 n 位玩家来玩德州扑克，玩家 i 的下一位玩家为玩家 $i + 1$ ，玩家 n 的下一位玩家为玩家 1。游戏流程分为 6 个阶段：准备、翻牌前、翻牌、转牌、河牌、比牌。

准备：每位玩家得到两张底牌，玩家 2 向底池投入 d_s 单位筹码，这一部分筹码**不属于**任何一位玩家的下注。

翻牌前：玩家 1 下注 d_1 单位筹码，玩家 2 下注 d_2 单位筹码（这一部分筹码称为盲注），从玩家 2 的下一位玩家开始，所有没有弃牌或 All in 的玩家依次进行从以下 3 个操作选择一个进行（翻牌，转牌，河牌也要进行）：

- 跟注：将自己本轮的下注筹码补至**本轮中**上一个选择加注的玩家的下注额，若当前没有玩家**在本轮中**选择加注，则将下注筹码补至 d_2 。若剩下的筹码不够跟注或跟注后没有筹码剩余，则下注所有筹码，并宣告 All in。
- 加注：若“跟注”需要下注 x 单位筹码，则选择一个整数 y 满足 $y > x$ ，下注 y 单位筹码。加注额不能超过现有筹码量，若加注后没有筹码剩余，宣告 All in。若有人选择加注，之前行动过的玩家需要在轮到他们时重新行动。
- 弃牌：退出本局游戏。弃牌不能拿回之前下注的筹码。任意时刻**未弃牌的玩家只剩一位，游戏立即结束**，未弃牌的玩家收取所有底池。**一旦一个玩家选择弃牌，他就不能再在本轮的后续下注中参与。**

一个阶段结束后，若只有一位玩家未弃牌且未 All in 且至少有一个玩家 All in，将公牌补至五张，直接开始比牌。

翻牌与翻牌前的区别为：发三张公牌，玩家 1 和玩家 2 不需要下盲注，行动顺序变为从玩家 1 开始，无人加注时选择跟注不需要下筹码。

转牌与翻牌的区别为：只发第四张公牌；河牌与翻牌的区别为：只发第五张公牌。

比牌时，所有未弃牌的玩家摊开底牌，将自己的底牌与 5 张公牌结合比大小，牌面最大的所有玩家均分底池。若玩家 i 的总下注额为 x ，他只能参与均分所有玩家的总下注额中不超过 x 的部分。若牌面最大的所有玩家中总下注额最高为 y ，且存在玩家的总下注额高于 y ，则将所有总下注额高于 y 的玩家拉出来，将总下注额减去 y 后再进行一次比牌。玩家 2 在准备阶段向底池投入 d_s 单位筹码仅在第一轮比牌参与均摊。

一副牌面共五张牌，五张牌的顺序与其牌型和大小没有关系，共有 9 种牌型，从大往小分别为同花顺，四条，同花，顺子，三条，两对，对子，高牌。（S,H,C,D 分别表示花色为黑桃，红桃，梅花，方片）

牌型	条件	同等牌型比较方式	示例
同花顺	同时满足同花和顺子的条件	比较最大一张牌的点数（5432A 的最大点数为 5）	9S 8S 7S 6S 5S
四条	存在四张牌的点数一致	比较四张部分的点数，若相同，比较第五张牌的点数	KS KH KC AD KD
葫芦	存在三张牌的点数一致，且另外两张牌的点数一致	比较三张部分的点数，若相同，比较另外两张牌的点数	JS JH JD 9C 9D
同花	五张牌花色一致	从大往小依次比较每张牌的点数	KH 9H 6H 3H 2H

牌型	条件	同等牌型比较方式	示例
顺子	五张牌的点数顺次排列	比较最大一张牌的点数（5432A 的最大点数为 5）	AS KH QC JD 10S
三条	存在三张牌的点数一致	比较三张部分的点数，若相同，从大往小依次比较剩下的每张牌的点数	KS 8H 8C 8D 5C
两对	存在两张牌的点数一致，且另外三张牌中存在两张牌的点数一致	比较点数较大两张部分的点数，若相同，比较点数较小两张部分的点数，若依然相同，比较第五张牌的点数	QC QD 8S 3H 3D
对子	存在两张牌的点数一致	比较两张部分的点数，若相同，从大往小依次比较剩下的每张牌的点数	AH JS JC 7H 6D
高牌	不满足以上任意一个条件	从大往小依次比较剩下的每张牌的点数	AH QC JD 9S 7S

其中，对于点数的大小关系： $A > K > Q > J > 10 > 9 > 8 > 7 > 6 > 5 > 4 > 3 > 2$ 。

你的任务是：计算每个阶段每个玩家的胜率，以及算出最后每个玩家筹码数变化。一个玩家在某个阶段的胜率指的是**这个阶段玩家下注前，在所有没有弃牌玩家的牌面中，若接下来的公牌随机均匀地发出，最后该玩家牌面为最大之一的概率。**

输入格式

第一行四个整数 n, d_1, d_2, d_s 。

接下来 n 行一个整数和一个由大写字母和数字组成的长度为 4 的字符串，分别表示第 i 位玩家的初始筹码数和底牌信息。具体地，第 i 行字符串前两个字符表示第 i 位玩家第一张底牌的点数和花色，后两个字符表示第二张底牌的点数和花色，**特殊地，我们用 T 表示 10**，SHCD 分别表示花色为黑桃，红桃，梅花，方片。接下来若干行：

- 若为一个整数，如果不为 -1 ，表示按照规则对应玩家下注的筹码数（为 0 表示没有下筹码）；若为 -1 ，表示按照规则对应的玩家弃牌。
- 否则为一个字符串，表示对应阶段开的公牌信息，公牌信息与底牌信息表示方法类似，每张底牌信息均由两个字符表示，注意是点数在前，点数在后。（即使一个阶段结束后，若只有一位玩家未弃牌且未 All in 且至少有一个玩家 All in，每个阶段的牌也会分行给出）

注意，一个玩家弃牌或 All in 后不会再有关于这个玩家的下注信息。当未弃牌的玩家只剩一位时，由于游戏结束，输入文件也会结束。

输出格式

若干行，除最后一行，表示每个阶段每个玩家的胜率（若已弃牌，输出 -----（5 个 -），否则已百分数的形式输出，保留一位小数），用空格隔开。输出完一个阶段后换行。

最后一行， n 个带符号实数（保留两位小数），第 i 个实数表示第 i 个玩家的筹码变化。

（具体可见样例输出）

样例 #1

样例输入 #1

```
1 4 100 200 200
2 200000 AC6C
3 200000 ASQS
4 200000 JSJD
5 200000 AHKD
6 2500
7 2500
8 2400
9 4800
10 2500
11 9500
12 -1
13 -1
14 7000
15 ADJHKH
16 6000
17 22000
18 16000
19 2D
20 0
21 50000
22 50000
23 6S
24 84000
25 84000
```

样例输出 #1

```
1 22.8% 22.5% 37.2% 18.8%
2 ----- ----- 85.1% 16.2%
3 ----- ----- 95.0% 5.0%
4 ----- ----- 100.0% 0.0%
5 -2500.00 -5200.00 +175700.00 -168000.00
```

样例 #2

样例输入 #2

1	4	500	1000	1000
2	2000	ASAH		
3	5000	ACAD		
4	100000	KSKC		
5	100000	QDJJ		
6	5000			
7	5000			
8	1500			
9	3000			
10	2S9STS			
11	95000			
12	95000			
13	7S			
14	KD			

样例输出 #2

1	54.1%	53.3%	34.6%	26.0%
2	77.2%	41.7%	17.6%	19.9%
3	100.0%	0.0%	0.0%	0.0%
4	100.0%	0.0%	0.0%	0.0%
5	+7000.00	-5000.00	+98000.00	-100000.00

样例 #3

样例输入 #3

1	5	200	400	400
2	5000	AS2S		
3	5000	3D3C		
4	5000	8D9D		
5	5000	KSJC		
6	5000	QD6S		
7	400			
8	1200			
9	-1			
10	-1			
11	800			
12	800			
13	QSTDAD			
14	0			
15	3000			
16	-1			
17	-1			

样例输出 #3

1	22.5%	21.1%	23.8%	20.4%	13.9%
2	-----	3.2%	33.6%	64.2%	-----
3	-200.00	-1600.00	+3000.00	-1200.00	0.00

样例 #4

样例输入 #4

1	4	100	200	200
2	137000	ASQC		
3	213000	8D9D		
4	217500	THJC		
5	116000	KS6S		
6	1200			
7	1200			
8	1100			
9	1000			
10	AHQHJC			
11	4000			
12	4000			
13	8000			
14	8000			
15	4000			
16	4000			
17	KD			
18	0			
19	10800			
20	10800			
21	30800			
22	30800			
23	20000			
24	20000			
25	TC			
26	0			
27	3000			
28	6000			
29	6000			
30	15000			
31	12000			
32	9000			
33	24000			
34	15000			
35	15000			
36	15000			
37	KD			
38	67000			
39	142800			
40	147500			
41	46000			

样例输出 #4

1	25.7%	28.2%	21.7%	24.8%
2	61.3%	1.7%	28.6%	11.5%
3	17.1%	7.3%	90.2%	7.3%
4	100.0%	100.0%	100.0%	100.0%
5	+50.00	-150.00	+50.00	+50.00

样例 #5

样例输入 #5

1	3	1	2	2
2	100	ASAH		
3	100	2S2C		
4	100	2H2D		
5	2			
6	-1			
7	96			
8	96			
9	KSKHKC			
10	KD			
11	QS			

样例输出 #5

1	97.0%	33.5%	33.5%
2	-----	100.0%	100.0%
3	-----	100.0%	100.0%
4	-----	100.0%	100.0%
5	-1.00	-0.50	+1.50

样例 #6

样例输入 #6

1	2	1	2	1
2	100	ASTD		
3	100	3S4S		
4	10			
5	9			
6	7HJC8H			
7	10			
8	-1			

样例输出 #6

1	56.8%	43.7%
2	77.5%	23.7%
3	+12.00	-12.00

提示

【样例 1 解释】

初始每个玩家有 200000 筹码，翻牌前底池有 $d_s + d_1 + d_2 = 500$ 筹码，其中玩家 1 下注 100 筹码，玩家 2 下注 200 筹码。此时玩家 1 有 199800 筹码，玩家 2 有 199600 筹码。

注意 d_s 不算玩家 2 的下注但仍是由玩家 2 投入。

底牌信息如下：

玩家 1: AC (梅花 A) 6C (梅花 6)

玩家 2: AS (黑桃 A) QS (黑桃 Q)

玩家 3: JS (黑桃 J) JD (方片 J)

玩家 4: AH (红桃 A) KH (红桃 K)

玩家 2 的下一位玩家 3 加注到 2500 筹码，玩家 4 跟注，玩家 1 跟注，（由于他本轮已经下过 100 筹码了，因此只要下注 2400 筹码）玩家 2 加注到 5000 筹码（注意这时只下注了 4800 筹码，理由和前面类似，后面不在赘述）。玩家 3 跟注，玩家 4 加注到 12000 筹码。玩家 1, 2 弃牌，玩家 3 跟注。

翻牌为 AD (方片 A) JH (红桃 J) KH (红桃 K)。

翻牌下注本应是从玩家 1 开始，但玩家 1 弃牌，一直顺延到玩家 3。玩家 3 加注到 6000，玩家 4 加注到 22000，玩家 3 跟注。

转牌为 2D (方片 2)。

玩家 3 跟注（由于这个阶段无人加注因此没有下筹码），玩家 4 加注到 50000，玩家 3 跟注。

河牌为 6S (梅花 6)，玩家 3 加注到 84000，玩家 4 跟注。

玩家 3 能组成最大的牌型为三条 (JS,JH,JD,AD,KH)，玩家 4 能组成最大的牌型为对子 (AK,AD,KH,KD,JH)，玩家 3 赢下底池。

【样例 2 解释】

翻牌前，玩家 1, 2 all in，翻牌阶段，玩家 3, 4 all in。

玩家 1 的牌型最大 (A high 同花)，但他只下注了 2000，因此他只能参与均分每人的下注中不超过 2000 的部分，每人 2000 加上底池 $d_s = 1000$ 共 9000，没有人的牌与玩家 1 同样大，所以玩家 1 手下这一部分底池，排开自己下的 2000，共赢 7000。

除开玩家 1 外，最大的牌为玩家 3 是 K high 同花，玩家 3 的下注充足，没有人的牌与玩家 3 同样大，他收下剩下的所有底池。

【样例 5 解释】

由于多人均分，筹码变化可能不是整数。

【样例 6 解释】

下注部分的第一个 10 指的是玩家 1 加注至 11。

【数据范围】

本题一共有 50 个测试点，每个测试点 2 分，各测试点的相关限制见下表。

测试点编号	$n =$	时间限制
1, 2	20	1s
3, 4	20	5s
5, 6	2	1s
7, 8	2	5s
9, 10	3	1s
11, 12	3	5s
13, 14	4	3s
15, 16	5	3s
17, 18	6	3s
19, 20	7	1s
21, 22	7	3s
23, 24	8	3s
25, 26	9	3s
27, 28	10	5s
29, 30	11	1s
31, 32	11	1s
33, 34	12	1s
35, 36	13	3s
37, 38	14	3s
39, 40	15	1s
41, 42	16	5s
43, 44	17	1s
45, 46	18	3s

测试点编号	$n =$	时间限制
47, 48	18	5s
49, 50	19	3s