全国信息学奥林匹克联赛（NOIP2009）复赛

普及组

（请选手务必仔细阅读本页内容）

一．题目概况

中文题目名称	多项式输出	分数线划定	细胞分裂	道路游戏
英文题目名称	poly	score	cell	game
可执行文件名	poly	score	cell	game
输入文件名	poly.in	score.in	cell.in	game.in
输出文件名	poly.out	score.out	cell.out	game.out
每个测试点时限	1 秒	1 秒	1 秒	1 秒
测试点数目	10	10	10	20
每个测试点分值	10	10	10	5
附加样例文件	有	有	有	有
结果比较方式	全文比较 过滤行末空格及文末回车	全文比较 过滤行末空格及文末回车	全文比较 过滤行末空格及文末回车	全文比较 过滤行末空格及文末回车
题目类型	传统	传统	传统	传统

二．提交源程序文件名

对于 pascal 语言	poly.pas	score.pas	cell.pas	game.pas
对于 C 语言	poly.c	score.c	cell.c	game.c
对于 C++语言	poly.cpp	score.cpp	cell.cpp	game.cpp

三．编译命令（不包含任何优化开关）

对于 pascal 语言	fpc poly.pas	fpc score.pas	fpc cell.pas	fpc game.pas
对于 C 语言	gcc -o poly poly.c -lm	gcc -o score score.c -lm	gcc -o cell cell.c -lm	gcc -o game game.c -lm
对于 C++语言	g++ -o poly poly.cpp -lm	g++ -o score score.cpp -lm	g++ -o cell cell.cpp -lm	g++ -o game game.cpp -lm

四．运行内存限制

内存上限	128M	128M	128M	128M

注意事项：

1、文件名（程序名和输入输出文件名）必须使用小写。

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、全国统一评测时采用的机器配置为：CPU 1.9GHz，内存 1G，上述时限以此配置为准。各省在自测时可根据具体配置调整时限。

1. 多项式输出

(poly.pas/c/cpp)

【问题描述】

一元 n 次多项式可用如下的表达式表示：

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \ldots + a_1 x + a_0 ,\ \ a_n \neq 0$$

其中，$a_i x^i$ 称为 i 次项，a_i 称为 i 次项的系数。给出一个一元多项式各项的次数和系数，请按照如下规定的格式要求输出该多项式：

1. 多项式中自变量为 x，从左到右按照次数递减顺序给出多项式。
2. 多项式中只包含系数不为 0 的项。
3. 如果多项式 n 次项系数为正，则多项式开头不出现“+”号，如果多项式 n 次项系数为负，则多项式以“-”号开头。
4. 对于不是最高次的项，以“+”号或者“-”号连接此项与前一项，分别表示此项系数为正或者系数为负。紧跟一个正整数，表示此项系数的绝对值（如果一个高于 0 次的项，其系数的绝对值为 1，则无需输出 1）。如果 x 的指数大于 1，则接下来紧跟的指数部分的形式为“x^b”，其中 b 为 x 的指数；如果 x 的指数为 1，则接下来紧跟的指数部分形式为“x”；如果 x 的指数为 0，则仅需输出系数即可。
5. 多项式中，多项式的开头、结尾不含多余的空格。

【输入】

输入文件名为 poly.in，共有 2 行

第一行 1 个整数，n，表示一元多项式的次数。

第二行有 n+1 个整数，其中第 i 个整数表示第 n-i+1 次项的系数，每两个整数之间用空格隔开。

【输出】

输出文件 poly.out 共 1 行，按题目所述格式输出多项式。

【输入输出样例 1】

poly.in	poly.out
5 100 -1 1 -3 0 10	100x^5-x^4+x^3-3x^2+10

【输入输出样例 2】

poly.in	poly.out
3 -50 0 0 1	-50x^3+1

【数据范围】

1≤n≤100，多项式各次项系数的绝对值均不超过 100。

2．分数线划定

(score.pas/c/cpp)

【问题描述】

世博会志愿者的选拔工作正在 A 市如火如荼的进行。为了选拔最合适的人才，A 市对所有报名的选手进行了笔试，笔试分数达到面试分数线的选手方可进入面试。面试分数线根据计划录取人数的150%划定，即如果计划录取m名志愿者，则**面试分数线为排名第m*150%（向下取整）名的选手的分数**，而最终进入面试的选手为笔试成绩不低于面试分数线的所有选手。

现在就请你编写程序划定面试分数线，并输出所有进入面试的选手的报名号和笔试成绩。

【输入】

输入文件名为 score.in。

第一行，两个整数 n，m（5 ≤ n ≤ 5000，3 ≤ m ≤ n），中间用一个空格隔开，其中 n 表示报名参加笔试的选手总数，m 表示计划录取的志愿者人数。输入数据保证 m*150%向下取整后小于等于 n。

第二行到第 n+1 行，每行包括两个整数，中间用一个空格隔开，分别是选手的报名号 k（1000 ≤ k ≤ 9999）和该选手的笔试成绩 s（1 ≤ s ≤ 100）。数据保证选手的报名号各不相同。

【输出】

输出文件 score.out。

第一行，有两个整数，用一个空格隔开，第一个整数表示面试分数线；第二个整数为进入面试的选手的实际人数。

从第二行开始，每行包含两个整数，中间用一个空格隔开，分别表示进入面试的选手的报名号和笔试成绩，按照笔试成绩从高到低输出，如果成绩相同，则按报名号由小到大的顺序输出。

【输入输出样例】

score.in	score.out
6 3 1000 90 3239 88 2390 95 7231 84 1005 95 1001 88	88 5 1005 95 2390 95 1000 90 1001 88 3239 88

【样例说明】

m*150% = 3*150% = 4.5，向下取整后为 4。保证 4 个人进入面试的分数线为 88，但因为 88 有重分，所以所有成绩大于等于 88 的选手都可以进入面试，故最终有 5 个人进入面试。

3. 细胞分裂

(cell.pas/c/cpp)

【问题描述】

Hanks 博士是 BT (Bio-Tech，生物技术) 领域的知名专家。现在，他正在为一个细胞实验做准备工作：培养细胞样本。

Hanks 博士手里现在有 N 种细胞，编号从 1~N，一个第 i 种细胞经过 1 秒钟可以分裂为 S_i 个同种细胞（S_i 为正整数）。现在他需要选取某种细胞的一个放进培养皿，让其自由分裂，进行培养。一段时间以后，再把培养皿中的所有细胞平均分入 M 个试管，形成 M 份样本，用于实验。Hanks 博士的试管数 M 很大，普通的计算机的基本数据类型无法存储这样大的 M 值，但万幸的是，M 总可以表示为 m_1 的 m_2 次方，即 $M = m_1^{m_2}$，其中 m_1，m_2 均为基本数据类型可以存储的正整数。

注意，整个实验过程中不允许分割单个细胞，比如某个时刻若培养皿中有 4 个细胞，Hanks 博士可以把它们分入 2 个试管，每试管内 2 个，然后开始实验。但如果培养皿中有 5 个细胞，博士就无法将它们均分入 2 个试管。此时，博士就只能等待一段时间，让细胞们继续分裂，使得其个数可以均分，或是干脆改换另一种细胞培养。

为了能让实验尽早开始，Hanks 博士在选定一种细胞开始培养后，总是在得到的细胞“刚好可以平均分入 M 个试管”时停止细胞培养并开始实验。现在博士希望知道，选择哪种细胞培养，可以使得实验的开始时间最早。

【输入】

输入文件名为 cell.in，共有三行。

第一行有一个正整数 N，代表细胞种数。

第二行有两个正整数 m_1，m_2，以一个空格隔开，$m_1^{m_2}$ 即表示试管的总数 M。

第三行有 N 个正整数，第 i 个数 S_i 表示第 i 种细胞经过 1 秒钟可以分裂成同种细胞的个数。

【输出】

输出文件 cell.out 共一行，为一个整数，表示从开始培养细胞到实验能够开始所经过的最少时间（单位为秒）。

如果无论 Hanks 博士选择哪种细胞都不能满足要求，则输出整数`-1`。

【输入输出样例 1】

cell.in	cell.out
1 2 1 3	-1

【输入输出样例 1 说明】

经过 1 秒钟，细胞分裂成 3 个，经过 2 秒钟，细胞分裂成 9 个，……，可以看出无论怎么分裂，细胞的个数都是奇数，因此永远不能分入 2 个试管。

【输入输出样例 2】

cell.in	cell.out
2 24 1 30 12	2

【输入输出样例 2 说明】

第 1 种细胞最早在 3 秒后才能均分入 24 个试管，而第 2 种最早在 2 秒后就可以均分（每试管 144/(24^1)=6 个）。故实验最早可以在 2 秒后开始。

【数据范围】

对于 50%的数据，有 ${m_1}^{m_2} \le 30000$。

对于所有的数据，有 $1 \le N \le 10000$，$1 \le m_1 \le 30000$，$1 \le m_2 \le 10000$，$1 \le S_i \le 2{,}000{,}000{,}000$。

4. 道路游戏

(game.pas/c/cpp)

【问题描述】

小新正在玩一个简单的电脑游戏。

游戏中有一条环形马路，马路上有 n 个机器人工厂，两个相邻机器人工厂之间由一小段马路连接。小新以某个机器人工厂为起点，按顺时针顺序依次将这 n 个机器人工厂编号为 1~n，因为马路是环形的，所以第 n 个机器人工厂和第 1 个机器人工厂是由一段马路连接在一起的。小新将连接机器人工厂的这 n 段马路也编号为 1~n，并规定第 i 段马路连接第 i 个机器人工厂和第 i+1 个机器人工厂（$1 \le i \le n-1$），第 n 段马路连接第 n 个机器人工厂和第 1 个机器人工厂。

游戏过程中，每个单位时间内，每段马路上都会出现一些金币，金币的数量会随着时间发生变化，即不同单位时间内同一段马路上出现的金币数量可能是不同的。小新需要机器人的帮助才能收集到马路上的金币。所需的机器人必须在机器人工厂用一些金币来购买，机器人一旦被购买，便会沿着环形马路按顺时针方向一直行走，在每个单位时间内行走一次，即从当前所在的机器人工厂到达相邻的下一个机器人工厂，并将经过的马路上的所有金币收集给小新，例如，小新在 i（$1 \le i \le n$）号机器人工厂购买了一个机器人，这个机器人会从 i 号机器人工厂开始，顺时针在马路上行走，第一次行走会经过 i 号马路，到达 i+1 号机器人工厂（如果 i=n，机器人会到达第 1 个机器人工厂），并将 i 号马路上的所有金币收集给小新。

游戏中，环形马路上不能同时存在 2 个或者 2 个以上的机器人，并且每个机器人最多能够在环形马路上行走 p 次。小新购买机器人的同时，需要给这个机器人设定行走次数，行走次数可以为 1~p 之间的任意整数。当马路上的机器人行走完规定的次数之后会自动消失，小新**必须立刻**在任意一个机器人工厂中购买一个新的机器人，并给新的机器人设定新的行走次数。

以下是游戏的一些补充说明：

1. 游戏从小新第一次购买机器人开始计时。
2. 购买机器人和设定机器人的行走次数是瞬间完成的，不需要花费时间。

3. 购买机器人和机器人行走是两个独立的过程，机器人行走时不能购买机器人，购买完机器人并且设定机器人行走次数之后机器人才能行走。
4. 在同一个机器人工厂购买机器人的花费是相同的，但是在不同机器人工厂购买机器人的花费不一定相同。
5. 购买机器人花费的金币，在游戏结束时再从小新收集的金币中扣除，所以在游戏过程中小新不用担心因金币不足，无法购买机器人而导致游戏无法进行。也因为如此，游戏结束后，收集的金币数量可能为负。

现在已知每段马路上每个单位时间内出现的金币数量和在每个机器人工厂购买机器人需要的花费，请你告诉小新，经过 m 个单位时间后，扣除购买机器人的花费，小新最多能收集到多少金币。

【输入】

输入文件名为 game.in。

第一行 3 个正整数，n，m，p，意义如题目所述。

接下来的 n 行，每行有 m 个正整数，每两个整数之间用一个空格隔开，其中第 i 行描述了 i 号马路上每个单位时间内出现的金币数量（1≤金币数量≤100），即第 i 行的第 j（1≤j≤m）个数表示第 j 个单位时间内 i 号马路上出现的金币数量。

最后一行，有 n 个整数，每两个整数之间用一个空格隔开，其中第 i 个数表示在 i 号机器人工厂购买机器人需要花费的金币数量（1≤金币数量≤100）。

【输出】

输出文件 game.out 共一行，包含 1 个整数，表示在 m 个单位时间内，扣除购买机器人花费的金币之后，小新最多能收集到多少金币。

【输入输出样例】

game.in	game.out
2 3 2 1 2 3 2 3 4 1 2	5

【数据范围】

对于 40%的数据，2≤n≤40，1≤m≤40。

对于 90%的数据，2≤n≤200，1≤m≤200。

对于 100%的数据，2≤n≤1000，1≤m≤1000，1≤p≤m。