

Contest 1

吴清月

October 1, 2020

中文题目	战术班	应急棍	擒敌拳
英文题目	Gun	Stick	Fist
程序名	gun.c/cpp	stick.c/cpp	fist.c/cpp
输入文件	gun.in	stick.in	fist.in
输出文件	gun.out	stick.out	fist.out
每个测试点时限	1.0 s	1.0 s	1.0 s
测试点数目	10	10	10
每个测试点分值	10	10	10
附加样例文件	有	有	有
题目类型	传统	传统	传统
运行内存上限	512M	512M	512M

注意事项:

1. 无特别声明时, 结果比较方式为逐行比较模式 (忽略多余空格和制表符).
2. C++ 语言源程序名称为题目名.cpp, C 语言源程序名称为题目名.c.
3. 编译命令为 `g++ -o 题目名 题目名.* -lm -Wl,--stack=512000000 -O2 -std=c++11` 或 `gcc -o 题目名 题目名.* -lm -Wl,--stack=512000000 -O2 -std=c11`. 即评测时开启 **O2** 优化开关. 开启无限栈. 开启 **C++11/C11**. 由于 **11** 特性导致的编译错误可以申请重测.
4. 文件名 (程序名和输入输出文件名) 必须使用英文小写.
5. 函数 `main()` 的返回值类型必须是 `int`, 程序正常结束时返回值必须是 `0`.
6. 评测在 Windows 10 环境下用 Lemon 进行, 对于 `long long` 类型使用 `%lld` 和 `%I64d` 都可以接受. 实际评测时限可能更改.

Problem A. 战术班 (gun.c/cpp)

Input file: gun.in
Output file: gun.out
Time Limit : 1.0 second
Memory Limit: 512 megabytes

今天是新生军训的第四天，成功入选特战连的小 H 同学进入了特战连的战术班。

战术班共有 n 个人，初始时他们被划分到了 m 个战略小组中，第 i 个小组里有 a_i 个人。每一次，教官会选择一个战略小组，并且将这个小组分成两个小组，要求这两个新的小组人数不能相同。

现在小 H 同学想知道，划分小组的过程最多能进行多少次，最少能进行多少次。

Input

第一行两个整数 n, m 。

接下来一行 m 个整数 a_1, a_2, \dots, a_m ，表示每一个小组中的人数。数据保证 $\sum a_i = n$ 。

Output

一行两个整数，分别表示最多能进行的次数和最少能进行的次数。

Examples

gun-sample0.in	gun-sample0.ans
5 1 5	3 2

Notes

数据规模与约定：

对于 20% 的数据，保证 $n \leq 10$ ；

对于另外 20% 的数据，保证 $m = 1$ 且 n 为奇数；

对于另外 30% 的数据，保证 $m = 1$ ；

对于 100% 的数据，保证 $1 \leq n \leq 1000000, 1 \leq m \leq n$ 。

Problem B. 应急棍 (stick.c/cpp)

Input file: `stick.in`
 Output file: `stick.out`
 Time Limit : 1.0 seconds
 Memory Limit: 512 megabytes

今天是新生军训的第五天，成功入选特战连的小 W 同学进入了特战连的应急棍方阵。

由于应急棍术需要的空间很大，因此所有人都想离其它同学尽可能远，避免误伤。

应急棍的场地可以描述为一个边长为 l 的正方形，左下角坐标为 $(0,0)$ ，右上角坐标为 (l,l) 。同学们会依次进入这个场地选择位置。所有人都会遵循如下三个原则：

- 离自己最近的同学距离尽可能远。
- 若有多个位置满足条件，则选择横坐标最小的位置。
- 若仍有多个位置满足条件，则选择纵坐标最小的位置。

注意一旦选好位置就不能再动了。（即使当前位置已经不是最优的）

现在小 W 同学想知道，第 n 个同学的坐标是多少。

Input

第一行两个整数 C, T ，其中 C 是要求的精度， T 是测试组数。

接下来一行一个实数 l ，表示正方形的边长。

接下来 T 行，每一行一个整数 n ，表示询问第 n 个同学的坐标是多少。

Output

对于每一个询问，输出一行两个实数 x, y ，表示第 n 个同学的坐标。

你的答案被认为是正确的，当且仅当与答案的绝对误差或相对误差不超过 10^{-C} 。即，若标准答案为 x_0, y_0 ，则你的答案被认为正确，当且仅当 $\max\left(\frac{|x-x_0|}{\max(x_0,1)}, \frac{|y-y_0|}{\max(y_0,1)}\right) \leq 10^{-C}$ 。

Examples

stick-sample0.in	stick-sample0.ans
10 12	0 0
8	8 8
1	0 8
2	8 0
3	4 4
4	0 4
5	4 0
6	4 8
7	8 4
8	2 2
9	2 6
10	6 2
11	
12	

Notes

数据规模与约定：

对于 20% 的数据，保证 $1 \leq n \leq 4$ ；

对于 40% 的数据，保证 $1 \leq n \leq 1000$ ；

对于 60% 的数据，保证 $1 \leq n \leq 100000$ ；

对于 60% 的数据，保证 $C \leq 9, l \leq 10^9$ 且 l 为整数；

对于 80% 的数据，保证 $1 \leq n \leq 10^9$ ；

对于 100% 的数据，保证 $1 \leq T \leq 100, C \leq 100, 1 \leq n \leq 10^{18}, l \leq 10^{50}$ 且 l 最多包含 50 位小数。

Problem C. 擒敌拳 (fist.c/cpp)

Input file: fist.in
Output file: fist.out
Time Limit : 1.0 seconds
Memory Limit: 512 megabytes

别折磨我了，我真的编不动题目背景了！

有一列 n 个矩形并排放置在一起，第 i 的矩形的宽度是 1，高度是 h_i 。

现在你需要对于每一个 i ，回答在前 i 个矩形中能够画出的最大的矩形的面积是多少。矩形的边必须是水平或竖直。

Input

第一行一个整数 n 。

接下来一行 n 个整数 h_1, h_2, \dots, h_n 。

Output

输出一行 n 个整数，表示所求答案。

Examples

fist-sample0.in	fist-sample0.ans
5 1 7 2 3 9	1 7 7 7 9

fist-sample1.in	fist-sample1.ans
5 4 5 3 8 7	4 8 9 12 15

Notes

数据规模与约定：

对于 20% 的数据，保证 $n \leq 100$ ；

对于 40% 的数据，保证 $n \leq 3000$ ；

对于另外 30% 的数据，保证 h_i 单调不降；

对于 100% 的数据，保证 $1 \leq n \leq 200000, 1 \leq h_i \leq 10^7$ 。