

## NOIP 提高班——数学专项练习

(请选手务必仔细阅读本页内容)

## 一、题目概况

中文题目名称	哈希函数	向量	仪仗队
英文题目与子目录名	hash	vector	honourguard
可执行文件名	hash	vector	honourguard
输入文件名	hash.in	vector.in	honourguard.in
输出文件名	hash.out	vector.out	honourguard.out
每个测试点时限	1 秒	1 秒	2 秒
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
结果比较方式	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
运行内存上限	256M	256M	1G

## 二、提交源程序文件名

对于 C++ 语言	hash.cpp	vector.cpp	honourguard.cpp
对于 C 语言	hash.c	vector.c	honourguard.c
对于 Pascal 语言	hash.pas	vector.pas	honourguard.pas

## 三、编译命令（不包含任何优化开关）

对于 C++ 语言	g++ -o hash hash.cpp -lm	g++ -o vector vector.cpp -lm	g++ -o honourguard honourguard.cpp -lm
对于 C 语言	gcc -o hash hash.c -lm	gcc -o vector vector.c -lm	gcc -o honourguard honourguard.c -lm
对于 Pascal 语言	fpc hash.pas	fpc vector.pas	fpc honourguard.pas

## 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. 只提供 Linux 格式附加样例文件。

## 1. 哈希函数

(hash.cpp/c/pas)

### 【问题描述】

明明觉得 hash 是个好算法, 代码短、效率高. 某天, 他碰到了一个求正方形个数的问题, 于是很淡定地枚举对角线, 然后用 hash 判存在, 妥妥的搞定, 但是提交后却 wa 了几个点. 仔细观察其 hash 函数为:  $h = xy + x + y$ . 为了让明明知道这个函数存在什么问题, 对于给出一个  $h$  值, 请你来告诉他有多少对  $(x, y)$  满足上述式子 ( $\max\{x, y\} \leq h$ ;  $h, x, y$  都为非负整数)?

### 【输入格式】

输入文件名为 hash.in.

输入文件第一行包含一个正整数  $T$ , 表示数据组数.

接下来  $T$  行每行包含一个整数  $h$ , 意义详见题目描述.

### 【输出格式】

输出文件名为 hash.out.

输出文件包含  $T$  行, 每行一个整数, 分别表示有多少组  $(x, y)$  满足要其对应的  $h$  值.

### 【输入输出样例 1】

hash.in	hash.out
3	2
1	3
3	2
4	

### 【输入输出样例 1 说明】

第一组:  $(1, 0), (0, 1)$ .

第二组:  $(0, 3), (1, 1), (3, 0)$ .

第三组:  $(4, 0), (0, 4)$ .

### 【输入输出样例 2】

见选手目录下的 hash/hash2.in 和 hash/hash2.ans.

### 【数据规模与约定】

对于 30% 的数据,  $h \leq 2,000, T \leq 1,000$ .

对于 70% 的数据,  $h \leq 10,000, T \leq 10,000$ .

对于 100% 的数据,  $h \leq 10^8, T \leq 10,000$ .

## 2. 向量

(vector.cpp/c/pas)

### 【问题描述】

给你一对数  $a, b$ , 你可以任意使用  $(a, b), (a, -b), (-a, b), (-a, -b), (b, a), (b, -a), (-b, a), (-b, -a)$  这些向量, 问你拼出另一个向量  $(x, y)$ .

说明: 这里的拼就是使得你选出的向量之和为  $(x, y)$ .

### 【输入格式】

输入文件名为 `vector.in`.

输入文件第一行包含一个正整数  $T$ , 为数据的组数.

接下来  $T$  行每行包含四个整数  $a, b, x, y$ .

### 【输出格式】

输出文件名为 `vector.out`.

输出文件包含  $T$  行, 每行为“Y”或“N”, 分别表示可以拼出来、不能拼出来.

### 【输入输出样例 1】

vector.in	vector.out
3	Y
2 1 3 3	N
1 1 0 1	Y
1 0 -2 3	

### 【输入输出样例 1 说明】

第一组:  $(2, 1) + (1, 2) = (3, 3)$ .

第二组不能拼成.

第三组:  $(-1, 0) + (-1, 0) + (0, 1) + (0, 1) + (0, 1) = (-2, 3)$ .

### 【输入输出样例 2】

见选手目录下的 `vector/vector2.in` 和 `vector/vector2.ans`.

### 【数据规模与约定】

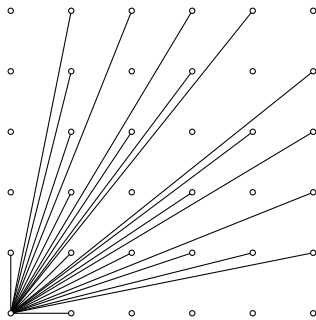
对于 100% 的数据,  $T \leq 500,000$ ,  $-2 \times 10^9 \leq a, b, x, y \leq 2 \times 10^9$ .

3. 仪仗队

(honourguard.cpp/c/pas)

【问题描述】

作为体育委员,C 君负责这次运动会仪仗队的训练. 仪仗队是由学生组成的  $N \times N$  的方阵, 为了保证队伍在行进中整齐划一, C 君会跟在仪仗队的左后方, 根据其视线所及的学生人数来判断队伍是否整齐 (如下图). 现在, C 君希望你告诉他队伍整齐时能看到的学生人数.



【输入格式】

输入文件名为 `honourguard.in`.

输入数据仅一行, 包含一个正整数  $N$ , 表示方阵的边长.

【输出格式】

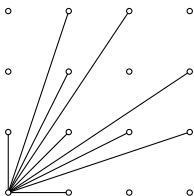
输出文件名为 `honourguard.out`.

输出文件仅一行, 包含一个正整数  $M$ , 为 C 君应看到的学生人数.

【输入输出样例 1】

<code>honourguard.in</code>	<code>honourguard.out</code>
4	9

【输入输出样例 1 说明】



【输入输出样例 2】

见选手目录下的 `honourguard/honourguard2.in` 和 `honourguard/honourguard2.ans`.

【数据规模与约定】

对于 30% 的数据,  $1 \leq N \leq 1,000$ .

对于 70% 的数据,  $1 \leq N \leq 40,000$ .

对于 100% 的数据,  $1 \leq N \leq 40,000,000$ .