

# NOIp Senior Simulated Contest

zhzh2001

verified by axs7384

题目名称	三角函数	牛奶测量	质数哈希
目录	trifunc	measurement	primehash
可执行文件名	trifunc	measurement	primehash
输入文件名	trifunc.in	measurement.in	primehash.in
输出文件名	trifunc.out	measurement.out	primehash.out
时间限制	1s	1-2s	0.5s
空间限制	256MB	256MB	256MB
测试点数量	20	20	25
测试点分数	5	5	4
比较方式	全文	全文	SPJ
部分分	无	无	有

提交源程序文件名

对于 C++ 语言	trifunc.cpp	measurement.cpp	primehash.cpp
对于 C 语言	trifunc.c	measurement.c	primehash.c
对于 Pascal 语言	trifunc.pas	measurement.pas	primehash.pas

编译选项

对于 C++ 语言	-O2 -std=gnu++11	-O2 -std=gnu++11	-O2 -std=gnu++11
对于 C 语言	-O2	-O2	-O2
对于 Pascal 语言	-O2	-O2	-O2

注意事项:

1. 注意编译选项，避免未定义行为或编译错误。
2. 代码长度限制为**25KB**。
3. 注意代码常数和 I/O 造成的效率影响。

# 1 三角函数 (trifunc.c/cpp/pas)

## 1.1 题目描述

已知

$$\cos \alpha = \frac{a}{b}, \alpha \in [0, \pi]$$

求

$$\sum_{i=1}^n b^i \times \sin i\alpha$$

四舍五入 后对  $10^9 + 7$  取模的值。

## 1.2 输入格式 (trifunc.in)

一行三个整数  $a, b, n$ 。

## 1.3 输出格式 (trifunc.out)

一行一个非负 整数，表示答案四舍五入后对  $10^9 + 7$  取模的值。

## 1.4 样例

### 1.4.1 样例输入 1

4 5 3

### 1.4.2 样例输出 1

144

### 1.4.3 样例解释 1

$$\alpha = \arccos \frac{4}{5} \approx 0.64350110879328426$$

$i$	$\sin i\alpha$	$b^i \times \sin i\alpha$
1	0.6	3
2	0.96	24
3	0.936	117

### 1.4.4 样例 2

见选手目录下的trifunc/trifunc2.in 与trifunc/trifunc2.ans。

## 1.5 数据范围和约定

测试点	$a$	$b$	$n$
1	$= 0$		$\leq 10^6$
2			$\leq 10^{18}$
3			$\leq 10^{1,000}$
4			
5	$= 4$	$= 5$	$\leq 20$
6	$= 5$	$= 13$	$\leq 12$
7	$= 7$	$= 25$	
8		$\leq 15$	
9		$\leq 100$	
10		$\leq 10^6$	
11			
12		$\leq 10^9$	
13		$\leq 10^{18}$	
14			
15		$\leq 10^{1,000}$	
16		$\leq 10^{20,000}$	
17		$\leq 10^{50,000}$	
18		$\leq 10^{200,000}$	
19			
20			

对于 100% 的数据,  $0 \leq |a| \leq b \leq 10^9, b > 0, 1 \leq n \leq 10^{500,000}, \sin \alpha \in \mathbb{Q}$ 。

## 1.6 提示

在 Pascal/C/C++ 中, 三角函数均采用弧度制, 转换关系为

$$\pi = 180^\circ$$

$\arcsin$  和  $\arccos$  为  $\sin$  和  $\cos$  的反函数, 定义域为  $[-1, 1]$ 。

在 Pascal 中,  $\sin$  和  $\cos$  为标准函数, 而  $\arcsin$  和  $\arccos$  需要使用 `math` 库。

在 C/C++ 中,  $\arcsin$  为 `asin`,  $\arccos$  为 `acos`, 这些函数都定义在 `math.h/cmath` 中。

## 2 牛奶测量 (measurement.c/cpp/pas)

### 2.1 题目描述

最初，农夫约翰的每头奶牛每天生产  $G$  加仑的牛奶。由于随着时间的推移，奶牛的产奶量可能会发生变化，农夫约翰决定定期对奶牛的产奶量进行测量，并将其记录在日志中。

他的日志中的记录如下：

```
35 1234 -2
```

```
14 2345 +3
```

第一个条目表明：在第 35 天，1234 号奶牛的产奶量比上次测量时降低了 2 加仑。

第二个条目表明：在第 14 天，2345 号奶牛的产奶量比上次测量时增加了 3 加仑。

农夫约翰在任何一天内最多只有做一次测量的时间。不幸的是，约翰有点杂乱无章，他不一定按照时间顺序记下测量结果。为了保持奶牛的产奶动力，农夫约翰自豪地在谷仓的墙上展示了目前产奶量最高的奶牛的照片（如果有若干头奶牛的产奶量最高，他就会展示所有的照片）。请求出哪些天约翰需要调整所展示的照片。

请注意，农夫约翰有一大群奶牛。所以尽管日志中记录了一些奶牛改变了产奶量，但仍然还有很多奶牛的产奶量保持在  $G$  加仑。

### 2.2 输入格式 (measurement.in)

第一行包含两个整数  $N$  和  $G$ ，分别表示日志条数和初始产奶量。

接下来  $N$  行，每行为一次测量，按照上面的格式，分别表示日期、奶牛编号和产量的变化值。

### 2.3 输出格式 (measurement.out)

第一行一个整数  $K$  表示调整照片的次数。

接下来  $K$  行，每行一个整数，表示调整照片的日期，按照升序输出。

### 2.4 样例

#### 2.4.1 样例输入 1

```
4 10
```

```
7 3 +3
```

```
4 2 -1
```

```
9 3 -1
```

```
1 1 +2
```

### 2.4.2 样例输出 1

3

1

7

9

### 2.4.3 样例解释 1

在第 1 天, 1 号奶牛的产量提高为 12 加仑, 最高产量变为 1 号。

在第 4 天, 2 号奶牛的产量降低为 9 加仑, 最高产量仍为 1 号。

在第 7 天, 3 号奶牛的产量提高为 13 加仑, 最高产量变为 3 号。

在第 9 天, 3 号奶牛的产量降低为 12 加仑, 最高产量变为 1、3 号并列。

### 2.4.4 样例 2

见选手目录下的measurement/measurement2.in 与measurement/measurement2.ans。

## 2.5 数据范围和约定

测试点	$N \leq$	特殊性质
1		每次测量同一只奶牛
2		每次测量的奶牛都不同
3		产量只增加
4		产量只减少
5		产量除一次减少外都增加
6	40	
7	750	
8	2,000	
9	6,000	
10	10,000	
11	30,000	
12		
13	50,000	
14	70,000	
15	90,000	
16	100,000	
17	200,000	
18		
19		
20		

对于 100% 的数据,  $1 \leq N \leq 300,000$ , 日期、奶牛编号在  $1 \dots 10^9$  的范围内, 日期互不相同, 产量总在  $0 \dots 10^9$  范围内。

## 3 质数哈希 (primehash.c/cpp/pas)

### 3.1 题目描述

求整数区间  $l \dots r$  间的质数表的哈希值。

对于有序数列  $A[1 \dots n]$ , 定义哈希函数为

$$h(A) = \sum_{i=1}^n A_i \times p^{n-i} \pmod{(10^9 + 7)}$$

其中  $p = 999,999,937$ 。

### 3.2 输入格式 (primehash.in)

一行两个整数  $l, r$ 。

### 3.3 输出格式 (primehash.out)

两个整数, 分别表示质数个数和哈希值。

### 3.4 样例

#### 3.4.1 样例输入 1

10 15

#### 3.4.2 样例输出 1

2 999999250

#### 3.4.3 样例解释 1

$$A = \{11, 13\}, h(A) = (11 \times 999,999,937 + 13) \pmod{1,000,000,007} = 999,999,250$$

#### 3.4.4 样例 2

见选手目录下的primehash/primehash2.in 与primehash/primehash2.ans。

### 3.5 数据范围和约定

测试点	$r - l \leq$	$r \leq$	
1	0		
2	1,000		
3	$10^5$		
4	$3 \times 10^5$		
5	$10^6$		
6	$3 \times 10^6$		
7	$10^7$		
8	$3 \times 10^7$		
9	$10^8$		
10		1,000	
11		$10^6$	
12		$3 \times 10^6$	
13		$10^7$	
14		$3 \times 10^7$	
15		$10^8$	
16		$3 \times 10^8$	
17		$10^9$	
18		$3 \times 10^9$	
19		$10^{10}$	
20		$3 \times 10^{10}$	
21		$5 \times 10^{10}$	
22		$8 \times 10^{10}$	
23			$10^{11}$
24			
25			

对于 100% 的数据,  $1 \leq l \leq r \leq 10^{11}$ 。

### 3.6 部分分

如果质数个数正确, 而哈希值错误, 可以得到测试点 75% 的分数, 即 3 分。