

# NOI2018 北京队选拔赛

## BJOI 2018

**竞赛时间：2018 年 4 月 15 日 8:00–13:00**

题目名称	双人猜数游戏	链上二次求和	治疗之雨
英文名称	guess	sum	heal
目录	guess	sum	heal
可执行文件名	guess	sum	heal
输入文件名	guess*.in	sum.in	heal.in
输出文件名	guess*.out	sum.out	heal.out
每个测试点时限	N/A	7 秒	4 秒
内存限制	N/A	512 MB	512 MB
试题总分	100	100	100
测试点数目	25	20	10
每个测试点分值	4	5	10
是否有部分分	否	否	否
题目类型	提交答案型	传统型	传统型

提交源程序须加后缀

对于 C++ 语言	N/A	sum.cpp	heal.cpp
-----------	-----	---------	----------

编译选项

对于 C++ 语言	N/A	-O2 -Wl, --stack=536870912	-O2 -Wl, --stack=536870912
-----------	-----	-------------------------------	-------------------------------

**注意：最终测试时，所有编译命令均打开 -O2 优化开关。**

**注意：请勿建立子文件夹，即请将第一题的 25 个答案文件和后两题的源代码共 27 个文件放在同一个文件夹内。**

## 双人猜数游戏

### 【问题描述】

Alice 和 Bob 是一对非常聪明的人，他们可以算出各种各样的游戏的最优策略。现在有个综艺节目《最强大佬》请他们来玩一个游戏。主持人写了三个正整数  $s$ 、 $m$ 、 $n$ ，然后一起告诉 Alice 和 Bob  $s \leq m \leq n$  以及  $s$  是多少。（即， $s$  是接下来要猜的  $m$ 、 $n$  的下限。）之后主持人单独告诉 Alice  $m$  与  $n$  的乘积是多少，单独告诉 Bob  $m$  与  $n$  的和是多少。

当然，如果一个人同时知道  $m$  与  $n$  的乘积以及  $m$  与  $n$  的和的话就能很容易地算出  $m$  和  $n$  分别是多少，但是现在 Alice 和 Bob 只分别知道其中一个，而且他们只能回答主持人的问题，不能交流。主持人从 Alice 或 Bob（见输入）开始依次询问 Alice/Bob 知不知道  $m$  和  $n$  分别是多少，Alice/Bob 只能回答知道/不知道。

为了节目效果，为了显示出 Alice 和 Bob 非常聪明，主持人希望 Alice 和 Bob 一共说了  $t$  次“不知道”以后两个人都知道  $m$  和  $n$  是多少了。现在主持人找到了你，希望你帮他构造一组符合条件的  $m$  和  $n$ 。

### 【输入格式】

输入文件 *guess\*.in* 共一行，格式为 “ $s$  <name>  $t$ ”（不含引号），其中  $s$  与  $t$  的定义见题目描述（注意 Alice 和 Bob 知道  $s$  是多少），<name> 为 “Alice” 或 “Bob”（不含引号），表示主持人第一次问的人。

### 【输出格式】

输出文件 *guess\*.out* 共一行两个数，以一个空格隔开，表示一组满足要求的  $m$  和  $n$ 。若有多组解，输出  $m$  与  $n$  的和最小的那组解。若仍有多组解，输出  $m$  与  $n$  的和最小的解中  $m$  最小的那组解。输入数据保证有解。

### 【样例输入】

5 Bob 2

### 【样例输出】

6 10

**【样例说明】**

主持人告诉 Alice 和 Bob  $5 \leq m \leq n$ ，单独告诉 Alice  $mn = 60$ ，单独告诉 Bob  $m + n = 16$ 。

主持人问 Bob 知不知道  $m$  和  $n$  分别是多少，Bob 说不知道。

主持人问 Alice 知不知道  $m$  和  $n$  分别是多少，Alice 说不知道。

主持人问 Bob 知不知道  $m$  和  $n$  分别是多少，Bob 说知道。

主持人问 Alice 知不知道  $m$  和  $n$  分别是多少，Alice 说知道。

**【样例输入】**

```
2 Alice 3
```

**【样例输出】**

```
4 4
```

**【样例说明】**

主持人告诉 Alice 和 Bob  $2 \leq m \leq n$ ，单独告诉 Alice  $mn = 16$ ，单独告诉 Bob  $m + n = 8$ 。

主持人问 Alice 知不知道  $m$  和  $n$  分别是多少，Alice 说不知道。

主持人问 Bob 知不知道  $m$  和  $n$  分别是多少，Bob 说不知道。

主持人问 Alice 知不知道  $m$  和  $n$  分别是多少，Alice 说不知道。

主持人问 Bob 知不知道  $m$  和  $n$  分别是多少，Bob 说知道。

主持人问 Alice 知不知道  $m$  和  $n$  分别是多少，Alice 说知道。

~~Alice 和 Bob 分别单独写出了正确的  $m$  和  $n$ ，观众们觉得 Alice 和 Bob 很厉害。~~

**【数据规模与约定】**

对于 40% 的数据， $t = 2$ 。

对于 100% 的数据， $1 \leq s \leq 200$ ， $2 \leq t \leq 15$ ，输入数据保证有解。

## 链上二次求和

### 【问题描述】

有一条长度为  $n$  的链 ( $\forall 1 \leq i < n$ , 点  $i$  与点  $i+1$  之间有一条边的无向图), 每个点有一个整数权值, 第  $i$  个点的权值是  $a_i$ 。现在有  $m$  个操作, 每个操作如下:

操作 1 (修改): 给定链上两个节点  $u$ 、 $v$  和一个整数  $d$ , 表示将链上  $u$  到  $v$  唯一的简单路径上每个点的权值都加上  $d$ 。

操作 2 (询问): 给定两个正整数  $l$ 、 $r$ , 表示求链上所有节点个数大于等于  $l$  且小于等于  $r$  的简单路径的节点权值和之和。由于答案很大, 只用输出对质数  $1000000007$  取模的结果即可。

一条节点个数为  $k$  的简单路径的节点权值和为这条路径上所有  $k$  个节点 (包括端点) 的权值之和, 而本题中要求的是对所有满足要求的简单路径, 求这一权值和的和。

由于是无向图, 路径也是无向的, 即点 1 到点 2 的路径与点 2 到点 1 的路径是同一条, 不要重复计算。

### 【输入格式】

从文件 *sum.in* 中输入数据。

输入第一行包含两个正整数  $n$ 、 $m$ , 分别表示节点个数和操作次数。

第二行包含  $n$  个整数, 其中第  $i$  个数  $a_i$  为第  $i$  个点的初始权值。

接下来  $m$  行, 每行为 “1  $u$   $v$   $d$ ” 或 “2  $l$   $r$ ” 的形式 (不含引号), 分别表示进行一次操作 1 (修改) 或操作 2 (询问)。

### 【输出格式】

输出到文件 *sum.out* 中。

对于每次询问, 输出一行一个整数, 表示答案对  $1000000007$  取模的余数。

### 【样例输入】

```
5 5
1 1 1 1 1
2 5 5
2 1 2
```

```
1 1 2 2
2 1 1
1 1 5 3
```

**【样例输出】**

```
5
13
9
```

**【样例说明】**

节点个数为 5 的简单路径只有 1 条，权值和为 5，故第 1 次询问输出 5。

节点个数为 1 的简单路径有 5 条，每条权值和都是 1；节点个数为 2 的简单路径有 4 条，每条权值和都是 2，故第 2 次询问输出 13。

在将点 1 和点 2 的权值加 2 后，5 条节点个数为 1 的简单路径的权值和分别为 3、3、1、1、1，故第 3 次询问输出 9。

**【数据规模与约定】**

记操作 1（修改）的次数为  $m'$ 。

对于全部数据，保证  $n \leq 200000$ ,  $m \leq 500000$ ,  $m' \leq 100000$ ,  $0 \leq a_i < 1000000007$ ,  $1 \leq u \leq n$ ,  $1 \leq v \leq n$ ,  $0 \leq d < 1000000007$ ,  $1 \leq l \leq r \leq n$ 。

对于每个数据点的详细规模与约定见下表。（见下页）

数据点编号	$n \leq$	$m \leq$	$m' \leq$	约束
1	50	50	50	无
2				
3	300	300	300	
4				
5	5000	5000	5000	
6		500000		
7				
8		100000		
9	200000	1	0	保证 $l=1, r=n$
10		500000	100000	无
11				保证 $u=v$
12				保证 $l=1, r=n$
13				保证 $u=1, v=n, ai=0$
14				保证 $u=1, v=n$
15				保证 $d=1, ai=0$
16				保证 $d=1$
17				保证 $ai=0$
18				无
19				
20		无		

## 治疗之雨

### 【背景描述】

（没玩过《炉石传说》的人可以跳过这一段）今天我们来探讨一下《炉石传说》中“治疗之雨”（恢复 12 点生命值，随机分配到所有友方角色上）和“暗影打击装甲”（每当一个角色获得治疗，便对一个随机敌人造成 1 点伤害）这两张卡牌之间的互动效果。假设你的场上有  $m$  个剩余生命值无限大且生命值上限减去剩余生命值也无限大的随从，而对方的场上有  $k$  个暗影打击装甲，你的英雄剩余生命值为  $p$ 、生命值上限为  $n$ ，现在你使用了一张可以恢复无限多（而不是 12 点）生命值的“治疗之雨”，问“治疗之雨”期望总共恢复了几点生命值以后你的英雄会死亡（生命值降为 0；“治疗之雨”的判定机制使得在此之后再也不会为英雄恢复生命值）。

下面让我们再形式化地描述一遍问题。

### 【问题描述】

你现在有  $m+1$  个数：第一个为  $p$ ，最小值为 0，最大值为  $n$ ；剩下  $m$  个都是无穷，没有最小值或最大值。你可以进行任意多轮操作，每轮操作如下：

在不为最大值的数中等概率随机选择一个（如果没有则不操作），把它加一；

进行  $k$  次这个步骤：在不为最小值的数中等概率随机选择一个（如果没有则不操作），把它减一。

现在问期望进行多少轮操作以后第一个数会变为最小值 0。

### 【输入格式】

从文件 *heal.in* 中输入数据。

输入包含多组数据。

输入第一行包含一个正整数  $T$ ，表示数据组数。

接下来  $T$  行，每行 4 个非负整数  $n$ 、 $p$ 、 $m$ 、 $k$ （含义见题目描述），表示一次询问。

### 【输出格式】

输出到文件 *heal.out* 中。

输出  $T$  行，每行一个整数，表示一次询问的答案。

如果无论进行多少轮操作，第一个数都不会变为最小值  $0$ ，那么输出 “-1”（不含引号）；否则，可以证明，答案一定为有理数，那么请输出答案模  $1000000007$  的余数，即，设答案为  $a/b$ （ $a$ 、 $b$  为互质的正整数），你输出的整数为  $x$ ，那么你需要保证  $0 \leq x < 1000000007$  且  $a$  与  $bx$  模  $1000000007$  同余。

#### 【样例输入】

```
2
2 1 1 1
2 2 1 1
```

#### 【样例输出】

```
6
8
```

#### 【数据规模与约定】

对于 10% 的数据， $n \leq 3$ ， $m, k \leq 2$ 。

对于 20% 的数据， $n, m, k \leq 5$ 。

对于 30% 的数据， $n, m, k \leq 30$ 。

对于 40% 的数据， $n, m, k \leq 50$ 。

对于 50% 的数据， $n, m, k \leq 200$ 。

对于 70% 的数据， $n \leq 200$ 。

对于 80% 的数据， $n \leq 500$ 。

对于 100% 的数据， $1 \leq T \leq 100$ ， $1 \leq p \leq n \leq 1500$ ， $0 \leq m, k \leq 1000000000$ 。