

# A 星球

时空限制：1s 256MB

文件名

svemir.in/svemir.out/svemir.cpp

## 题目描述

太空帝国要通过建造隧道来联通它的  $N$  个星球。

每个星球用三维坐标  $(x_i, y_i, z_i)$  来表示，而在两个星球  $A, B$  之间建造隧道的价格为  $\min\{|x_A - x_B|, |y_A - y_B|, |z_A - z_B|\}$ 。

现要建造  $N - 1$  条隧道使得所有的星球都能直接或间接相连。求完成该任务所需的最小总价。

## 输入格式

第一行，一个整数  $N$ 。

接下来的  $N$  行，每行三个整数  $x_i, y_i, z_i$ ，表示第  $i$  个星球的坐标。

数据保证不存在两个具有相同坐标的星球。

## 输出格式

输出所需的最小总价。

## 样例 #1

### 样例输入 #1

```
2
1 5 10
7 8 2
```

### 样例输出 #1

```
3
```

## 样例 #2

## 样例输入 #2

```
3
-1 -1 -1
5 5 5
10 10 10
```

## 样例输出 #2

```
11
```

## 样例 #3

### 样例输入 #3

```
5
11 -15 -15
14 -5 -15
-1 -1 -5
10 -4 -1
19 -4 19
```

### 样例输出 #3

```
4
```

## 提示

**【数据规模与约定】**

- 对于 50% 的数据， $1 \leq N \leq 10^3$ 。
- 对于 100% 的数据， $1 \leq N \leq 10^5$ ， $-10^9 \leq x_i, y_i, z_i \leq 10^9$ 。

# B 农场改造

时空限制：1s 256MB

## 文件名

updates.in/updates.out/updates.cpp

## 题目描述

Farmer John 经营着总共  $N$  个农场（ $1 \leq N \leq 10^5$ ），编号为  $1 \dots N$ 。最初，这些农场之间没有道路连接，并且每个农场都在活跃地生产牛奶。

由于经济的动态性，Farmer John 需要根据  $Q$  次更新操作（ $0 \leq Q \leq 2 \cdot 10^5$ ）对他的农场进行改造。更新操作有三种可能的形式：

- (D  $x$ ) 停用一个活跃的农场  $x$ ，使其不再生产牛奶。
- (A  $x$   $y$ ) 在两个活跃的农场  $x$  和  $y$  之间添加一条道路。
- (R  $e$ ) 删除之前添加的第  $e$  条道路（ $e = 1$  是添加的第一条道路）。

一个农场  $x$  如果正在活跃地生产牛奶，或者可以通过一系列道路到达另一个活跃的农场，则称之为一个「有关的」农场。对于每个农场  $x$ ，计算最大的  $i$ （ $0 \leq i \leq Q$ ），使得农场  $x$  在第  $i$  次更新后是有关的。

## 输入格式

输入的第一行包含  $N$  和  $Q$ 。以下  $Q$  行每行包含如下格式之一的一次更新操作：

```
D x
A x y
R e
```

输入保证对于 R 类更新， $e$  不超过已经添加的道路的数量，并且没有两次 R 类更新具有相等的  $e$  值。

## 输出格式

输出  $Q$  行，每行包含一个  $0 \dots Q$  范围内的整数。

## 样例 #1

### 样例输入 #1

```
5 9
A 1 2
A 2 3
D 1
D 3
A 2 4
D 2
R 2
R 1
R 3
```

## 样例输出 #1

```
7
8
6
9
9
```

## 提示

### 样例解释

在这个例子中，道路以顺序  $(2, 3)$ ,  $(1, 2)$ ,  $(2, 4)$  被删除。

- 农场 1 在道路  $(1, 2)$  被删除之前是有关的。
- 农场 2 在道路  $(2, 4)$  被删除之前是有关的。
- 农场 3 在道路  $(2, 3)$  被删除之前是有关的。
- 农场 4 和 5 在所有更新结束后仍然是活跃的。所以它们一直保持为有关的，两者的输出均应为  $Q$ 。

### 数据范围

- 测试点 2-5 满足  $N \leq 10^3$ ， $Q \leq 2 \cdot 10^3$ 。
- 测试点 6-20 没有额外限制。

# C 饥饿的奶牛

时空限制：1s 256MB

## 文件名

hunger.in/hunger.out/hunger.cpp

## 题目描述

Farmer John 的草地里的草在一场大旱中都干死了。经过数小时的绝望和沉思，FJ 想到了一个绝妙的主意，购买玉米来喂养他宝贵的奶牛。

FJ 的  $N$  头奶牛 ( $1 \leq N \leq 100$ ) 排成一行，队伍中的第  $i$  头奶牛的饥饿度为一个非负整数  $h_i$ 。由于 FJ 的奶牛是社会性动物，她们坚持一起进食，FJ 降低奶牛饥饿度的唯一方法是选择两头相邻的奶牛  $i$  和  $i + 1$  并分别喂她们一袋玉米，令她们的饥饿度各减少 1。

FJ 想将他的奶牛喂至所有的奶牛都具有相同的非负饥饿度。尽管他不知道他的奶牛们具体的饥饿度，他知道每一头奶牛的饥饿度上界；具体地说，第  $i$  头奶牛的饥饿度  $h_i$  至多为  $H_i$  ( $0 \leq H_i \leq 1000$ )。

你的工作是计算符合上述上界的  $N$  元组  $[h_1, h_2, \dots, h_N]$  的数量，使得 FJ 有可能达到他的目标，答案对  $10^9 + 7$  取模。

## 输入格式

输入的第一行包含  $N$ 。

第二行包含  $H_1, H_2, \dots, H_N$ 。

## 输出格式

输出符合条件的饥饿度的  $N$  元组数量，对  $10^9 + 7$  取模。

## 样例 #1

### 样例输入 #1

```
3
9 11 7
```

### 样例输出 #1

```
241
```

## 样例 #2

## 样例输入 #2

```
4
6 8 5 9
```

## 样例输出 #2

```
137
```

## 提示

【样例解释】

共有  $(9 + 1) \cdot (11 + 1) \cdot (7 + 1)$  个 3 元组  $h$  与  $H$  相符合。

$h = [8, 10, 5]$  是其中一个元组。在这个情况中，有可能使得所有的奶牛具有相同的饥饿度：给奶牛 2 和 3 各两袋玉米，然后给奶牛 1 和 2 各五袋玉米，可以使得所有奶牛的饥饿度均为 3。

$h = [0, 1, 0]$  是另一个元组。在这个情况中，不可能使得奶牛们的饥饿度相等。

【数据范围】

- 编号为偶数的测试点中的  $N$  均为偶数，编号为奇数的测试点中的  $N$  均为奇数。
- 测试点 3-4 满足  $N \leq 6$  以及  $H_i \leq 10$ 。
- 测试点 5-10 满足  $N \leq 50$  以及  $H_i \leq 100$ 。
- 测试点 11-20 没有额外限制。