

A. 神奇数字

题目描述：

某天你突然得到一串秘密数字，M 告诉你解密的方法是，将这些数字按照第一行 1 个，第二行 3 个，第三行 5 个，第 n 行 $(n-1)*2+1$ 个的顺序排放这些数，然后将每行最后一个数组合起来就是最终的密码。

你需要完成这个任务。

示例：

1 2 3 4 5 6 7 8 9 10

摆放：

1

2 3 4

5 6 7 8 9

10

则密码为：1 4 9 10

输入：

输入第一行为 T ，表示有 T 组测试数据， $(1 \leq T \leq 10)$ 。

接下来有 T 组测试数据，每组数据第一行是一个数 n ($1 \leq n \leq 1000$)，表示有 n 个神秘数字，接下来一行，包含 n 个整数 $a_0, a_1 \dots a_{n-1}$ ，相邻之间用一个空格分开，对每个数字 a_i ($0 \leq a_i \leq 1000, 0 \leq i < n$)。

输出：

你的输出包含 T 行，每行是一组测试数据的结果，

对每组测试数据，输出若干个数字，也即密码，数字之间用空格分开。

样例输入：

3

10

1 2 3 4 5 6 7 8 9 10

2

2 1

6

4 3 2 2 5 2

样例输出：

1 4 9 10

2 1

4 2 2

B. IP 地址统计

题目描述：

数据中心有一些 ip 地址库，包括每个地址区间以及归属省份(ip 地址区间不会重叠交叉)，现在要对用户访问日志统计每个省份的访问数量。

输入：

输入第一行为 T，表示包含 T 组测试数据。

对于每组测试数据，第一行为 n ($1 \leq n \leq 500000$)，表示 ip 地址库的 ip 区间数量。

接下来的 n 行，每行包含三个数据，ip_start, ip_end, province_id，表示 ip 区间的开始区间，结束区间，以及对应的省份 id。ip 区间为闭区间，而且各个区间不会重叠交叉。ip 地址以 a.b.c.d 形式提供。

接下来是 m($1 \leq m \leq 500000$)，表示用户访问日志数。

接下来的 m 行，每行是一个 ip 地址，表示用户访问的 ip 记录。

输出：

针对每组测试数据，按省份 id 输出每个省的访问数，每行第一个为省份 id，第二个为该省份的访问数量，中间用空格分开。如果某个省份没有访问过，则访问次数为 0。每组数据按省份 id 从小到大输出。

样例输入：

```
2
1
192.168.12.2 192.168.12.3 1
2
10.100.108.34
192.168.12.2
3
10.1.1.1 10.2.255.255 1
223.199.12.2 224.200.19.12 2
172.2.12.0 172.16.2.2 3
3
10.2.123.123
10.2.255.255
223.199.13.255
```

样例输出：

```
1 1
1 2
2 1
3 0
```

C. 数据回放

题目描述：

数据恢复是所有 dba 的噩梦，在某次数据库灾难之后，dba 终于挽救回一批数据，但只找回了一些数据库操作日志，你需要根据这些操作日志，重放整个数据库的操作结果。

数据库的操作命令有以下几种：

1. `set <key> <value>`
对 `key` 设置对应的 `value` 值（所有 `value` 都是整数）
2. `get <key>`
获取 `key` 的 `value` 值，并输出。如果 `key` 不存在，则输出 `nil`
3. `del <key>`
删除 `key`
4. `min`
输出整个数据库中 `value` 最小的值，如果数据库为空，则输出 `nil`

所有 `key` 都是由 `a-z`，`0-9` 字符组成的字符串，`value` 为整数且 $0 \leq \text{value} < 2^{32}$ 。

输入：

输入第一行是 `t` ($1 \leq t \leq 11$)，表示有 `t` 组数据。

接下来 `t` 组数据，每组第一行是 `n` ($1 \leq n \leq 500000$)，表示有 `n` 个操作。

接下来的 `n` 行，每行是上述 4 个操作的其中一个。

操作命令，`key`，`value` 之间用一个空格隔开。

输出：

对于每个 `get` 命令，输出对应 `key` 的值，如果 `key` 已不存在，则输出 `nil`

对于每个 `min` 命令，输出当前数据库中最小的 `value` 值，如果数据库为空，则输出 `nil`。

样例输入：

```
1
9
set a 10
set b 5
set c 30
get d
min
del b
min
set b 15
min
```

样例输出：

nil

5

10

10