

CSP-J 模拟

题目名称	小球	网络连接	合并	城市电网
输入/输出文件名	ball.in/out	network.in/out	merge.in/out	electric.in/out
测试点时限	1 s	1 s	1 s	1 s
内存限制	256MB	256MB	256MB	256MB
分值	100	100	100	100
测试点/子任务个数	10	20	4	10
题目类型	传统型	传统型	传统型	传统型

注意事项

1. 需要建立子文件夹。
2. 文件名（包括程序名和输入输出文件名）必须使用英文小写。
3. 结果比较方式为忽略行末空格、文末回车后的全文比较。
4. 程序可使用的栈空间大小与该题内存空间限制一致。
5. 编译选项： `-O2 -std=c++14 -lm` 。

1. 小球 (ball.cpp)

描述

数轴上有 n 个球，第 i 个位于 x_i 。

初始时，每个球的滚动方向要么向左，要么向右。

接下来每 1 秒，每个球都会向着它的滚动方向滚动 0.5 个单位。

当两个球重合时，它们就会一起原地消失。

请问每个球在第几秒末消失，或者永远不会消失。

格式

输入格式

第一行一个正整数 n 。

第二行 n 个正整数 x_i 。

第三行一个长为 n 的只包含 L 和 R 的字符串，描述每个球的滚动方向。

输出格式

一行 n 个数，依次表示每个球的消失时刻，若永远不会消失则消失时刻视为 -1。

样例

样例输入1

```
5
1 3 5 6 9
RLRRL
```

样例输出1

```
2 2 -1 3 3
```

样例输入2,3/样例输出2,3

见下发文件。

数据范围

对于 60% 的数据： $x_n \leq 5000$ 。

对于 100% 的数据： $2 \leq n \leq 10^5$, $1 \leq x_1 \leq x_2 \leq \dots \leq x_n \leq 10^9$ 。

均匀分布着 $\frac{1}{2}$ 的数据，保证所有球都会消失。

2. 网络连接 (network.cpp)

描述

TCP/IP 协议是网络通信领域的一项重要协议。今天你的任务，就是尝试利用这个协议，还原一个简化后的网络连接场景。

在本问题中，计算机分为两大类：服务机 (server) 和客户机 (client)。服务机负责建立连接，客户机负责加入连接。

需要进行网络连接的计算机共有 n 台，编号为 $1 \sim n$ ，这些机器将按编号递增的顺序，依次发起一条建立连接或加入连接的操作。

每台机器在尝试建立或加入连接时需要提供一个地址串。服务机提供的地址串表示它尝试建立连接的地址，客户机提供的地址串表示它尝试加入连接的地址。

一个符合规范的地址串应当具有以下特征：

1. 必须形如 `a.b.c.d:e` 的格式，其中 a, b, c, d, e 均为非负整数；
2. $0 \leq a, b, c, d \leq 255, 0 \leq e \leq 65535$ ；
3. a, b, c, d, e 均不能含有多余的前导 0。

相应地，不符合规范的地址串可能具有以下特征：

1. 不是形如 `a.b.c.d:e` 格式的字符串，例如含有多于 3 个字符 `.` 或多于 1 个字符 `:` 等情况；
2. 整数 a, b, c, d, e 中某一个或多个超出上述范围；
3. 整数 a, b, c, d, e 中某一个或多个含有多余的前导 0。

例如，地址串 `192.168.0.255:80` 是符合规范的，但 `192.168.0.999:80`、`192.168.00.1:10`、`192.168.0.1:088`、`192:168:0:1.233` 均是不符合规范的。

如果服务机或客户机在发起操作时提供的地址串不符合规范，这条操作将被直接忽略。

在本问题中，我们假定凡是符合上述规范的地址串均可参与正常的连接，你无需考虑每个地址串的实际意义。

由于网络阻塞等原因，不允许两台服务机使用相同的地址串，如果此类现象发生，后一台尝试建立连接的服务机将会无法成功建立连接；除此之外，凡是提供符合规范的地址串的服务机均可成功建立连接。

如果某台提供符合规范的地址的客户机在尝试加入连接时，与先前某台已经成功建立连接的服务机提供的地址串相同，这台客户机就可以成功加入连接，并称其连接到这台服务机；如果找不到这样的服务机，则认为这台客户机无法成功加入连接。

请注意，尽管不允许两台不同的服务机使用相同的地址串，但多台客户机使用同样的地址串，以及同一台服务机同时被多台客户机连接的情况是被允许的。

你的任务很简单：在给出每台计算机的类型以及地址串之后，判断这台计算机的连接情况。

格式

输入格式

第一行，一个正整数 n 。

接下来 n 行，每行两个字符串 op, ad ，按照编号从小到大给出每台计算机的类型及地址串。

其中 op 保证为字符串 `Server` 或 `Client` 之一, ad 为一个长度不超过 25 的, 仅由数字、字符 `.` 和字符 `:` 组成的非空字符串。

每行的两个字符串之间用恰好一个空格分隔开, 每行的末尾没有多余的空格。

输出格式

输出共 n 行, 每行一个正整数或字符串表示第 i 台计算机的连接状态。其中:

如果第 i 台计算机为服务机, 则:

1. 如果其提供符合规范的地址串且成功建立连接, 输出字符串 `OK`。
2. 如果其提供符合规范的地址串, 但由于先前有相同地址串的服务机而无法成功建立连接, 输出字符串 `FAIL`。
3. 如果其提供的地址串不是符合规范的地址串, 输出字符串 `ERR`。

如果第 i 台计算机为客户机, 则:

1. 如果其提供符合规范的地址串且成功加入连接, 输出一个正整数表示这台客户机连接到的服务机的编号。
2. 如果其提供符合规范的地址串, 但无法成功加入连接时, 输出字符串 `FAIL`。
3. 如果其提供的地址串不是符合规范的地址串, 输出字符串 `ERR`。

样例

样例输入1

```
5
Server 192.168.1.1:8080
Server 192.168.1.1:8080
Client 192.168.1.1:8080
Client 192.168.1.1:80
Client 192.168.1.1:99999
```

样例输出1

```
OK
FAIL
1
FAIL
ERR
```

样例输入2

```
10
Server 192.168.1.1:80
Client 192.168.1.1:80
Client 192.168.1.1:8080
Server 192.168.1.1:80
Server 192.168.1.1:8080
Server 192.168.1.999:0
Client 192.168.1.1.8080
Client 192.168.1.1:8080
Client 192.168.1.1:80
Client 192.168.1.999:0
```

样例输出2

```
OK
1
FAIL
FAIL
OK
ERR
ERR
5
1
ERR
```

样例输入3,4/样例输出3,4

见下发文件。

样例解释

对于样例 1：

- 计算机 1 为服务机，提供符合规范的地址串 192.168.1.1:8080，成功建立连接；
- 计算机 2 为服务机，提供与计算机 1 相同的地址串，未能成功建立连接；
- 计算机 3 为客户机，提供符合规范的地址串 192.168.1.1:8080，成功加入连接，并连接到服务机 1；
- 计算机 4 为客户机，提供符合规范的地址串 192.168.1.1:80，找不到服务机与其连接；
- 计算机 5 为客户机，提供的地址串 192.168.1.1:99999 不符合规范。

数据范围

测试点编号	$n \leq$	特殊性质
1	10	性质 1 2 3
2 ~ 3	100	性质 1 2 3
4 ~ 5	1000	性质 1 2 3
6 ~ 8	1000	性质 1 2

测试点编号	$n \leq$	特殊性质
9 ~ 11	1000	性质 1
12 ~ 13	1000	性质 2
14 ~ 15	1000	性质 4
16 ~ 17	1000	性质 5
18 ~ 20	1000	无特殊性质

- “性质 1”为：保证所有的地址串均符合规范；
- “性质 2”为：保证对于任意两台不同的计算机，如果它们同为服务机或者同为客户机，则它们提供的地址串一定不同；
- “性质 3”为：保证任意一台服务机的编号都小于所有的客户机；
- “性质 4”为：保证所有的地址串均形如 `a.b.c.d:e` 的格式，其中 a, b, c, d, e 均为不超过 10^9 且不含有多余前导 0 的非负整数；
- “性质 5”为：保证所有的地址串均形如 `a.b.c.d:e` 的格式，其中 a, b, c, d, e 均为只含有数字的非空字符串。

对于 100% 的数据，保证 $1 \leq n \leq 1000$ 。

3. 合并 (merge.cpp)

描述

你有 n 个小球排成一行，依次编号为 $1 \sim n$ ，第 i 个小球的权值记为 v_i 。

你可以对小球进行如下操作：

- 如果两个**相邻**小球的权值相等，你可以将它们合并为一个新的小球，新小球的权值是原来两个小球的权值之和，他将占据两个原小球先前占据的位置。
- 如果两个小球权值相等，且它们之间只有一个小球，你也可以把它们合并为一个新的小球（无需考虑中间小球的权值）。新小球的权值是原来的三个小球的权值之和，并占据三个原小球先前占据的位置。

你可以按照你的意愿进行任意次操作。请你回答，最终能得到的权值最大的小球，它的权值是多少。

格式

输入格式

第一行一个整数 n 表示小球数目。

第二行 n 个正整数，表示每个小球的权值，按照从左到右的顺序给出。

输出格式

一行一个整数表示答案。

样例

样例输入1

```
7
47 12 12 3 9 9 3
```

样例输出1

```
48
```

样例输入2

```
4
1 2 3 1
```

样例输出2

```
3
```

样例输入3,4/样例输出3,4

见下发文件。

样例解释

对于样例 1：

可以先合并权值同为 12 的两个小球，得到一个权值为 24 的小球。然后合并权值同为 9 的两个小球，得到一个权值为 18 的小球。接着合并权值为 3, 18 和 3 的三个小球，得到一个权值为 24 的小球。最后合并权值同为 24 的两个小球，得到一个权值为 48 的小球。

对于样例 2：

无法进行任何操作，答案为 3。

数据范围

对于 10% 的数据， $n = 4$ 。

对于 25% 的数据， $1 \leq n \leq 10$ 。

对于 50% 的数据， $1 \leq n \leq 50$ 。

对于 100% 的数据， $1 \leq n \leq 400$, $1 \leq v_i \leq 10^6$ 。

4. 城市电网 (electric.cpp)

描述

城市中一共有 n 个位置架设了电线杆，这些电线杆分别编号为 $1 \sim n$ 。然而，地震损坏了城市的电力网络。

你的任务是让城市中的 1 号电线杆和 n 号电线杆连通。你有 p 个可供选择的方案，第 i 个方案是连接编号为 u_i, v_i 的两个电线杆并花费 w_i 的费用。

市政府决定为你的工作提供资金支持。你可以从要执行的所有方案中，选出不超过 k 个方案，这些方案产生的费用由市政府提供，除此之外的方案产生的方案需要自行支付。

请问，如何选择方案，可以使得由自己支付的方案中，花费最多的方案，所需费用最少？

请输出这个费用。

格式

输入格式

第一行 3 个正整数 n, p, k ，含义如题所述。

接下来 p 行，每行 3 个正整数 u_i, v_i, w_i ，含义如题所述。

输出格式

一行一个整数表示答案。如果无法完成任务，输出 -1。

样例

样例输入1

```
5 7 1
1 2 5
3 1 4
2 4 8
3 2 3
5 2 9
3 4 7
4 5 6
```

样例输出1

```
4
```

数据范围

对于 20% 的数据： $n, p \leq 7, k = 1$ 。

对于 40% 的数据： $n, p \leq 50, k \leq 3$ 。

对于 100% 的数据: $1 \leq n \leq 1000$, $1 \leq p \leq 10000$, $1 \leq k \leq 30$, $1 \leq u_i, v_i \leq n$,
 $1 \leq w_i \leq 10^6$ 。