

# 2014 考研机试

由于 912-project 原文档题面不全，故进行重构补全

## T1 相反数

第一次 CCF 认证 [第一题](#)

### 题目描述

有  $N$  个非零且各不相同的整数。请你编一个程序求出它们中有多少对相反数 ( $a$  和  $-a$  为一对相反数)

### 输入格式

第一行包含一个正整数  $N(1 \leq N \leq 500)$

第二行为  $N$  个用单个空格隔开的非零整数，每个数的绝对值不超过 1000，保证这些整数各不相同

### 输出格式

只输出一个整数，即这  $N$  个数中包含多少对相反数

### 样例 #1

#### 样例输入 #1

```
1 | 5
2 | 1 2 3 -1 -2
```

#### 样例输出 #1

```
1 | 2
```

## T2 窗口

第一次 CCF 认证 [第二题](#)

### 题目描述

在某图形操作系统中,有  $N$  个窗口，每个窗口都是一个两边与坐标轴分别平行的矩形区域。窗口的边界上的点也属于该窗口。窗口之间有层次的区别，在多于一个窗口重叠的区域里，只会显示位于顶层的窗口里的内容

当你点击屏幕上一个点的时候，你就选择了处于被点击位置的最顶层窗口，并且这个窗口就会被移到所有窗口的最顶层,而剩余的窗口的层次顺序不变。如果你点击的位置不属于任何窗口，则系统会忽略你这次点击

现在希望你写一个程序模拟点击窗口的过程

### 输入格式

输入的第一行有两个正整数，即  $N$  和  $M$  ( $1 \leq N \leq 10, 1 \leq M \leq 10$ )

接下来  $N$  行按照从最下层到最顶层的顺序给出  $N$  个窗口的位置。每行包含四个非负整数  $x_1, y_1, x_2, y_2$ ，表示该窗口的一对顶点坐标分别为  $(x_1, y_1)$  和  $(x_2, y_2)$ 。保证  $x_1 < x_2, y_1 < y_2$

接下来  $M$  行每行包含两个非负整数  $x, y$ ，表示一次鼠标点击的坐标

题目中涉及到的所有点和矩形的顶点的  $x, y$  坐标分别不超过 2559 和 1439

## 输出格式

输出包括  $M$  行，每一行表示一次鼠标点击的结果。如果该次鼠标点击选择了一个窗口，则输出这个窗口的编号（窗口按照输入中的顺序从 1 编号到  $N$ ）；如果没有，则输出 `IGNORED`

## 样例 #1

### 样例输入 #1

```
1 3 4
2 0 0 4 4
3 1 1 5 5
4 2 2 6 6
5 1 1
6 0 0
7 4 4
8 0 5
```

### 样例输出 #1

```
1 2
2 1
3 1
4 IGNORED
```

## 提示

### 样例说明

第一次点击的位置同时属于第 1 和第 2 个窗口，但是由于第 2 个窗口在上面，它被选择并且被置于顶层

第二次点击的位置只属于第 1 个窗口，因此该次点击选择了此窗口并将其置于顶层。现在的三个窗口的层次关系与初始状态恰好相反了

第三次点击的位置同时属于三个窗口的范围，但是由于现在第 1 个窗口处于顶层，它被选择

最后点击的  $(0, 5)$  不属于任何窗口

## T3 命令行选项

第一次 CCF 认证 [第三题](#)

## 题目描述

请你写一个命令行分析程序，用以分析给定的命令行里包含哪些选项。每个命令行由若干个字符串组成，它们之间恰好由一个空格分隔。这些字符串中的第一个为该命令行工具的名字，由小写字母组成，你的程序不用对它进行处理。在工具名字之后可能会包含若干选项，然后可能会包含一些不是选项的参数

选项有两类：带参数的选项和不带参数的选项。一个合法的无参数选项的形式是一个减号后面跟单个小写字母，如 `-a` 或 `-b`。而带参数选项则由两个由空格分隔的字符串构成，前者的格式要求与无参数选项相同，后者则是该选项的参数，是由小写字母，数字和减号组成的非空字符串

该命令行工具的作者提供给你一个格式字符串以指定他的命令行工具需要接受哪些选项。这个字符串由若干小写字母和冒号组成，其中的每个小写字母表示一个该程序接受的选项。如果该小写字母后面紧跟了一个冒号，它就表示一个带参数的选项，否则则为不带参数的选项。例如，`ab:m` 表示该程序接受三种选项，即 `-a`（不带参数），`-b`（带参数），以及 `-m`（带参数）

命令行工具的作者准备了若干条命令行用以测试你的程序。对于每个命令行，你的工具应当一直向后分析。当你的工具遇到某个字符串既不是合法的选项，又不是某个合法选项的参数时，分析就停止。命令行剩余的未分析部分不构成该命令的选项，因此你的程序应当忽略它们

## 输入格式

输入的第一行是一个格式字符串，它至少包含一个字符，且长度不超过 52。格式字符串只包含小写字母和冒号，保证每个小写字母至多出现一次，不会有两个相邻的冒号，也不会以冒号开头

输入的第二行是一个正整数  $N$  ( $1 \leq N \leq 20$ )，表示你需要处理的命令行的个数

接下来有  $N$  行，每行是一个待处理的命令行，它包括不超过 256 个字符。该命令行一定是若干个由单个空格分隔的字符串构成，每个字符串里只包含小写字母，数字和减号

## 输出格式

输出有  $N$  行。其中第  $i$  行以 `Case i:` 开始，然后应当有恰好一个空格，然后应当按照字母升序输出该命令行中用到的所有选项的名称，对于带参数的选项，在输出它的名称之后还要输出它的参数。如果一个选项在命令行中出现了多次，只输出一次。如果一个带参数的选项在命令行中出现了多次，只输出最后一次出现时所带的参数

## 样例 #1

### 样例输入 #1

```
1 albw:x
2 4
3 ls -a -l -a documents -b
4 ls
5 ls -w 10 -x -w 15
6 ls -a -b -c -d -e -l
```

### 样例输出 #1

```
1 Case 1: -a -l
2 Case 2:
3 Case 3: -w 15 -x
4 Case 4: -a -b
```

## T4 无线网络

# 题目背景

第一次 CCF 认证 [第四题](#)

# 题目描述

目前在一个很大的平面房间里有  $n$  个无线路由器，每个无线路由器都固定在某个点上。任何两个无线路由器只要距离不超过  $r$  就能互相建立网络连接

除此以外，另有  $m$  个可以摆放无线路由器的位置。你可以在这些位置中选择至多  $k$  个增设新的路由器

你的目标是使得第 1 个路由器和第 2 个路由器之间的网络连接经过尽量少的中转路由器。请问在最优方案下中转路由器的最少个数是多少？

# 输入格式

第一行包含四个正整数  $n, m, k, r$  ( $2 \leq n \leq 100, 1 \leq k \leq m \leq 100, 1 \leq r \leq 10^8$ )

接下来  $n$  行，每行包含两个整数  $x_i$  和  $y_i$ ，表示一个已经放置好的无线路由器在  $(x_i, y_i)$  点处。输入数据保证第 1 和第 2 个路由器在仅有这  $n$  个路由器的情况下已经可以互相连接（经过一系列的中转路由器）

接下来  $m$  行，每行包含两个整数  $x_i$  和  $y_i$ ，表示  $(x_i, y_i)$  点处可以增设一个路由器

输入中所有的坐标的绝对值不超过  $10^8$ ，保证输入中的坐标各不相同

# 输出格式

输出只有一个数，即在指定的位置中增设  $k$  个路由器后，从第 1 个路由器到第 2 个路由器最少经过的中转路由器的个数

# 样例 #1

## 样例输入 #1

```
1 | 5 3 1 3
2 | 0 0
3 | 5 5
4 | 0 3
5 | 0 5
6 | 3 5
7 | 3 3
8 | 4 4
9 | 3 0
```

## 样例输出 #1

```
1 | 2
```

# T5 任务调度

# 题目背景

第一次 CCF 认证 [第五题](#)

## 题目描述

有若干个任务需要在一台机器上运行。它们之间没有依赖关系，因此 可以被按照任意顺序执行

该机器有两个 CPU 和一个 GPU。对于每个任务，你可以为它分配不同的硬件资源：

1. 在单个 CPU 上运行；
2. 在两个 CPU 上同时运行；
3. 在单个 CPU 和 GPU 上同时运行；
4. 在两个 CPU 和 GPU 上同时运行。

一个任务开始执行以后，将会独占它所用到的所有硬件资源，不得中断，直到执行结束为止。第  $i$  个任务用单个 CPU，两个 CPU，单个 CPU 加 GPU，两个 CPU 加 GPU 运行所消耗的时间分别为  $a_i, b_i, c_i$  和  $d_i$

现在需要你计算出至少需要花多少时间可以把所有给定的任务完成

## 输入格式

输入的第一行只有一个正整数  $n$  ( $1 \leq n \leq 40$ )，是总共需要执行的任务个数

接下来的  $n$  行每行有四个正整数  $a_i, b_i, c_i, d_i$  ( $a_i, b_i, c_i, d_i \leq 10$ )，以空格隔开

## 输出格式

输出只有一个整数，即完成给定的所有任务所需的最少时间

## 样例 #1

### 样例输入 #1

```
1 | 3
2 | 4 4 2 2
3 | 7 4 7 4
4 | 3 3 3 3
```

### 样例输出 #1

```
1 | 7
```

## 提示

### 样例说明

有很多种调度方案可以在 7 个时间单位里完成给定的三个任务，以下是其中的一种方案：

同时运行第一个任务（单 CPU 加上 GPU）和第三个任务（单 CPU），它们分别在时刻 2 和时刻 3 完成。在时刻 3 开始双 CPU 运行任务 2，在时刻 7 完成