**集团软件研发岗位（编程）大比武试题**

时间：2013年10月26日

**一．题目概况**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 中文题目名称 | 三角形数阵 | 队列 | 浇水 | 迷宫 |
| 测试数据输入文件名 | noname.in | change.in | water.in | maze.in |
| 测试数据输出文件名 | noname.out | change.out | water.out | maze.out |
| 测试点数目 | 10 | 10 | 10 | 10 |
| 每个测试点分值 | 10分 | 10分 | 10分 | 10分 |
| 每个测试点时间限制 | 2秒 | 2秒 | 2秒 | 2秒 |
| 内存限制 | 512MB | 512MB | 512MB | 512MB |
| 结果比较方式 | 全文比较（过滤掉行末空格及文末回车） | | | |

**二．选手提交源程序文件名**(选手任选一种语言即可，提交其中一种语言的源程序)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| C语言 | noname.c | change.c | water.c | maze.c |
| C++语言 | noname.cpp | change.cpp | water.cpp | maze.cpp |
| C# | noname.cs | change.cs | water.cs | maze.cs |
| Java | noname.java | change.java | water.java | maze.java |
| PB | noname.exe | change.exe | water.exe | maze.exe |

**三．编译命令（不含任何优化开关）**

|  |  |  |
| --- | --- | --- |
| 语言 | 编译命令 | 评测版本 |
| C语言 | gcc.exe | GCC 4.4.5 |
| C++语言 | g++.exe | G++ 4.4.5 |
| C# | csc.exe | Framework 4.0 |
| Java | javac.exe | JDK 1.7 |
| PB | - | - |

**三角形数阵**

**【题目描述】**

你也许见过各种各样的三角形数阵。下面给出一种三角形数阵（你可能没见过），这种数阵是由一个有序三元组（S,T,N）决定的，其中S,T,N是三个正整数，观察下面几个样例：

|  |  |  |  |
| --- | --- | --- | --- |
| 三元组（S,T,N） | **（1 , 4 , 6）** | **( 3 , 5 , 8 )** | **( 4 , 7 , 9 )** |
| 三角形数阵 |  |  |  |

对于给定一个具体的三元组（S,T,N），能唯一确定一个相对应的三角形数阵，请自己寻找三元组与数阵的对应关系（提示：数阵的第一个数；按列读读看…..）。

你的任务是：根据给定的三元组（S,T,N），输出相对应的三角形数阵。

**【输入】**

一行，三个正整数S,T,N，两个数字中间用一个空格分隔，行的末尾没有空格。

**【输出】**

与三元组(S,T,N)对应的三角形数阵。

格式要求：每行的两个数字之间用一个空格隔开；行的末尾不允许添加空格。

**【样例】**

|  |  |
| --- | --- |
| noname.in | noname.out |
| 3 5 8 |  |

**【数据规模】**

1<=S<=T<=9 ；1<=N<=20。

# 队 列

【问题描述】

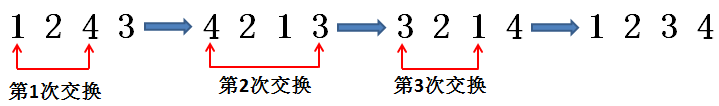
n个士兵的编号分别为1,2,3，…，n，每次站队时长官都是要求n个士兵从排头到排尾依次是编号为1,2,3…，n的士兵。如果次序乱了，长官很不高兴，后果会很严重。

有一次士兵按编号次序刚刚站好队，突然长官下令让所有士兵立刻回宿舍拿水壶，要去执行一个临时任务，由于时间紧张，慌乱的士兵们拿到水壶回来后，有的没能站到自己应该站的位置。

长官不高兴了，于是下令让士兵们通过两两交换位置的方法按原来次序站好，并且规定：每一次交换位置，**只能是排头的人和除排头外的任一个位置的人交换**。

长官要求士兵们用最少的交换次数站好队列，否则等待他们是跑10公里的惩罚。

如n=4时，4个士兵的顺序是：1 2 4 3。可以经过下面的3次交换让队列恢复。



现在告诉你士兵们的队形，问最少交换多少次能恢复到1,2,3,。。。,n的顺序。

【输入】

第一行是n，代表队列的长度。

第二行共有n个用空格隔开的数，依次是队形中n个士兵的编号（第一个是排头士兵的编号）。

【输出】

一个正整数m，代表最少经过m次交换可以变成1 2 3 ... n。

【输入输出样例1】

|  |  |
| --- | --- |
| change.in | change.out |
| 3  3 1 2 | 2 |

【输入输出样例1解释】

2次变换过程：3 1 2 🡪 2 1 3🡪 1 2 3。

【输入输出样例2】

|  |  |
| --- | --- |
| change.in | change.out |
| 4  1 2 4 3 | 3 |

【输入输出样例2解释】

3次变换过程： 1 2 4 3 🡪 4 2 1 3 🡪 3 2 1 4 🡪 1 2 3 4。

【数据范围限制】

30%的数据范围：n<=30。

100%的数据范围：n<=100。

# 浇 水

【题目描述】

瓜农王大爷去年种西瓜赚了不少钱。看到收入不错，今年他又重新开辟了n个西瓜地。

为了能给他的n个西瓜地顺利的浇上水，对于每个西瓜地他可以选择在本地打井，也可以修管道从另一个瓜地（这个瓜地可能打了井；也可能没打井，他的水也是从其他瓜地引来的）将水引过来。

当然打井和修管道的费用有差别。已知在第i个西瓜地打井需要耗费wi元，在第i、j个西瓜地之间修管道需要耗费pi,j元。

现在的问题是：王大爷要想使所有瓜地都被浇上水，至少需要花费多少钱（打井与修管道的费用和）？

由于瓜地较多，王大爷无法选择在哪些（个）瓜地打井，哪些西瓜地之间修管道。

请你编程帮王大爷做出决策，求出最小费用。

【输入格式】

第1行，一个正整数n，代表西瓜地的数量。

以下n行，依次给出整数w1..wn（每块西瓜地的打井费用）。

紧接着是一个n\*n的整数矩阵，矩阵的第i行第j列的数代表pi,j（两块西瓜地之间建立管道的费用）。每行的两个数之间有一个空格隔开。

【输出格式】

一个正整数，所求的最小花费。

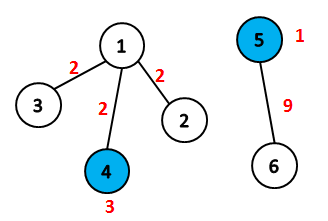
|  |  |
| --- | --- |
| water.in | water.out |
| 6  5  4  4  3  1  20  0 2 2 2 9 9  2 0 3 3 9 9  2 3 0 4 9 9  2 3 4 0 9 9  9 9 9 9 0 9  9 9 9 9 9 0 | 19 |

【样例说明】

在第4个瓜地打井（费用为3），然后将第2,3，4个瓜地与第1个瓜地间修管道（费用分别是2,2,2），这样水可以经过管道从4流向1，然后经1再流向2和3；

在第5个瓜地打井（费用为1），5和6之间修管道（费用为9）。

这样一共打了2口井，修了4条管道，能给所有的6个瓜地浇水，费用是：3+2+2+2+1+9=19。



【数据规模和约定】

对于所有数据，1<=N<=300，1<=wi<=100000；pi,i=0，1<=pi,j=pj,i<=100000。

# 迷 宫

**【问题描述】**

迷宫游戏有多种。有的迷宫只要你能走出来就算成功；有的是要求不但能走出来，而且走的步数越少得分越高。为了能得到高分，你肯定在路线的选择上下功夫，尽量用最少的步数走出迷宫。

现有一个迷宫，可以用M行N列的矩阵来描述，迷宫中有以下6种标志：

（1）**$** ：代表你所在的初始位置；

（2）**&** ：迷宫的出口位置；

（3）**#** ：墙，无法通过；

（4）**.** ：地面，可以通过；

（5）**A、B、C、...、Z** ：每个大写字母代表一个门，可以用对应的小写字母的钥匙打开。

（6）**a、b、c、...、z** ：每个小写字母代表一把钥匙，可以打开相应大写字母对应的门。

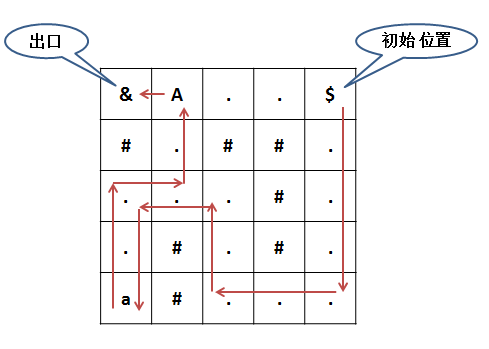
**迷宫说明**：

（1）迷宫的四周都有墙，所以你无法走出这片M\*N的区域，只能从出口’&’处离开迷宫；

（2）在迷宫中你只能向东西南北四个方向走（当然有墙的地方你无法达到；如果没有相应的钥匙，有门的地方你也无法到达）。

（3）如果能走到有小写字母（钥匙）的位置，你就可以拿到它，并用来打开相应大写字母的门，钥匙a只能开A门，钥匙b只能开B门，……

如下列迷宫



最快走出迷宫的方法是：从初始位置$走12步到达钥匙a处拿到钥匙后，然后再走5步到达A处，打开门A后再走1步到达出口&，共走了12+5+1=18步。

上述迷宫中，如果你拿不到钥匙a，就无法打开门A,也就到不了出口。

有的迷宫可能不需要走有门的地方，就能最快走到迷宫出口。

你的任务是，计算出走出迷宫需要的最少步数是多少？

**【输入格式】**

第1行两个正整数M和N，表示迷宫的长和宽；

第2行一个正整数P，表示门和钥匙的数量（钥匙和门的标号只会出现字母表的前P个字母，如p=2，则只出现钥匙a,b和门A,B）。

第3行至第M+2行，描述整个迷宫。

**【输出格式】**

一个整数，为走出迷宫需要的最少步数或-1（如果不可能走出迷宫）。

**【输入输出样例1】**

|  |  |
| --- | --- |
| maze.in | maze.out |
| 5 5  1  &A..$  #.##.  ...#.  .#.#.  a#... | 18 |

**【输入输出样例2】**

|  |  |
| --- | --- |
| maze.in | maze.out |
| 1 4  1  &aA$ | -1 |

**【数据规模和约定】**

存在10%的数据，P=0；

存在20%的数据，P<=1；

存在40%的数据，P<=2；

对于30%的数据，M,N<=15；

对于100%的数据，1<=M,N<=50，0<=P<=10。

保证迷宫中’$’、’&’只出现一次；同一个大小写字母只出现一次（p=0的迷宫里面没有钥匙和门）。