

第二十八届全国信息学奥林匹克竞赛

CCF NOI 2011

第二试

竞赛时间：2011 年 8 月 10 日 8:00–13:00

题目名称	道路修建	NOI 嘉年华	兔兔与蛋蛋的游戏
目录	road	show	game
可执行文件名	road	show	game
输入文件名	road.in	show.in	game.in
输出文件名	road.out	show.out	game.out
每个测试点时限	2 秒	1 秒	1 秒
内存限制	256M	256M	256M
测试点数目	20	10	20
每个测试点分值	5	10	5
是否有部分分	否	有	否
题目类型	传统型	传统型	传统型

提交源程序须加后缀

对于 Pascal 语言	road.pas	show.pas	game.pas
对于 C 语言	road.c	show.c	game.c
对于 C++ 语言	road.cpp	show.cpp	game.cpp

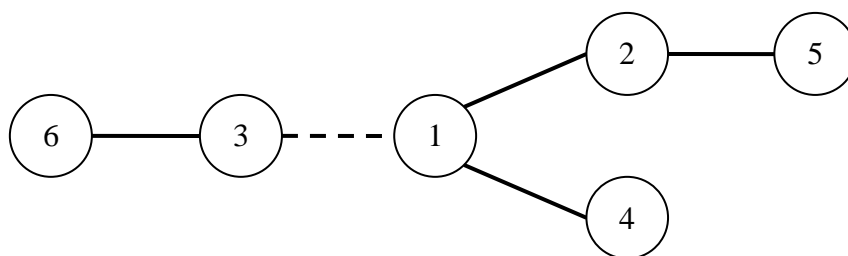
注意：最终测试时，所有编译命令均不打开任何优化开关。

道路修建

【问题描述】

在 W 星球上有 n 个国家。为了各自国家的经济发展，他们决定在各个国家之间建设双向道路使得国家之间连通。但是每个国家的国王都很吝啬，他们只愿意修建恰好 $n-1$ 条双向道路。

每条道路的修建都要付出一定的费用，这个费用等于道路长度乘以道路两端的国家个数之差的绝对值。例如，在下图中，虚线所示道路两端分别有 2 个、4 个国家，如果该道路长度为 1，则费用为 $1 \times |2 - 4| = 2$ 。图中圆圈里的数字表示国家的编号。



由于国家的数量十分庞大，道路的建造方案有很多种，同时每种方案的修建费用难以用人工计算，国王们决定找人设计一个软件，对于给定的建造方案，计算出所需要的费用。请你帮助国王们设计一个这样的软件。

【输入格式】

从文件 **road.in** 中读入数据。

输入的第一行包含一个整数 n ，表示 W 星球上的国家的数量，国家从 1 到 n 编号。

接下来 $n-1$ 行描述道路建设情况，其中第 i 行包含三个整数 a_i 、 b_i 和 c_i ，表示第 i 条双向道路修建在 a_i 与 b_i 两个国家之间，长度为 c_i 。

【输出格式】

输出到文件 **road.out** 中。

输出一个整数，表示修建所有道路所需要的总费用。

【样例输入】

```

6
1 2 1
1 3 1
1 4 2
6 3 1
5 2 1

```

【样例输出】

```

20

```

【数据规模与约定】

所有测试数据的范围和特点如下表所示

测试点编号	n 的规模（注意是等于号）	约定
1	$n = 2$	$1 \leq a_i, b_i \leq n$ $0 \leq c_i \leq 10^6$
2	$n = 10$	
3	$n = 100$	
4	$n = 200$	
5	$n = 500$	
6	$n = 600$	
7	$n = 800$	
8	$n = 1000$	
9	$n = 10,000$	
10	$n = 20,000$	
11	$n = 50,000$	
12	$n = 60,000$	
13	$n = 80,000$	
14	$n = 100,000$	
15	$n = 600,000$	
16	$n = 700,000$	
17	$n = 800,000$	
18	$n = 900,000$	
19	$n = 1,000,000$	
20	$n = 1,000,000$	

NOI 嘉年华

【问题描述】

NOI2011 在吉林大学开始啦！为了迎接来自全国各地最优秀的信息学选手，吉林大学决定举办两场盛大的 NOI 嘉年华活动，分在两个不同的地点举办。每个嘉年华可能包含很多个活动，而每个活动只能在一个嘉年华中举办。

现在嘉年华活动的组织者小安一共收到了 n 个活动的举办申请，其中第 i 个活动的起始时间为 S_i ，活动的持续时间为 T_i 。这些活动都可以安排到任意一个嘉年华的会场，也可以不安排。

小安通过广泛的调查发现，如果某个时刻，两个嘉年华会场同时有活动在进行（不包括活动的开始瞬间和结束瞬间），那么有的选手就会纠结于到底去哪个会场，从而变得不开心。所以，为了避免这样不开心的事情发生，小安要求不能有两个活动在两个会场同时进行（同一会场内的活动可以任意进行）。

另外，可以想象，如果某一个嘉年华会场的活动太少，那么这个嘉年华的吸引力就会不足，容易导致场面冷清。所以小安希望通过合理的安排，使得活动相对较少的嘉年华的活动数量最大。

此外，有一些活动非常有意义，小安希望能举办，他想知道，如果第 i 个活动必须举办（可以安排在两场嘉年华中的任何一个），活动相对较少的嘉年华的活动数量的最大值。

【输入格式】

从文件 *show.in* 中读入数据。

输入的第一行包含一个整数 n ，表示申请的活动个数。

接下来 n 行描述所有活动，其中第 i 行包含两个整数 S_i 、 T_i ，表示第 i 个活动从时刻 S_i 开始，持续 T_i 的时间。

【输出格式】

输出到文件 *show.out* 中。

输出的第一行包含一个整数，表示在没有任何限制的情况下，活动较少的嘉年华的活动数的最大值。

接下来 n 行每行一个整数，其中第 i 行的整数表示在必须选择第 i 个活动的前提下，活动较少的嘉年华的活动数的最大值。

【评分标准】

对于一个测试点：

- 如果输出格式不正确（比如输出不足 $n+1$ 行），得 0 分；
- 如果输出文件第一行不正确，而且后 n 行至少有一行不正确，得 0 分；
- 如果输出文件第一行正确，但后 n 行至少有一行不正确，得 4 分；

- 如果输出文件第一行不正确，但后 n 行均正确，得 6 分；
- 如果输出文件中的 $n+1$ 行均正确，得 10 分。

【样例输入】

```
5
8 2
1 5
5 3
3 2
5 3
```

【样例输出】

```
2
2
1
2
2
2
```

【样例说明】

在没有任何限制的情况下，最优安排可以在一个嘉年华安排活动 1, 4，而在另一个嘉年华安排活动 3, 5，活动 2 不安排。

【数据规模与约定】

所有测试数据的范围和特点如下表所示

测试点编号	n 的规模	约定
1	$1 \leq n \leq 10$	$0 \leq S_i \leq 10^9$ $1 \leq T_i \leq 10^9$
2	$1 \leq n \leq 40$	
3		
4	$1 \leq n \leq 200$	
5		
6		
7		
8		
9		
10		

兔兔与蛋蛋游戏

【问题描述】

这些天，兔兔和蛋蛋喜欢上了一种新的棋类游戏。

这个游戏是在一个 n 行 m 列的棋盘上进行的。游戏开始之前，棋盘上有一个格子是空的，其它的格子中都放置了一枚棋子，棋子或者是黑色，或者是白色。

每一局游戏总是兔兔先操作，之后双方轮流操作，具体操作为：

- 兔兔每次操作时，选择一枚与空格相邻的白色棋子，将它移进空格。
- 蛋蛋每次操作时，选择一枚与空格相邻的黑色棋子，将它移进空格。

第一个不能按照规则操作的人输掉游戏。为了描述方便，下面将操作“将第 x 行第 y 列中的棋子移进空格中”记为 $M(x,y)$ 。

例如下面是三个游戏的例子。



图 1

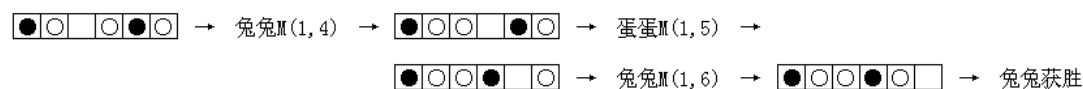


图 2

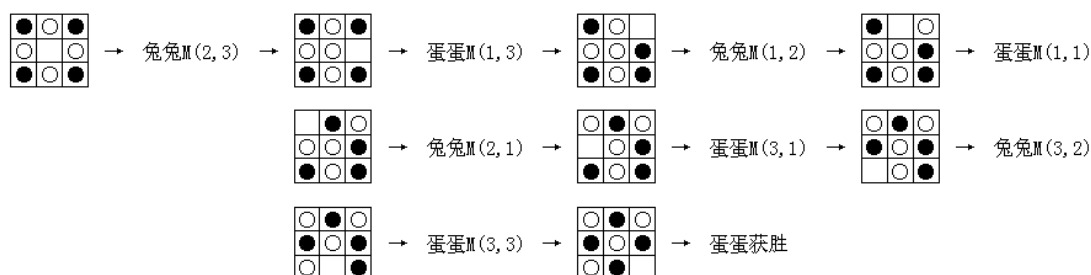


图 3

最近兔兔总是输掉游戏，而且蛋蛋格外嚣张，于是兔兔想请她的好朋友——你——来帮助她。她带来了一局输给蛋蛋的游戏的实录，请你指出这一局游戏中所有她“犯错误”的地方。

注意：

- 两个格子相邻当且仅当它们有一条公共边。
- 兔兔的操作是“犯错误”的，当且仅当，在这次操作前兔兔有必胜策略，而这次操作后蛋蛋有必胜策略。

【输入格式】

从文件 *game.in* 中读入数据。

输入的第一行包含两个正整数 n 、 m 。

接下来 n 行描述初始棋盘。其中第 i 行包含 m 个字符，每个字符都是大写英文字母“X”、大写英文字母“O”或点号“.”之一，分别表示对应的棋盘格中有黑色棋子、有白色棋子和没有棋子。其中点号“.”恰好出现一次。

接下来一行包含一个整数 k ($1 \leq k \leq 1000$)，表示兔兔和蛋蛋各进行了 k 次操作。

接下来 $2k$ 行描述一局游戏的过程。其中第 $2i-1$ 行是兔兔的第 i 次操作（编号为 i 的操作），第 $2i$ 行是蛋蛋的第 i 次操作。每个操作使用两个整数 x,y 来描述，表示将第 x 行第 y 列中的棋子移进空格中。

输入保证整个棋盘中只有一个格子没有棋子，游戏过程中兔兔和蛋蛋的每个操作都是合法的，且最后蛋蛋获胜。

【输出格式】

输出到文件 *game.out* 中。

输出文件的第一行包含一个整数 r ，表示兔兔犯错误的总次数。

接下来 r 行按递增的顺序给出兔兔“犯错误”的操作编号。其中第 i 行包含一个整数 a_i 表示兔兔第 i 个犯错误的操作是他在游戏中的第 a_i 次操作。

【输入样例 1】

```
1 6
XO.OXO
1
1 2
1 1
```

【输出样例 1】

```
1
1
```

【输入样例 2】

```
3 3
XOX
O.O
XOX
4
2 3
1 3
1 2
1 1
2 1
3 1
3 2
3 3
```

【输出样例 2】

```
0
```

【输入样例 3】

```
4 4
OOXX
OXXO
OO.O
XXXO
2
3 2
2 2
1 2
1 3
```

【输出样例 3】

```
2
1
2
```

【样例说明】

样例 1 对应图一中的游戏过程。

样例 2 对应图三中的游戏过程。

【数据规模】

所有测试数据的范围和特点如下表所示

测试点编号	n 的规模	m 的规模
1	$n = 1$	$1 \leq m \leq 20$
2		
3	$n = 3$	$m = 4$
4	$n = 4$	$m = 4$
5		
6	$n = 4$	$m = 5$
7		
8	$n = 3$	$m = 7$
9	$n = 2$	$1 \leq m \leq 40$
10		
11		
12		
13		
14		
15	$1 \leq n \leq 16$	$1 \leq m \leq 16$
16		
17	$1 \leq n \leq 40$	$1 \leq m \leq 40$
18		
19		
20		