

# NOIP2016模拟赛

## 一. 题目概况

中文题目名称	挖金矿	道路规划	排队
英文题目与子目录名	gold	road	queue
可执行文件名	gold	road	queue
输入文件名	gold.in	road.in	queue.in
输出文件名	gold.out	road.out	queue.out
每个测试点时限	1 秒	1 秒	1 秒
测试点数目	10	10	20
每个测试点分值	10	10	5
附加样例文件	有	有	有
结果比较方式	全文比较（过滤行末空格及文末回车）		
题目类型	传统	传统	传统
运行内存上限	512M	512M	512M

## 二. 提交源程序文件名

对于 C++语言	gold.cpp	road.cpp	queue.cpp
对于 C 语言	gold.c	road.c	queue.c
对于 pascal 语言	gold.pas	road.pas	queue.pas

## 1. 挖金矿

(gold.cpp/c/pas)

### 【问题描述】

矿工吉丽得到了一个任务：挖金矿！

这是一个深度为 $h$ ，宽度为 $n$ 的矿场。吉丽站在地面上，第 $i$ 层第 $j$ 列有价值为 $a[i][j]$ 的金矿。如图是一个 $h \times n$ 的矩阵，左上角为 $(1, 1)$ ，右下角为 $(h, n)$ 。

4	5	2	3
3	1	6	2
3	6	1	9

对于每一列，吉丽可以选择往下挖 $k$  ( $1 \leq k \leq h$ ) 层。

吉丽希望挖到的金矿的平均价值最大，请你告诉他答案。

### 【输入格式】

输入文件名gold.in

第一行，两个整数， $n$ ， $h$ 。

接下来 $n$ 行，每行 $h$ 个数，第 $i+1$ 行第 $j$ 个数表示的是 $a[j][i]$ ；

### 【输出格式】

输出文件名为 gold.out。

输出挖出金矿的最大平均价值，保留4位小数。

### 【输入输出样例 1】

gold.in	gold.out
4 3 4 3 3 5 1 6 2 6 1 3 2 9	4.4286

见选手目录下的 gold/gold1.in 和 gold/gold1.ans。

### 【输入输出样例 2】

见选手目录下的 gold/gold2.in 和 gold/gold2.ans。

### 【输入输出样例说明】

如图是一种合法方案，吉丽决定每一列分别挖1, 2, 3层。涂色的格子代表吉丽选

择挖的金矿，价值的平均值为  $(4+5+2+6+3+2+9)/7=31/7 \approx 4.4286$ 。

### 【数据规模与约定】

对于30%的数据， $h \times n \leq 100$ ,  $1 \leq a[i][j] \leq 100$ ;

另有10%的数据， $n=1$ ;

对于100%的数据， $h \times n \leq 10^5$ ,  $1 \leq a[i][j] \leq 10^9$ 。

## 2. 道路规划

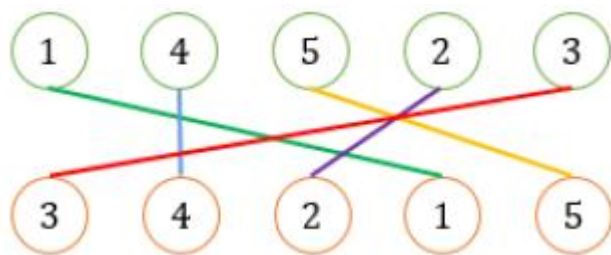
(road.cpp/c/pas)

### 【问题描述】

吉丽王国  $n$  个城市，每个城市有两个“附属城市”，其中北部有  $n$  个城市，每个城市的编号都是  $1 \sim n$  中的一个，且互不相同，南部的  $n$  个城市也是如此。

很遗憾，南北两边的城市之间还没有道路连接，这给交通运输带来了很大的麻烦。

国王吉丽设计规划了一种方案，决定先建  $n$  条道路，即编号相同的城市之间连上一条道路。



如图所示，这就是建完道路后的样子。

如果第  $i$  个城市和第  $j$  个城市的两条道路相交了，那我们称这两个城市“平等互惠”。

吉丽想找出一个城市集合，使得集合中的任意两城市都是“平等互惠”。吉丽还希望这个集合的大小尽可能大。

那最大是多少呢，请你回答他。

### 【输入格式】

输入文件名road.in

第一行，一个整数， $n$ 。

第二行， $n$  个整数，第  $i$  个数表示北部第  $i$  个城市的编号。

第三行， $n$  个整数，第  $i$  个数表示南部第  $i$  个城市的编号。

### 【输出格式】

输出文件名为 road.out。  
输出合法的城市集合的最大大小。

### 【输入输出样例 1】

road.in	road.out
5 1 4 5 2 3 3 4 2 1 5	3

见选手目录下的 road/road1.in 和 road/road1.ans。

### 【输入输出样例 2】

见选手目录下的 road/road2.in 和 road/road2.ans。

### 【输入输出样例说明】

{1, 3, 4}, {1, 2, 3} 和 {2, 3, 5} 都是合法的方案，大小是3，显然是最大值。

### 【数据规模与约定】

对于 20%的数据， $n \leq 20$ ；  
对于 50%的数据， $n \leq 500$ ；  
对于 70%的数据， $n \leq 5000$ ；  
对于 100%的数据， $n \leq 10^5$ ；

## 3. 排队

(queue.cpp/c/pas)

### 【问题描述】

想膜拜吉丽的人太多，只能排队一个一个来了。

想膜拜吉丽的人必须先经过一个树形结构的迷宫，迷宫由  $n$  个小房间组成，有  $n-1$  条走廊。迷宫的入口只有一个，就是 1 号房间，而叶子节点的房间最接近吉丽爷爷。

有趣的是一个房间只能容纳一个人。

一开始，每个房间内都是没有人的。

每个人走路的策略都是一样的：

因为他们都迫不及待的相见吉丽，所以要尽可能的靠近吉丽，只要他能继续往前（前方有空着的房间），他就会走进去。如果同时又多个空位，他会选择编号最小的房间走进去。

接下来  $t$  个时刻，每个时刻都会发生下列两种事情的一种：

1. 有  $x$  ( $1 \leq x \leq \text{剩余空房间的个数}$ ) 个人前来膜拜吉丽。  
这  $x$  个人将依次进入 1 号房间，按照策略尽可能靠近吉丽。  
对于这种情况下，请输出第  $x$  个人，最后停下来的房间编号。
2. 吉丽爷爷突然心情大好派人把第  $x$  ( $1 \leq x \leq n$  且  $x$  房间一等不为空) 号房间的等待的人直接接到吉丽房间，接受他的膜拜。  
这个人被带走后，他后面的人依次按照策略补上来。  
对于这种情况，请输出有多少人在这个人被带走之后移动过。

### 【输入格式】

输入文件名 queue.in

第一行，两个整数， $n$ ， $t$ 。

接下来  $t$  行，每行两个整数  $op, x$ 。 $op=1$  或  $2$ ，表示两种情况的一种。

### 【输出格式】

输出文件名为 queue.out。

输出  $t$  行，表示对应情况的答案。

### 【输入输出样例 1】

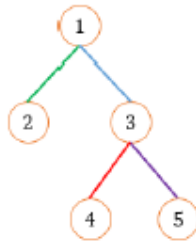
queue.in	queue.out
5 4	3
1 2	1
1 3	1
3 4	2
3 5	
1 4	
2 4	
1 2	
2 5	

见选手目录下的 queue/queue1.in 和 queue/queue1.ans。

### 【输入输出样例 2】

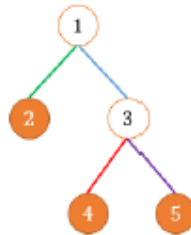
见选手目录下的 queue/queue2.in 和 queue/queue2.ans。

### 【输入输出样例说明】



样例是这样一个树形结构。

1. 有 4 个人走了进来。
  - a) 第一个人进来，发现 2,3 都能走，选择编号较小的 2 号房间，然后停了下来；
  - b) 第二个人进来，发现只能走 3，然后发现 4,5，都能走，选择编号较小的 4 号房间，然后停了下来；
  - c) 第三个人进来，只能一路走到 5 了；
  - d) 第四个人进来，走到 3 就只能停下了。  
第四个人停在了 3 号房间，输出 3。
2. 4 号房间的人被接走了。
  - a) 那么 3 号房间的人为了更加接近吉丽，补到了 4 号房间；  
只有这个一个人移动了，输出 1。



这是第二个时刻过后的情况，涂色的房间表示有人。

3. 有 2 个人走了进来。
  - a) 第一个人走进了 3 号房间；
  - b) 第二个人走到 1 号房间不能再走了，只能停下。  
第二个人停在了 1 号房间，输出 1。
4. 5 号房间的人被接走了。
  - a) 3 号房间的人走了下来，补到 5 号房间；
  - b) 1 号房间的人走了下来，补到 3 号房间。  
有两人移动了，输出 2。

### 【数据规模与约定】

- 对于 10% 的数据， $n \leq 10$ ， $t \leq 10$ ；
- 对于 30% 的数据， $n \leq 100$ ， $t \leq 100$ ；
- 对于 50% 的数据， $n \leq 2000$ ， $t \leq 2000$ ；
- 另有 10% 的数据，数据是一条链。
- 另有 10% 的数据，数据随机生成。
- 对于 100% 的数， $n \leq 10^5$ ， $t \leq 10^5$