

# CSP 模拟赛

-By 华中师大一附中

2019 年 x 月 xx 日 8:30~12:00

## 一. 题目概况

中文题目名称	中位数	互连网	技能树
英文题目名称	middle	network	tree
可执行文件名	middle.cpp	network.cpp	tree.cpp
输入文件名	middle.in	network.in	tree.in
输出文件名	middle.out	network.out	tree.out
每个测试点时限	2 秒	2 秒	1 秒
内存上限	512MB	512MB	512MB
测试点数目	20	20	20
每个测试点分值	5	5	5
结果比较方式	全文比较	spj	全文比较
题目类型	传统	传统	传统

## 二. 编译命令

对于 C++ 语言	<pre>g++ middle.cpp -o -O2 middle</pre>	<pre>g++ network.cpp -o -O2 network</pre>	<pre>g++ tree.cpp -o -O2 tree</pre>
-----------	---	---	-------------------------------------

---

# 中位数(middle)

## 1.1 Description

中位数在统计中拥有十分重要的作用，比如我们取每次模拟赛的成绩中位数很快就能知道这场考试的难度。

在本题中我们的中位数和普通的中位数略有不同，一个长为  $x$  的连续序列的中位数定义为它所有元素中第  $(\lfloor \frac{x}{2} \rfloor + 1)$  大的那一项。

现在给定一个长为  $n$  的序列  $a$  (序列从 1 开始标号)，你要回答  $m$  次询问，每次给定  $l_1, r_1, l_2, r_2$  询问左端点在  $[l_1, r_1]$  且右端点在  $[l_2, r_2]$  所有连续子序列的中位数中最大的一个是多少，数据保证  $l_2 > r_1$ 。

## 1.2 Input

第一行一个整数  $n$  意义如上。

接下来一行  $n$  个数，表示  $a$  序列中的元素。

接下来一行一个数  $m$ ，表示询问的个数。

接下来  $m$  行，每行 4 个整数  $l_1, r_1, l_2, r_2$ ，表示一次询问。

## 1.3 Output

共  $m$  行，第  $i$  行的数表示第  $i$  次询问的答案。

## 1.4 Sample

Sample 1:

middle.in	middle.out
6	7
7 5 9 6 5 3	6
5	9
1 1 4 4	6
1 4 6 6	9

---

1 2 3 3 2 3 5 6 1 2 3 5	
-------------------------------	--

这 5 次询问对应的答案区间分别是：

[1, 4] 排序后结果为：5 6 7 9 中位数为 7

[3, 6] 排序后结果为：3 5 6 9 中位数是 6

[2, 3] 排序后结果为：5 9 中位数是 9

[3, 5] 排序后结果为：5 6 9 中位数为 6

[2, 3] 排序后结果为：5 9 中位数为 9

Sample 2:

见下发文件

## 1.5 Data Range

20%的数据满足  $n \leq 100$

40%的数据满足  $n \leq 300$

另有 20%的数据满足  $a_i = 0$  或 1

对于全部数据，满足  $n, m \leq 50000, 0 \leq a_i \leq 10^9$

---

# 互连网 (network)

## 1.3 Description

9102 年, “互连网” 在生活中已无处不在, 人类不能忍受哪怕一秒钟没有互连网的生活。

一张网是“互连”的, 当且仅当网上任意两点都可以互相到达。华懿高级中学也拥有自己的互连网络, 但是本着勤俭节约的原则, 华懿高级中学内部  $n$  个点的互连网仅由  $n-1$  条无向边连接, 并且任意两点之间有且仅有一条简单路径。

但是这样有很大的弊端, 若某天某一条边突然断开, 整个网络就不再联通了, 于是华懿高级中学的领导们决定翻新它。依旧本着勤俭节约的原则, 他们希望添加最少的边, 使得不管断开哪条边, 整个网络依旧是“互连”的。

请告诉华懿高级中学, 最少要添加几条边, 并给出一组合法的方案。

## 1.4 Input

第一行两个整数  $n$ , 意义如上。

接下来  $n$  行, 每行两个整数  $x_i, y_i$ , 表示一条原网络的边。

## 1.3 Output

第一行一个数  $n$ , 表示至少要添加几条新的边。

接下来  $n$  行, 每行两个整数  $x_i, y_i$ , 描述一条新添加的边。

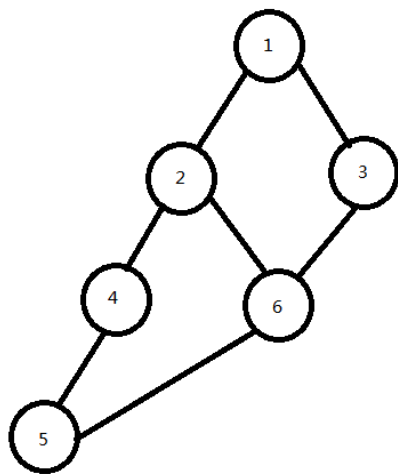
## 1.4 Sample

Sample 1:

network.in	network.out
6	2
1 2	3 6
1 3	6 5

2 4 4 5 2 6	
-------------------	--

加了边之后的图如下：



可以验证该图中断开任意一条边图依旧是联通的，且加边最少。

Sample 2:

见下发文件

## 1.5 Data Rang

本题评测方式为捆绑评测，你必须通过该子任务的全部测试点才能获得该子任务的分数。

子任务 1 (20pts)：  $n \leq 10$

子任务 2 (10pts)：数据满足所有边的  $x_i = 1$

子任务 3 (20pts)：  $n \leq 1000$

子任务 4 (50pts)：数据满足  $n \leq 100000$

---

# 技能树 (tree)

## 1.6 Description

“OI 的学习，理应循序渐进”。

OI 中技能的加点形如一棵以 1 为根的树形结构，每个点的父亲就是它的前置知识，比如“min25 筛”应该在“莫比乌斯反演”之后学习，“主席树”应该在“树状数组”之后学习，形式化的来说，对于树上每个节点  $u$ ，必须先学前置知识，也就是它在技能树上父亲  $fa(u)$ 。

学习知识的过程可能是痛苦的，也可能是愉悦的，学习  $i$  号知识需要消耗  $a_i$  点愉悦度，学完之后因为满满的成就感会增加  $b_i$  点愉悦度，注意，此处先结算  $a_i$  后结算  $b_i$ ，二者并非同一时刻结算。

现在，我们已经学习完了 1 号知识点，并拥有了一个未知的初始愉悦度，因为我们不希望学习过程中出现过分痛苦的情况，所以我们想知道：至少需要多少点初始愉悦度，使得能通过合理的学习顺序学习完技能树上所有知识，并且学习过程中任意时刻愉悦度都不为负？

## 1.7 Input

第一行 1 个整数， $n$  意义如上。

接下来  $n-1$  行，每行两个整数  $a_i, b_i$  表示 2、3、4... $n$  号节点学习所消耗的愉悦度和学完后增加的愉悦度。

在接下来  $n-1$  行，每行两个整数  $x_i, y_i$ ，表示一条连接  $x_i$  与  $y_i$  的边。

## 1.3 Output

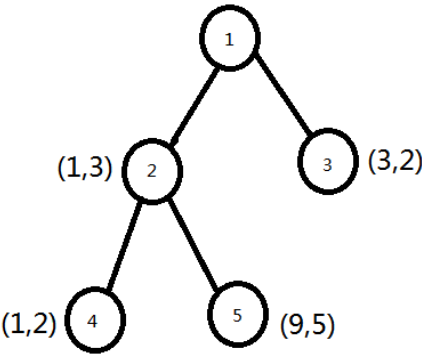
一个整数，表示最少需要多少点初始愉悦度。

# 1.4 Sample

Sample 1:

tree.in	tree.out
5 1 3 3 2 1 2 9 5 2 1 3 1 4 2 5 2	6

该技能树如图，其中括号内的数字表示  $a_i$  和  $b_i$  (左边为  $a$ ，右边为  $b$ )。



可以证明按照 2->4->5->3 的顺序依次学习是最优的，仅需要 6 点初始愉悦度就可以保证在学习过程中任意时刻愉悦度不为负。

以下是该过程的模拟：

- 初始愉悦度是 6
- 拿 2 号点，先-1 变成 5，再+3 变成 8
- 拿 4 号点，先-1 变成 7，再+2 变成 9
- 拿 5 号点，先-9 变成 0，再+5 变成 5
- 拿 3 号点，先-3 变成 2，再+2 变成 4

---

在该过程中，最低愉悦度为 0，满足“时刻不为负数”的要求。  
而其它方案都不会更优，下面给出一组其它方案的例子模拟，拿取顺序为 2→3→4→5：

设初始值仍为 6。

拿 2 号点，先-1 变成 5，再+3 变成 8

拿 3 号点，先-3 变成 5，再+2 变成 7

拿 4 号点，先-1 变成 6，后+2 变成 8

此时因为  $a_5=9$ ，所以此时无法拿 5 号节点，故该方案不合法，可以发现要使得这个选取顺序合法，必须要至少 7 点初始愉悦度。

Sample 2:

见下发文件

## 1.5 Data Range

对于 10%的数据，满足  $n \leq 10$

另有 20%的数据满足所有节点与 1 号节点相连。

另有 20%的数据满足对于每个节点  $i$ ,  $a_i \leq b_i$

另有 20%的数据满足对于每个节点  $i$ ,  $a_i > b_i$

对于全部测试点，满足  $n \leq 100000$ ,  $0 \leq a_i, b_i \leq 10^9$