

NOIP2016 模拟试题

提高组day1

(请选手务必仔细阅读本页内容)

一. 题目概况

中文题目名称	二叉树	列车调度	保留道路
英文题目与子目录名	binary	manage	road
可执行文件名	binary	manage	road
输入文件名	binary.in	manage.in	road.in
输出文件名	binary.out	manage.out	road.out
每个测试点时限	1 秒	1 秒	1 秒
测试点数目	10	10	10
每个测试点分值	10	10	10
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	128M	128M	128M

二. 提交源程序文件名

对于C++语言	binary.cpp	manage.cpp	road.cpp
对于C语言	binary.c	manage.c	road.c
对于pascal语言	binary.pas	manage.pas	road.pas

三. 编译命令（不包含任何优化开关）

对于C++语言	g++ -o binary binary.cpp -lm	g++ -o manage manage.cpp -lm	g++ -o road road.cpp -lm
对于C语言	gcc -o binary binary.c -lm	gcc -o manage manage.c -lm	gcc -o road road.c -lm
对于pascal语言	fpc binary.pas	fpc manage.pas	fpc road.pas

注意事项：

- 1、文件名（程序名和输入输出文件名）必须使用英文小写。
- 2、C/C++中函数 main() 的返回值类型必须是int，程序正常结束时的返回值必须是 0。
- 3、全国统一评测时采用的机器配置为：CPU AMD Athlon(tm) II x2 240 processor, 2.8GHz，内存4G，上述时限以此配置为准。
- 4、只提供 Linux 格式附加样例文件。
- 5、特别提醒：评测在当前最新公布的 NOI Linux 下进行，各语言的编译器版本以其为准。

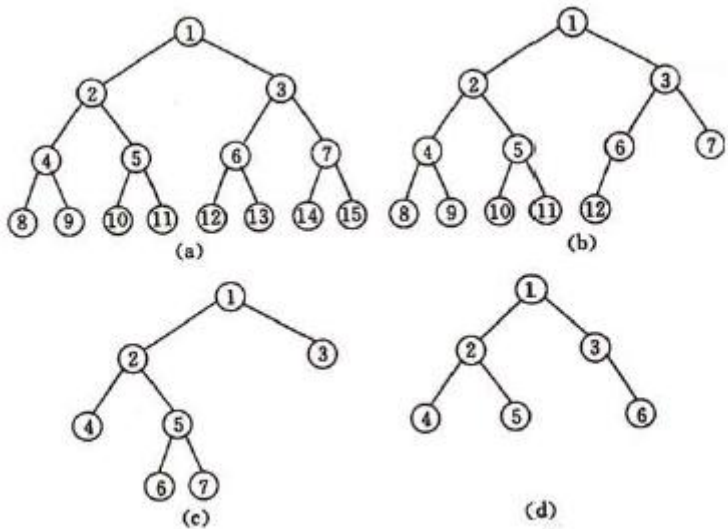
1. 二叉树
(binary.cpp/c/pas)

【问题描述】

二叉排序树或者是一棵空树，或者是具有下列性质的二叉树：

- (1) 若左子树不空，则左子树上所有结点的值均小于它的根结点的值；
- (2) 若右子树不空，则右子树上所有结点的值均大于它的根结点的值；
- (3) 左、右子树也分别为二叉排序树；
- (4) 没有键值相等的结点。

完全二叉树：只有最下面的两层结点度能够小于2，并且最下面一层的结点都集中在该层最左边的若干位置的二叉树。



上图中，(a)和(b)是完全二叉树，(c)和(d)是非完全二叉树。

给出N个数，且这N个数构成1至N的排列。现在需要你按顺序构建一棵二叉排序树，并按照层次遍历的方式输出它，然后判断它是否是一棵完全二叉树。

【输入格式】

输入文件名为binary.in。

输入文件包含两行。第一行为一个正整数N；第二行为1至N的排列。

【输出格式】

输出文件名为binary.out。

输出文件包含两行。第一行为构建出的二叉排序树的层次遍历；第二行判断是否是完全二叉树：若是输出yes，否则输出no。

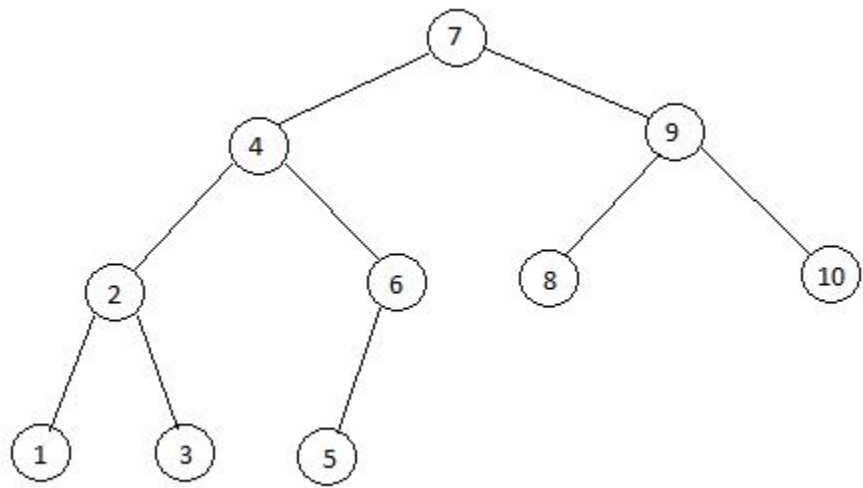
【输入输出样例1】

binary.in	binary.out
10	7 4 9 2 6 8 10 1 3 5
7 9 8 4 6 2 10 1 5 3	yes

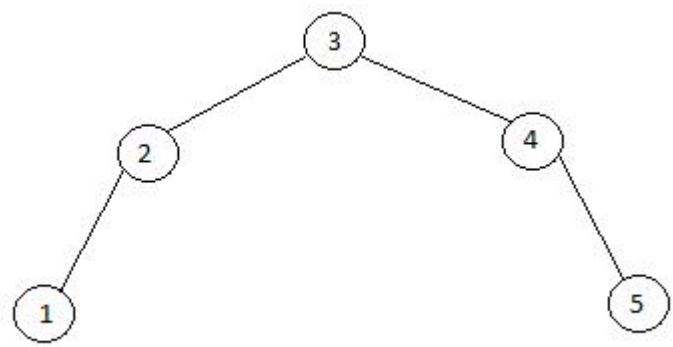
【输入输出样例2】

binary.in	binary.out
5 3 4 5 2 1	3 2 4 1 5 no

样例1:



样例2:



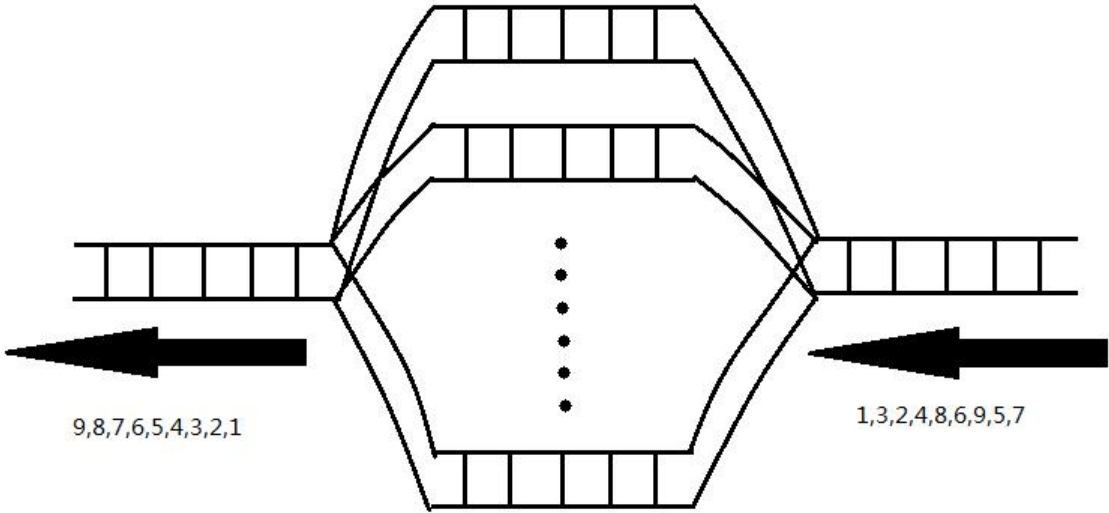
【数据规模与约定】

对于100%的数据， $1 \leq N \leq 20$ 。

2. 列车调度
(manage. cpp/c/pas)

【问题描述】

有N辆列车，标记为1, 2, 3, ..., N。它们按照一定的次序进站，站台共有K个轨道，轨道遵从先进先出的原则。列车进入站台内的轨道后可以等待任意时间后出站，且所有列车不可后退。现在要使出站的顺序变为N, N-1, N-2, ..., 1，询问K的最小值是多少。



例如上图中进站的顺序为1, 3, 2, 4, 8, 6, 9, 5, 7，则出站的顺序变为9, 8, 7, 6, 5, 4, 3, 2, 1。

【输入格式】

- 输入文件名为manage. in。
- 输入共2行。
- 第 1 行包含1个正整数N，表示N辆列车。
- 第 2 行包含N个正整数，为1至N的一个排列，表示进站次序。

【输出格式】

- 输出文件名为manage. out。
- 输出共1行，包含1个整数，表示站台内轨道数K的最小值。

【输入输出样例1】

manage. in	manage. out
3	3
1 2 3	

【输入输出样例2】

manage. in	manage. out
9	5
1 3 2 4 8 6 9 5 7	

【数据规模与约定】

对于 30%的数据， $N \leq 10$ ；

对于 70%的数据， $N \leq 2000$ ；

对于 100%的数据， $N \leq 100000$ 。

3. 保留道路

(road.cpp/c/pas)

【问题描述】

很久很久以前有一个国家，这个国家有 N 个城市，城市由 $1, 2, 3, \dots, N$ 标号，城市间有 M 条双向道路，每条道路都有两个属性 g 和 s ，两个城市间可能有多条道路，并且可能存在将某一城市与其自身连接起来的道路。后来由于战争的原因，国王不得不下令减小花费从而关闭一些道路，但是必须要保证任意两个城市相互可达。

道路花费的计算公式为 $wG * \max\{\text{所有剩下道路的属性}g\} + wS * \max\{\text{所有剩下道路的属性}s\}$ ，其中 wG 和 wS 是给定的值。国王想要在满足连通性的前提下使这个花费最小，现在需要你计算出这个花费。

【输入格式】

输入文件名为road.in。

第一行包含两个正整数 N 和 M 。

第二行包含两个正整数 wG 和 wS 。

后面的 M 行每行描述一条道路，包含四个正整数 u, v, g, s ，分别表示道路连接的两个城市以及道路的两个属性。

【输出格式】

输出文件名为road.out。

输出一个整数，表示最小花费。若无论如何不能满足连通性，输出-1。

【输入输出样例】

road.in	road.out
3 3 2 1 1 2 10 15 1 2 4 20 1 3 5 1	30

【数据规模与约定】

对于 10%的数据， $N \leq 10$ ， $M \leq 20$ ；

对于 30%的数据， $N \leq 100$ ， $M \leq 1000$ ；

对于 50%的数据， $N \leq 200$ ， $M \leq 5000$ ；

对于 100%的数据， $N \leq 400$ ， $M \leq 50000$ ， $wG, wS, g, s \leq 1000000000$ 。