

2015 株洲二中高一 6 月模拟赛

一、题目概况

中文题目名称	质因数分解	NBA 总冠军	出栈序列统计	数塔
子目录名	prime	nba	stack1	shuta
可执行文件名	prime.exe	nba.exe	stack1.exe	shuta.exe
每个测试点时限	1000 毫秒	1000 毫秒	1000 毫秒	1000 毫秒
测试点数目	10	10	20	10
每个测试点分值	10	10	5	10
附加样例文件	无	无	无	无
结果比较方式	全文比较			
题目类型	传统	传统	传统	传统
运行内存上限	128 兆字节	128 兆字节	128 兆字节	128 兆字节

二、提交源程序文件名

对于 C 语言	prime.c	nba.c	stack1.c	shuta.c
对于 C++语言	prime.cpp	nba.cpp	stack1.cpp	shuta.cpp

三、编译命令

对于 C 语言	gcc -Wall -std=c99 -O2 -DOJ -o foo src.c -lm
对于 C++语言	g++ -Wall -std=c++11 -O2 -DOJ -o foo src.cpp -lm

注意事项:

- 1、比赛使用标准输入输出。
- 2、文件名必须使用英文小写。
- 3、C/C++中函数 main 的返回值类型必须是 int，程序正常结束时返回值必须是 0。
- 4、评测时采用的机器配置为：Intel Core i5-3470 3.2 GHz × 2 处理器，8GB 内存。上述时限以此配置为准。
- 5、特别提醒：评测在 Ubuntu Server 14.04 LTS x64 操作系统上进行，各语言的编译器版本以其为准。GCC 版本目前为 4.8.2。

1、质因数分解

【问题描述】

已知正整数 n 是两个不同的质数的乘积，试求出较大的那个质数。

【输入】

输入只有一行，包含一个正整数 n 。

【输出】

输出只有一行，包含一个正整数 p ，即较大的那个质数。

【输入输出样例】

Input	Output
21	7

样例输出说明：

$21=3*7$ ， $7>3$ ，故输出 7。

【数据说明】

对于 60% 的数据， $6 \leq n \leq 1000$ 。

对于 100% 的数据， $6 \leq n \leq 5*10^{12}$ 。

2、NBA 总冠军

【问题描述】

又要考试了，Ljw 决定放松一下，就打开电视，看见了篮球赛，他立即想到了每年的 NBA 总冠军队伍。由于复习紧张，他只记起了一部分，记忆的内容是正确的，可能不是按时间顺序排列的，记忆的内容可能有重复。

现在请求学过编程的你帮助 Ljw，按时间顺序依次输出总冠军的球队(不能重复)。

(NBA 从 1947A. D 到 2009A. D)

【输入】

第一行是一个整数 $n(0 < n < 100)$ 。

接下来的 n 行，每行先是城市名(由大小写字母、空格组成)，后是时间(由数字组成)。二者之间用空格隔开。

【输出】

共 n 行，即排序后的 NBA 总冠军队伍。每行先是时间，后是城市名。

【输入输出样例】

Input	Output
3	1947 Philly
Boston 1963	1959 Boston
Boston 1959	1963 Boston
Philly 1947	

【数据说明】

$0 < n < 100$

3、出栈序列统计

【问题描述】

栈是常用的一种数据结构，有 n 个元素在栈顶端一侧等待进栈，栈顶端另一侧是出栈序列。你已经知道栈的操作有两种：push 和 pop，前者是将一个元素进栈，后者是将栈顶元素弹出。现在要使用这两种操作，由一个操作序列可以得到一系列的输出序列。请你编程求出对于给定的 n ，计算并输出由操作数序列 1, 2, ..., n ，经过一系列操作可能得到的输出序列总数。

【输入】

一个整数 n ($1 \leq n \leq 15$ (50%), $1 \leq n \leq 1000$ (100%))

【输出】

一个整数，即可能输出序列的总数目。

【输入输出样例】

Input	Output
3	5

【数据说明】

50%数据 $1 \leq n \leq 15$

100%数据 $1 \leq n \leq 1000$

4、数塔

【问题描述】

设有一个三角形的数塔，顶点结点称为根结点，每个结点有一个整数数值（小于 3000）。从顶点出发，可以向左走，也可以向右走。如图所示：从根结点 13 出发，向左走到达 11，再向右走到达 7，再向左走到达 14，再向左走到达 7。由于 7 是最底层，无路可走。此时，我们找到一条从根结点开始到达底层的路径：

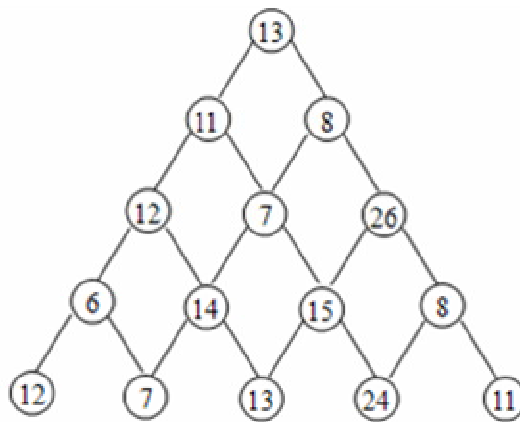
13--11--7--14--7

路径上结点中数字的和称为路径的值，如上面路径的值为 $13+11+7+14+7=52$ 。同时，从图中可以看到除了上述路径外，还存在其他的路径，如：

13--11--12--14--13

其路径的值为 63。

问题：当三角形数塔给出后，找出一条路径，使路径上的值为最大。若这样的路径存在多条，



选根部偏左的路径。

【输入】

输入由若干行组成，第一行有一个整数， n ($1 \leq n \leq 80$)； n 表示数塔的层数。第 2 至 $n+1$ 行是数塔的信息。

【输出】

输出由两行组成，第一行有一个整数，为所选路径的值。

第二行有 n 个整数，为所选路径，中间用一个空格隔开。

【输入输出样例】

Input	Output
5	86
13	13 8 26 15 24
11 8	
12 7 26	
6 14 15 8	
12 7 13 24 11	

【数据说明】

$1 \leq n \leq 80$

结点的整数值 ≤ 3000