

2022 年CFS非专业级软件能力认证第二轮

入门级模拟训练

时间：2022 年 10 月 05 日 13: 00 - 16: 30

中文题目名称	智商测试	灵石采集	足球队	波浪数
英文题目与子目录名	test	collect	footballteam	wave
可执行文件名	test	collect	footballteam	wave
输入文件名	test.in	collect.in	footballteam.in	wave.in
输出文件名	test.out	collect.out	footballteam.out	wave.out
每个测试点时限	1 秒	1 秒	1 秒	1 秒
测试点数目	10	10	12	10
每个测试点分值	10	10	8	10
附加样例文件	有	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	传统
运行内存上限	512M	512M	512M	512M

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. 提交的程序代码文件的放置位置请参照各省的具体要求。
4. 因违反以上三点而出现的错误或问题，申诉时一律不予受理。

智商测试

(test.cpp/c/pas)

【问题描述】

PBC同学智商为 $-1e9$ ，但是他既不谦虚又不低调，还很喜欢炫耀他的智商。作为他的资深杠友GPD非常看不惯PBC的这种行为，于是GPD决定测试一下PBC的真实智商了，他出了一个简单题让PBC回答，题目是这样的：

已知一个 n 乘以 n 的矩阵，有一个小球一开始在 $(1, 1)$ 处，小球每秒都会向右走一格，如果到达最右边，小球就会掉落在下一行的第一格。如果到达 (n, n) 则返回 $(1, 1)$ 。

现在给定 n 和时间 t ，输出小球最后所在的行和列。GPD希望大家写个程序PBC比拼一下IQ。

【输入格式】

一行两个正整数， n 和 t 。

【输出格式】

输出两个整数代表小球最终的行和列。

【输入输出样例 1】

test.in	test.out
3 5	2 3

【输入输出样例 2】

test.in	test.out
4 2	1 3

【输入输出样例 3】

test.in	test.out
2 7	2 2

【数据规模与约定】

对于 50%数据保证： $t \leq n^2 - 1$

对于 80% 数据保证： $t \leq n^2$

对于 100% 数据保证： $1 \leq n \leq 10, 0 \leq t \leq 6000$

题目数据比较毒瘤，请考虑所有情况。

灵石采集

(collect.cpp/c/pas)

【问题描述】

小C编程编累了，于是他趁老Q去办公室倒水的期间偷偷打开了一个有趣的游戏。在游戏中，灵石是一种非常重要的资源。每个玩家每天有且仅有一次采集灵石的机会。

灵石会在许多灵岛上出现，每个灵岛上灵石出现数量可能不同。这些灵岛之间通过传送法阵相连，激活每个岛屿上的传送法阵花费的灵石数量也不同。玩家可以耗费 m_i 块灵石从任意一个其他灵岛或初始平台前往第 i 个灵岛。采集完毕后玩家可以从任何浮岛直接退出地图。

现在，小C手中有着 K 块灵石，他想知道自己今天采集结束后最多能拥有多少块灵石。小C只能趁老Q倒水的时间偷偷玩一会，不然被发现了就麻烦了。他希望你编写一个程序帮他及时算出来。

【输入格式】

第一行两个正整数 N , K 分别表示灵岛的数量和小C手中初始的灵石数量。

接下来 N 行，每行两个正整数 k_i , m_i ，第 i 行的正整数 k_i 表示第 i 个灵岛上今日出现的灵石数量， m_i 表示传送到第 i 个灵岛所需的灵石数量。

【输出格式】

一个正整数，表示小C今天采集后最多能拥有的灵石数量

【输入输出样例 1】

collect.in	collect.out
3 5	8
4 3	
4 3	
4 3	

【输入输出样例 2】

collect.in	collect.out
------------	-------------

2 1 5 2 8 2	1
-------------------	---

【输入输出样例 3】

collect.in	collect.out
2 6 2 3 4 3	7

【数据规模与约定】

对于 30%的数据，满足 $k_i > m_i$

对于 100%的数据，满足 $0 < N, K \leq 10^5, 0 < k_i, m_i \leq 10^9$

保证答案在int范围内。

足球队

(footballteam.cpp/c/pas)

【问题描述】

为了强身健体，老Q准备组织CFSOIer的学生组成一支足球队，众所周知，一只足球队由11球员组成，其中有一名守门员和一名队长。

现在，有22人组成的一个候选名单，从中要选出11人参加比赛。选拔规则如下：

首先提供一个阵型，例如，4-4-2。第一个4表示有4名后卫，第二个4表示有4名中场，第三个2表示有2名前锋。当然，还有一个位置就是留给守门员的。

每个人都有自己唯一能打的位置（前锋、中场、后卫、守门员）。在每个位置上，以球员的编号为关键字，从小到大依次选取。

选出球队后还要选出队长。队长是球队中参加比赛次数最多的球员。如果有并列，取编号较大的球员做队长

【输入格式】

前22行每行按如下格式表示出一名球员的信息：

Number Name Role year1-year' 1 year2-year' 2 ...

整数Number(≤ 100)是该球员编号。字符串Name($\text{length} \leq 20$)是该球员名字。字符Role是该球员所能打的位置（S前锋、M中场、D后卫、G守门员）。每一对yeari-year' i($\text{yeari} \leq \text{year}' i$)表示该球员效力的时间段，例如2001-2002表示该球员效力了两年2001年和2002年。至少有1对年份，最多20对年份。年份是一个四位数。同一个年份不会出现在两对年份中。球员的编号在球队中是唯一的。

第23行给出一个阵型，例如4-4-2。保证阵型有且仅有3个正整数组成，且和等于10。例如，4-3-2-1是不合法的。

【输出格式】

输出选出的11名球员，每一行包括球员编号，姓名，场上所打位置。之间用一个空格隔开。球员按守门员、后卫、中场、前锋的顺序排序，相同位置按编号从小到大排序。如果一名球员是队长，那么无论他打什么位置，都放在序列的第一个。如果无法找出11个满足条件的球员，输出“IMPOSSIBLE TO ARRANGE”。

【输入输出样例 1】

footballteam.in	footballteam.out
9 PlayerA M 2000-2001 2003-2006	7 PlayerM S
2 PlayerB M 2004-2006	15 PlayerP G
10 PlayerC D 2001-2005	1 PlayerD D
1 PlayerD D 2000-2001 2002-2004	3 PlayerU D
11 PlayerE S 2003-2006	6 PlayerI D
8 PlayerF M 2005-2006	10 PlayerC D
22 PlayerG S 2005-2006	2 PlayerB M
25 PlayerH G 2000-2001 2002-2003 2005-2006	4 PlayerZ M
6 PlayerI D 2003-2006	8 PlayerF M
26 PlayerJ D 2003-2004 2000-2001	9 PlayerA M
18 PlayerK M 2003-2004	5 PlayerR S
19 PlayerL M 2000-2001 2003-2006	
7 PlayerM S 2003-2006 1999-2001	

21 PlayerN S 2003-2006 13 PlayerO S 2005-2006 15 PlayerP G 2001-2006 14 PlayerQ D 2003-2004 5 PlayerR S 2000-2005 20 PlayerS G 2000-2002 2003-2003 12 PlayerT M 2004-2005 3 PlayerU D 2000-2005 4 PlayerZ M 2001-2004 4-4-2	
--	--

波浪数

(wave.cpp/c/pas)

【问题描述】

LZX用了两个半小时完成了今天的所有题目，他觉得今天稳稳的能AK所有题，所以他计划给大家明天出一道试题，他想起曾经在DOTA游戏中操刀水人在关键时刻释放了一个波浪技能，扭转了他团队的劣势，反败为胜。为了好好纪念这个技能，他决定让大家帮他一起研究波浪数，波浪数的概念是这样的：对于除首尾位置之外的元素，每一个位置要么比两侧相邻的数字小，要么比两侧相邻的数字大。例如 **【1, 3, 2, 5, 3, 4】** 就是一个波浪数组，而

【2, 3, 4, 1, 2】 则不是，因为第二个位置 3比左边的数字 2大，比右边的数字 4小。

****注意****：根据定义，长度小于等于 2 的数组一定是波浪数组。

现在有一个长度为n数组，每次操作可以将任意一个位置的数字修改成任意一个新数字。

小C想要将其变成一个波浪数组，请问最少需要修改几次？

【输入格式】

输入第一行包含一个正整数n，代表数组长度。

接下来一行包含n个正整数，其中第i个数字表示数组的第 i 个元素 a_i

【输出格式】

一个整数，最少需要修改的次数

【输入输出样例 1】

wave.in	wave.out
6	3
1 1 2 2 3 3	

【输入输出样例 1 说明】

可以将数组改为 $[1, 5, 2, 4, 1, 3]$ 或者 $[1, 0, 2, -1, 5, 3]$ 等，都是波浪数组，其中加粗的数字表示被修改的数字。我们可以发现至少修改三个数字才可以将原来的数组变为波浪数组，所以答案为 3。

【输入输出样例 2】

wave.in	wave.out
11	3
703 702 703 703 702 703 702 702 702 700 702	

【数据范围】

对于 10% 的数据，有 $1 \leq n \leq 20$ 。对于

30% 的数据，有 $1 \leq n \leq 1000$ 。

对于另外 10% 的数据，有 $1 \leq n \leq 10^5$ ，且数组元素各不相同。

对于另外 10% 的数据，有 $1 \leq n \leq 10^5$ ，且数组元素全部相同。

对于 100% 的数据，有 $1 \leq n \leq 10^5$ ， $1 \leq a_i \leq 10^9$ 。