

重庆育才信息学奥林匹克测试试题

提高组

(请选手务必仔细阅读本页内容)

一. 题目概况

中文题目名称	寻宝	组合数问题	货币系统	旅行
英文题目名	treasure	problem	money	travel
可执行文件名	treasure	problem	money	travel
输入文件名	treasure.in	problem.in	money.in	travel.in
输出文件名	treasure.out	problem.out	money.out	travel.out
每个测试点时限	1秒	0.2秒	0.2秒	0.2 秒
测试点数目	10	20	20	25
每个测试点分值	10	5	5	4
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
内存上限	128M	512MB	512MB	512MB

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、特别提醒：评测在 NOI Linux 下进行。

1. 寻宝

(treasure.cpp/c/pas)

【问题描述】

传说很遥远的藏宝楼顶层藏着诱人的宝藏。小明历尽千辛万苦终于找到传说中的这个藏宝楼，藏宝楼的门口竖着一个木板，上面写有几个大字：寻宝说明书。说明书的内容如下：

藏宝楼共有 $N+1$ 层，最上面一层是顶层，顶层有一个房间里面藏着宝藏。除了顶层外，藏宝楼另有 N 层，每层 M 个房间，这 M 个房间围成一圈并按逆时针方向依次编号为 $0, \dots, M-1$ 。其中一些房间有通往上一层的楼梯，每层楼的楼梯设计可能不同。每个房间里有一个指示牌，指示牌上有一个数字 x ，表示从这个房间开始按逆时针方向选择第 x 个有楼梯的房间（假定该房间的编号为 k ），从该房间上楼，上楼后到达上一层的 k 号房间。比如当前房间的指示

牌上写着 2，则按逆时针方向开始尝试，找到第 2 个有楼梯的房间，从该房间上楼。如果当前房间本身就有楼梯通向上层，该房间作为第一个有楼梯的房间。

寻宝说明书的最后用红色大号字体写着：“寻宝须知：**帮助你找到每层上楼房间的指示牌上的数字（即每层第一个进入的房间内指示牌上的数字）总和为打开宝箱的密钥**”。

请帮助小明算出这个打开宝箱的密钥。

【输入】

输入文件为 `treasure.in`。

第一行 2 个整数 N 和 M ，之间用一个空格隔开。 N 表示除了顶层外藏宝楼共 N 层楼， M 表示除顶层外每层楼有 M 个房间。

接下来 $N*M$ 行，每行两个整数，之间用一个空格隔开，每行描述一个房间内的情况，其中第 $(i-1)*M+j$ 行表示第 i 层 $j-1$ 号房间的情况 ($i=1, 2, \dots, N$; $j=1, 2, \dots, M$)。第一个整数表示该房间是否有楼梯通往上一层 (0 表示没有，1 表示有)，第二个整数表示指示牌上的数字。**注意，从 j 号房间的楼梯爬到上一层到达的房间一定也是 j 号房间。**

最后一行，一个整数，表示小明从藏宝楼底层的几号房间进入开始寻宝（注：房间编号从 0 开始）。

【输出】

输出文件名为 `treasure.out`。

输出只有一行，一个整数，表示打开宝箱的密钥，这个数可能会很大，请输出对 20123 取模的结果即可。

【输入输出样例】

<code>treasure.in</code>	<code>treasure.out</code>
2 3 1 2 0 3 1 4 0 1 1 5 1 2 1	5

【输入输出样例说明】

第一层：

0 号房间，有楼梯通向上层，指示牌上的数字是 2；

1 号房间，无楼梯通向上层，指示牌上的数字是 3；

2 号房间，有楼梯通向上层，指示牌上的数字是 4；

第二层：

0 号房间， 无楼梯通往上层， 指示牌上的数字是 1；

1 号房间， 有楼梯通往上层， 指示牌上的数字是 5；

2 号房间， 有楼梯通往上层， 指示牌上的数字是 2；

小明首先进入第一层(底层)的 1 号房间， 记下指示牌上的数字为 3， 然后从这个房间 开始， 沿逆时针方向选择第 3 个有楼梯的房间2 号房间进入， 上楼后到达第二层的2 号房间， 记下指示牌上的数字为 2， 由于当前房间本身有楼梯通向上层， 该房间作为第一个有楼梯的 房间。因此， 此时沿逆时针方向选择第 2 个有楼梯的房间即为 1 号房间， 进入后上楼梯到达 顶层。这时把上述记下的指示牌上的数字加起来， 即 $3+2=5$ ， 所以打开宝箱的密钥就是 5。

【数据范围】

对于 50%数据， 有 $0 < N \leq 1000$ ， $0 < x \leq 10000$ ；

对于 100%数据， 有 $0 < N \leq 10000$ ， $0 < M \leq 100$ ， $0 < x \leq 1,000,000$ 。

2、组合数问题 (problem)

【问题描述】

组合数 C_n^m 表示的是从 n 个物品中选出 m 个物品的方案数。举个例子，从 $(1, 2, 3)$ 三个物品中选择两个物品可以有 $(1, 2)$ ， $(1, 3)$ ， $(2, 3)$ 这三种选择方法。根据组合数的定义，我们可以给出计算组合数 C_n^m 的一般公式：

$$C_n^m = \frac{n!}{m!(n-m)!}$$

其中 $n! = 1 \times 2 \times \dots \times n$ 。

小葱想知道如果给定 n, m 和 k ， 对于所有的 $0 \leq i \leq n, 0 \leq j \leq \min(i, m)$ 有多少对 (i, j) 满足 C_i^j 是 k 的倍数。

【输入格式】

从文件 `problem.in` 中读入数据。

第一行有两个整数 t, k ， 其中 t 代表该测试点总共有多少组测试数据， k 的意义见【问题描述】。

接下来 t 行每行两个整数 n, m ， 其中 n, m 的意义见【问题描述】。

【输出格式】

输出到文件 `problem.out` 中。

t 行， 每行一个整数代表所有的 $0 \leq i \leq n, 0 \leq j \leq \min(i, m)$ 中有多少对 (i, j) 满足 C_i^j 是 k 的倍数。

样例

【样例1输入】 1 2 3 3 【样例1输出】 1 【样例1说明】 在所有可能的情况中, 只有 $C_2^1 = 2$ 是 2 的倍数。	【样例2输入】 2 5 4 5 6 7 【样例2输出】 0 7
-------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------

【子任务】

测试点	n	m	k	t
1	s 3	s 3	= 2	= 1
2			= 3	s 10^4
3	s 7	s 7	= 4	= 1
4			= 5	s 10^4
5	s 10	s 10	= 6	= 1
6			= 7	s 10^4
7	s 20	s 100	= 8	= 1
8			= 9	s 10^4
9	s 25	s 2000	= 10	= 1
10			= 11	s 10^4
11	s 60	s 20	= 12	= 1
12			= 13	s 10^4
13	s 100	s 25	= 14	= 1
14			= 15	s 10^4
15		s 60	= 16	= 1
16			= 17	s 10^4
17	s 2000	s 100	= 18	= 1
18			= 19	s 10^4
19		s 2000	= 20	= 1
20			= 21	s 10^4

3. 货币系统

(money.cpp/c/pas)

【问题描述】

在网友的国度中共有 n 种不同面额的货币，第 i 种货币的面额为 $a[i]$ ，你可以假设每一种货币都有无穷多张。为了方便，我们把货币种数为 n 、面额数组为 $a[1..n]$ 的货币系统记作 (n, a) 。

在一个完善的货币系统中，每一个非负整数的金额 x 都应该可以被表示出，即对每一个非负整数 x ，都存在 n 个非负整数 $t[i]$ 满足 $a[i] \times t[i]$ 的和为 x 。然而，在网友的国度中，**货币系统可能是不完善的**，即可能存在金额 x 不能被该货币系统表示出。例如在货币系统 $n=3, a=[2, 5, 9]$ 中，金额 $1, 3$ 就无法被表示出来。

两个货币系统 (n, a) 和 (m, b) 是等价的，当且仅当**对于任意非负整数 x ，它要么均可以被两个货币系统表出，要么不能被其中任何一个表出。**

现在网友们打算简化一下货币系统。他们希望找到一个货币系统 (m, b) ，满足

(m, b) 与原来的货币系统 (n, a) 等价，且 m 尽可能的小。他们希望你来协助完成这个艰巨的任务：找到最小的 m 。

【输入格式】

输入文件名为 money.in。

输入文件的第一行包含一个整数 T ，表示数据的组数。接下来按照如下格式分别给出 T 组数据。

每组数据的第一行包含一个正整数 n 。接下来一行包含 n 个由空格隔开的正整数

$a[i]$ 。

【输出格式】

输出文件名为 money.out。

输出文件共有 T 行，对于每组数据，输出一行一个正整数，表示所有与 (n, a) 等价的货币系统 (m, b) 中，最小的 m 。

【输入输出样例 1】

money.in	money.out
2	2
4	5
3 19 10 6	
5	
11 29 13 19 17	

【输入输出样例 1 说明】

在第一组数据中，货币系统 $(2, [3, 10])$ 和给出的货币系统 (n, a) 等价，并可以验证不存在 $m < 2$ 的等价的货币系统，因此答案为 2。

在第二组数据中，可以验证不存在 $m < n$ 的等价的货币系统，因此答案为 5。

【数据规模与约定】

测试点	n	ai	测试点	n	ai
1	= 2	≤ 1000	11	≤ 13	≤ 16
2			12		
3			13		
4	= 3		14	≤ 25	≤ 40
5			15		
6			16		
7	= 4		17	≤ 100	≤ 25000
8			18		
9	= 5		19		
10			20		

对于 100% 的数据，满足 $1 \leq T \leq 20, n, a[i] \geq 1$ 。

4. 旅行

(travel.cpp/c/pas)

【问题描述】

小 Y 是一个爱好旅行的 OIer。她来到 X 国，打算将各个城市都玩一遍。

小 Y 了解到，X 国的 n 个城市之间有 m 条双向道路。每条双向道路连接两个城市。不存在两条连接同一对城市的道路，也不存在一条连接一个城市和他本身的道路。并且，从任意一个城市出发，通过这些道路都可以到达任意一个其他城市。小 Y 只能通过这道路从一个城市前往另一个城市。

小 Y 的旅行方案是这样的：任意选定一个城市作为起点，然后从起点开始，每次可以选择一条与当前城市相连的道路，走向一个**没有去过**的城市，或者沿着**第一次**访问该城市时经过的道路后退到上一个城市。当小 Y 回到起点时，她可以选择结束这次旅行或继续旅行。需要注意的是，小 Y 要求在旅行方案中，每个城市都被访问到。

为了让自己的旅行更有意义，小 Y 决定在每到达一个新的城市(包括起点)时，将它的编号记录下来。她知道这样会形成一个长度为 n 的序列。她希望这个序列的字典序最小，你能帮她吗？

对于两个长度均为 n 的序列 A 和 B，当且仅当存在一个正整数 x ，满足以下条件时，我们说序列 A 的字典序小于 B。

- 对于任意正整数 $1 \leq i < x$ ，序列 A 的第 i 个元素 A_i 和序列 B 的第 i 个元素 B_i 相同。
- 序列 A 的第 x 个元素的值小于序列 B 的第 x 个元素的值。

【输入格式】

输入文件名为 `travel.in`。

输入文件共 $m + 1$ 行。 第一行包含两个整数 $n, m (m \leq n)$ ，中间用一个空格分隔。

接下来 m 行，每行包含两个整数 $u, v (1 \leq u, v \leq n)$ ，表示编号为 u 和 v 的城市之间有一条道路，两个整数之间用一个空格分隔。

【输出格式】

输出文件名为 `travel.out`。

输出文件包含一行， n 个整数，表示字典序最小的序列。相邻两个整数之间用一个空格分隔。

【输入输出样例 1】		【输入输出样例 2】	
Travel.in	Travel.out	Travel.in	Travel.out
6 5 1 3 2 3 2 5 3 4 4 6	1 3 2 5 4 6	6 6 1 3 2 3 2 5 3 4 4 5 4 6	1 3 2 4 5 6

【数据规模与约定】

对于 100% 的数据和所有样例， $1 \leq n \leq 5000$ 且 $m = n - 1$ 或 $m = n$ 。对于不同的测试点，我们约定数据的规模如下：

测试点编号	$n =$	$m =$	特殊性质
1, 2, 3	10	$n - 1$	无
4, 5	100		无
6, 7, 8	1000		每个城市最多与两个城市相连
9, 10	1000		无
11, 12, 13	5000		每个城市最多与三个城市相连
14, 15	5000		无
16, 17	10	n	无
18, 19	100		无
20, 21, 22	1000		每个城市最多与两个城市相连
23, 24, 25	5000		无