**信息学测试题**

复赛试题

（请选手务必仔细阅读本页内容）

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 中文题目名称 | 数字游戏 | 幸运素数 | 谢幕排演 | 群山总数 | 规划食谱 | 户外运动 |
| 题目类型 | 传统型 | 传统型 | 传统型 | 传统型 | 传统型 | 传统型 |
| 目录 | acng | lucky | arrange | zone | recipe | outdoor |
| 可执行文件名 | acng | lucky | arrange | zone | recipe | outdoor |
| 输入文件名 | acng.in | lucky.in | arrange.in | zone.in | recipe.in | outdoor.in |
| 输出文件名 | acng.out | lucky.out | arrange.out | zone.out | recipe.out | outdoor.out |
| 每个测试点时限 | 1.0秒 | 1.0秒 | 1.0秒 | 1.0秒 | 1.0秒 | 1.0秒 |
| 内存限制 | 128M | 128M | 128M | 128M | 128M | 128M |
| 测试点数目 | 10 | 5 | 10 | 10 | 10 | 10 |
| 测试点是否等分 | 是 | 是 | 是 | 是 | 是 | 是 |

**提交源程序文件名**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 对于C++语言 | acng.cpp | lucky.cpp | arrange.cpp | zone.cpp | recipe.cpp | outdoor.cpp |
| 对于C语言 | acng.c | lucky.c | arrange.c | zone.c | recipe.c | outdoor.c |
| 对于pascal语言 | acng.pas | lucky.pas | arrange.pas | zone.pas | recipe.pas | outdoor.pas |

**编译选项**

|  |  |
| --- | --- |
| 对于C++语言 | -O2 -lm |
| 对于C语言 | -O2 -lm |
| 对于pascal语言 | -O2 |

**注意事项（请仔细阅读）**

1. 文件名（程序名和输入输出文件名）必须使用小写。
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考各省的具体要求。
4. 因违反以上三点而出现的错误或问题，申述时一律不予受理。
5. 若无特殊说明，结果的比较方式为全文比较(过滤行末空格及文末回车).
6. 程序可使用的栈空间内存限制与题目的内存限制一致。
7. 全国统一评测时采用的机器间置为:Inter(R) Core(TM) i7-8700K CPU @3.70GHz, 内存32GB。上述时限以此配置为准。

**1.数字游戏（acng）**

【问题描述】

班级里最近流行玩一种数字游戏，宁宁想让你帮她预测一下结果。游戏开始时，宁宁将得到一个正整数N。此时她的分数为0。

同学们按照以下规则对 N 进行变换：

如果 N 是奇数，那么将它乘以 3 后再加 1。

如果 N 是偶数，那么将它除以 2。

数字每变换一次，宁宁就得到1分。当N=1时，游戏结束。此时的分数就是她的最终得分。

【输入】

第1行，一行，一个整数 N。

【输出】

一行，一个整数，为宁宁的最终得分。

【输入输出样例1】

|  |  |
| --- | --- |
| acng.in | acng.out |
| 5 | 5 |

【数据范围】

对于 100% 的数据，1≤N≤106。

【样例说明】

当 N 的初始值为 5 时，游戏的过程如下：

| N | 变换后的数字 | 变换过程 | 总分 |
| --- | --- | --- | --- |
| 5 | 16 | 3×5+1 | 1 |
| 16 | 8 | 16÷2 | 2 |
| 8 | 4 | 8÷2 | 3 |
| 4 | 2 | 4÷2 | 4 |
| 2 | 1 | 2÷2 | 5 |

宁宁 的最终得分为 5。

**2.幸运素数（lucky）**

【问题描述】

素数，又称质数，是指除1和其自身之外，没有其他约数的正整数。例如 2,3,5,13都是素数，而 4,9,12,18则不是。特别地，规定1不是素数（因此自然数的质因数分解就是唯一的）。

如果一个数本身是素数，并且把最低位删除后得到的数仍是素数、再把最低位删除后得到的数仍是素数……如此往复，直到得到一个一位素数，我们就称它是“幸运素数”。以 233 为例：

233本身是素数；

23是素数；

2是素数。

因此233是“幸运”素数。而211则不是幸运素数：虽然211是素数，但21不是素数。请编程求出一定范围内的所有幸运素数。

【输入】

输入一行两个用空格分隔的正整数m,n，表示我们希望求幸运素数的范围。

【输出】

从小到大输出 m,m+1,m+2,⋯,n 中所有的幸运素数，每行输出一个。

【输入输出样例1】

|  |  |
| --- | --- |
| lucky.in | lucky.out |
| 6 30 | 7  23  29 |

【说明/提示】

对于 100% 的测试数据，有1≤m≤n≤9999。

**3.谢幕排演（arrange）**

【问题描述】

学校元旦晚会的最后有一项谢幕活动，参加晚会表演的共有 n 名学生，现在要从中选出 k 人排成一列拍照。

请按字典序输出所有可能的排列方式。

【输入】

仅一行，两个正整数n,k。

【输出】

若干行，每行 k 个正整数，表示一种可能的队伍顺序。

【样例】

|  |  |
| --- | --- |
| arrange.in | arrange.out |
| 3 2 | 1 2  1 3  2 1  2 3  3 1  3 2 |

【说明/提示】

对于 100% 的数据，1≤k≤n≤10。

**4.群山总数（zone）**

【问题描述】

宁宁在玩一个沙盘模拟游戏，在一个N×M的地图上，有平地和山两种地势（‘-’表示平地，‘#’表示山或山的一部分）。宁宁想要知道如果海水倒灌，将平地淹没，地图上有几座山？请你用设计一个程序帮助宁宁计数山的个数。

所有的山都是由若干个‘#’组成的图形，上下或左右相连即为同一座山的一部分。

求出该沙盘上山的总数。

【输入】

第一行为两个整数 N 和 M，用空格隔开，分别表示沙盘的行数和列数（1≤N,M≤100）。

接下来 N行，每行 M 个字符，为‘-’或‘#’。‘-’表示平地，‘#’表示山或山的一部分。

【输出】

输出山的总数。

【样例】

|  |  |
| --- | --- |
| zone.in | zone.out |
| 6 8  -----#-#  ##-----#  ##-----#  -------#  #------#  #--#---# | 5 |

。

**5.规划食谱（recipe）**

【问题描述】

青青为了保持身材想要规划一份食谱。

有这样n种食物，第i份食物含有的能量为w[i]。青青可以选择其中一些食物，获得这些食物的能量之和。当能量之和处于目标区间[L,R]，即为一份合适的食谱。显然，可能有很多种选择食物的方式可以达成这个目标，因此青青想知道方案总数。

【输入】

第一行，三个整数 n，L，R。

第二行，n 个整数，表示每份食物含有的能量w[i]。

【输出】

输出只有一行，一个整数，表示方案数。

【样例】

|  |  |
| --- | --- |
| recipe.in | recipe.out |
| 4 70 85  10 10 20 50 | 4 |

【样例解释】  
所有方案如下：

选择食物 1, 2, 4，能量 10+10+50 = 70  
选择食物 1, 3, 4，能量 10+20+50 = 80  
选择食物 2, 3, 4，能量 10+20+50 = 80  
选择食物 3, 4，能量 50+20 = 70

共 4 种方案。

【数据规模与约定】

对于 50% 的数据，满足 n≤20。

对于 100%的数据，满足 n≤40,20≤w[i]≤100,L≤R≤300。

w[i] 在范围内随机生成。

**6.户外运动（outdoor）**

【问题描述】

城市里有N(1≤N≤1000) 个适合户外运动的地点，有K(1≤K≤100) 个成员分散在这N个地点。由于他们参加了同一项户外活动，现在要集中到同一个地点去。地点之间有M(1≤M≤10000) 条有向路连接，而且不存在起点和终点相同的有向路。活动地点必须是所有人都能到达的地方。那么，有多少这样的地点可以举办活动？

注：有向路即只有一个方向的路，如果③→④表示存在一条有向路，那么从③可以到④，从④无法到达③。

【输入】

第 1 行：三个空格分隔的整数，分别是：K、N 和 M

第 2..K+1 行：第 i+1 行包含一个整数 （1..N），这是成员 i 所在的户外地点编号。

行 K+2..M+K+1：每行包含两个空格分隔的整数，分别是 A 和 B（均为 1..N 和 A ！= B），表示从地点 A 到地点 B 的单向路径。 。

【输出】

一行，单个整数，即所有成员可通过单向路径到达的户外地点数量。

【样例】

|  |  |
| --- | --- |
| outdoor.in | outdoor.out |
| 2 4 4  2  3  1 2  1 4  2 3  3 4 | 2 |

【样例说明】

成员可以在户外地点 3 或 4 中聚集。