

质因数分解

(prime.cpp/c/pas)

【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】

输入文件名为 prime.in。

输入只有一行，包含一个正整数 n 。

【输出】

输出文件名为 prime.out。

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】

prime.in	prime.out
21	7

【数据范围】

对于 60% 的数据, $6 \leq n \leq 1000$ 。对于 100% 的数据, $6 \leq n \leq 2 \times 10^9$ 。

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】

treasure.in	treasure.out
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：

1. 号房间，有楼梯通往上层，指示牌上的数字是 2 ；

2. 号房间, 无楼梯通往上层, 指示牌上的数字是 3;

3. 号房间, 有楼梯通往上层, 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 $0 < N \leq 10000$, $0 < M \leq 100$, $0 < x \leq 1,000,000$ 。

数字反转

(reverse.cpp/c/pas)

【问题描述】

给定一个整数，请将该数各个位上数字反转得到一个新数。新数也应满足整数的常见形式，即除非给定的原数为零，否则反转后得到的新数的最高位数字不应为零（参见样例 2）。

【输入】

输入文件名为 reverse.in。输入共 1 行，一个整数 N。

【输出】

输出文件名为 reverse.out。
输出共 1 行，一个整数，表示反转后的新数。

【输入输出样例 1】

reverse.in	reverse.out
123	321

【输入输出样例 2】

Reverse.in	reverse.out
-380	-83

【数据范围】

$-1,000,000,000 \leq N \leq 1,000,000,000$ 。

统计单词数

(stat.cpp/c/pas)

【问题描述】

一般的文本编辑器都有查找单词的功能，该功能可以快速定位特定单词在文章中的位置，有的还能统计出特定单词在文章中出现的次数。

现在，请你编程实现这一功能，具体要求是：给定一个单词，请你输出它在给定的文章中出现的次数和第一次出现的位置。**注意：匹配单词时，不区分大小写，但要求完全匹配，即给定单词必须与文章中的某一独立单词在不区分大小写的情况下完全相同（参见样例 1），如果给定单词仅是文章中的某一单词的一部分则不算匹配（参见样例 2）。**

【输入】

输入文件名为 stat.in，2 行。

第 1 行为一个字符串，其中只含字母，表示给定单词；

第 2 行为一个字符串，其中只可能包含字母和空格，表示给定的文章。

【输出】

输出文件名为 stat.out。

只有一行，如果在文章中找到给定单词则输出两个整数，两个整数之间用一个空格隔开，分别是单词在文章中出现的次数和第一次出现的位置（即在文章中第一次出现时，单词首字母在文章中的位置，位置从 0 开始）；如果单词在文章中没有出现，则直接输出一个整数-1

【输入输出样例 1】

stat.in	stat.out
To to be or not to be is a question	2 0

【输入输出样例 1 说明】

输出结果表示给定的单词 To 在文章中出现两次，第一次出现的位置为 0。

【输入输出样例 2】

stat.in	stat.out
to Did the Ottoman Empire lose its power at that time	-1

【输入输出样例 2 说明】

表示给定的单词 to 在文章中没有出现，输出整数-1。

【数据范围】

1 ≤ 单词长度 ≤ 10。

1 ≤ 文章长度 ≤ 1,000,000。

多项式输出
(poly.pas/c/cpp)

一元 n 次多项式可用如下的表达式表示：

$$f(x) = a_n x^n + a_{n-1} x^{n-1} + \dots + a_1 x + a_0, \quad a_n \neq 0$$

其中， $a_i x^i$ 称为 i 次项， a_i 称为 i 次项的系数。给出一个一元多项式各项的次数和系数，请按照如下规定的格式要求输出该多项式：

- 1 多项式中自变量为 x ，从左到右按照次数递减顺序给出多项式。
- 2 多项式中只包含系数不为 0 的项。
- 3 如果多项式 n 次项系数为正，则多项式开头不出现“+”号，如果多项式 n 次项系数为负，则多项式以“-”号开头。
- 4 对于不是最高次的项，以“+”号或者“-”号连接此项与前一项，分别表示此项系数为正或者系数为负。紧跟一个正整数，表示此项系数的绝对值（如果一个高于 0 次的项，其系数的绝对值为 1，则无需输出 1）如果 x 的指数大于 1，则接下来紧跟的指数部分的形式为“ x^b ”，其中 b 为 x 的指数；如果 x 的指数为 1，则接下来紧跟的指数部分形式为“ x ”；如果 x 的指数为 0，则仅需输出系数即可。
- 5 多项式中，多项式的开头、结尾不含多余的空格。

【输入】

输入文件名为 `poly.in`，共有 2 行

第一行 1 个整数， n ，表示一元多项式的次数。

第二行有 $n+1$ 个整数，其中第 i 个整数表示第 $n-i+1$ 次项的系数，每两个整数之间用空格隔开。

【输出】

输出文件 `poly.out` 共 1 行，按题目所述格式输出多项式。

【输入输出样例 1】

<code>poly.in</code>	<code>poly.out</code>
5 100 -1 1 -3 0 10	$100x^5 - x^4 + x^3 - 3x^2 + 10$

【输入输出样例 2】

<code>poly.in</code>	<code>poly.out</code>
3 -50 0 0 1	$-50x^3 + 1$

【数据范围】

$1 \leq n \leq 100$ ，多项式各次项系数的绝对值均不超过 100。

接水问题

(water.pas/c/cpp)

【问题描述】

学校里有一个水房，水房里一共装有 m 个龙头可供同学们打开水，每个龙头每秒钟的供水量相等，均为 1。

现在有 n 名同学准备接水，他们的初始接水顺序已经确定。将这些同学按接水顺序从 1 到 n 编号， i 号同学的接水量为 w_i 。接水开始时，1 到 m 号同学各占一个水龙头，并同时打开水龙头接水。当其中某名同学 j 完成其接水量要求 w_j 后，下一名排队等候接水的同学 k 马上接替 j 同学的位置开始接水。这个换人的过程是瞬间完成的，且没有任何水的浪费。即 j 同学第 x 秒结束时完成接水，则 k 同学第 $x+1$ 秒立刻开始接水。若当前接水人数 n' 不足 m ，则只有 n' 个龙头供水，其它 $m-n'$ 个龙头关闭。

现在给出 n 名同学的接水量，按照上述接水规则，问所有同学都接完水需要多少秒。

【输入】

输入文件名为 water.in。

第 1 行 2 个整数 n 和 m ，用一个空格隔开，分别表示接水人数和龙头个数。

第 2 行 n 个整数 w_1, w_2, \dots, w_n ，每两个整数之间用一个空格隔开， w_i 表示 i 号同学的接水量。

【输出】

输出文件名为 water.out。

输出只有一行，1 个整数，表示接水所需的总时间。

【输入输出样例 1】

water.in	water.out
5 3 4 4 1 2 1	4

【输入输出样例 1 说明】

第1 秒，3 人接水。第 1 秒结束时，1、2、3 号同学每人的已接水量为 1，3 号同学接完水，4 号同学接替 3 号同学开始接水。

第2 秒，3 人接水。第 2 秒结束时，1、2 号同学每人的已接水量为 2，4 号同学的已接水量为 1。

第3 秒，3 人接水。第 3 秒结束时，1、2 号同学每人的已接水量为 3，4 号同学的已接水量为 2。4 号同学接完水，5 号同学接替 4 号同学开始接水。

第4 秒，3 人接水。第 4 秒结束时，1、2 号同学每人的已接水量为 4，5 号同学的已接水量为 1。1、2、5 号同学接完水，即所有人完成接水。

总接水时间为 4 秒。

【输入输出样例 2】

water.in	water.out
8 4 23 71 87 32 70 93 80 76	163

【数据范围】

$1 \leq n \leq 10000$, $1 \leq m \leq 100$ 且 $m \leq n$;

$1 \leq w_i \leq 100$ 。

数字统计

(two.pas/c/cpp)

【问题描述】

请统计某个给定范围[L, R]的所有整数中，数字 2 出现的次数。

比如给定范围[2, 22]，数字 2 在数 2 中出现了 1 次，在数 12 中出现 1 次，在数 20 中出现 1 次，在数 21 中出现 1 次，在数 22 中出现 2 次，所以数字 2 在该范围内一共出现了 6 次。

【输入】

输入文件名为 two.in。

输入共 1 行，为两个正整数 L 和 R，之间用一个空格隔开。

【输出】

输出文件名为 two.out。

输出共 1 行，表示数字 2 出现的次数。

【输入输出样例 1】

two.in	two.out
2 22	6

【输入输出样例 2】

two.in	two.out
2 100	20

【数据范围】

$1 \leq L \leq R \leq 10000$ 。

分数线划定

(score.pas/c/cpp)

【问题描述】

世博会志愿者的选拔工作正在 A 市如火如荼的进行。为了选拔最合适的人才，A 市对所有报名的选手进行了笔试，笔试分数达到面试分数线的选手方可进入面试。面试分数线根据计划录取人数的150%划定，即如果计划录取 m 名志愿者，则**面试分数线为排名第 $m*150\%$ （向下取整）名的选手的分数**，而最终进入面试的选手为笔试成绩不低于面试分数线的所有选手。

现在就请你编写程序划定面试分数线，并输出所有进入面试的选手的报名号和笔试成绩。

【输入】

输入文件名为 score.in。

第一行，两个整数 n, m ($5 \leq n \leq 5000, 3 \leq m \leq n$)，中间用一个空格隔开，其中 n 表示报名参加笔试的选手总数， m 表示计划录取的志愿者人数。输入数据保证 $m*150\%$ 向下取整后小于等于 n 。

第二行到第 $n+1$ 行，每行包括两个整数，中间用一个空格隔开，分别是选手的报名号 k ($1000 \leq k \leq 9999$) 和该选手的笔试成绩 s ($1 \leq s \leq 100$)。数据保证选手的报名号各不相同。

【输出】

输出文件 score.out。

第一行，有两个整数，用一个空格隔开，第一个整数表示面试分数线；第二个整数为进入面试的选手的实际人数。

从第二行开始，每行包含两个整数，中间用一个空格隔开，分别表示进入面试的选手的报名号和笔试成绩，按照笔试成绩从高到低输出，如果成绩相同，则按报名号由小到大的顺序输出。

【输入输出样例】

score.in	score.out
6 3	88 5
1000 90	1005 95
3239 88	2390 95
2390 95	1000 90
7231 84	1001 88
1005 95	3239 88
1001 88	

【样例说明】

$m*150\% = 3*150\% = 4.5$ ，向下取整后为 4。保证 4 个人进入面试的分数线为 88，但因为 88 有重分，所以所有成绩大于等于 88 的选手都可以进入面试，故最终有 5 个人进入面试。

