目录

[1001. A+B和C (15) 6](#_Toc499400533)

[题目描述 6](#_Toc499400534)

[输入描述: 6](#_Toc499400535)

[输出描述: 6](#_Toc499400536)

[输入例子: 6](#_Toc499400537)

[输出例子: 6](#_Toc499400538)

[1002.数字分类 (20) 7](#_Toc499400539)

[题目描述 7](#_Toc499400540)

[输入描述: 7](#_Toc499400541)

[输出描述: 8](#_Toc499400542)

[输入例子: 8](#_Toc499400543)

[输出例子: 8](#_Toc499400544)

[1003.数素数 (20) 8](#_Toc499400545)

[题目描述 8](#_Toc499400546)

[输入描述: 8](#_Toc499400547)

[输出描述: 9](#_Toc499400548)

[输入例子: 9](#_Toc499400549)

[输出例子: 9](#_Toc499400550)

[1004.福尔摩斯的约会 (20) 9](#_Toc499400551)

[题目描述 9](#_Toc499400552)

[输入描述: 10](#_Toc499400553)

[输出描述: 10](#_Toc499400554)

[输入例子: 10](#_Toc499400555)

[输出例子: 10](#_Toc499400556)

[1005.德才论 (25) 11](#_Toc499400557)

[题目描述 11](#_Toc499400558)

[输入描述: 11](#_Toc499400559)

[输出描述: 12](#_Toc499400560)

[输入例子: 12](#_Toc499400561)

[输出例子: 13](#_Toc499400562)

[1006.部分A+B (15) 13](#_Toc499400563)

[题目描述 14](#_Toc499400564)

[输入描述: 14](#_Toc499400565)

[输出描述: 14](#_Toc499400566)

[输入例子: 14](#_Toc499400567)

[输出例子: 14](#_Toc499400568)

[1007.A除以B (20) 14](#_Toc499400569)

[题目描述 15](#_Toc499400570)

[输入描述: 15](#_Toc499400571)

[输出描述: 15](#_Toc499400572)

[输入例子: 15](#_Toc499400573)

[输出例子: 15](#_Toc499400574)

[1008.锤子剪刀布 (20) 15](#_Toc499400575)

[题目描述 16](#_Toc499400576)

[输入描述: 16](#_Toc499400577)

[输出描述: 16](#_Toc499400578)

[输入例子: 16](#_Toc499400579)

[输出例子: 17](#_Toc499400580)

[1009.数字黑洞 (20) 17](#_Toc499400581)

[题目描述 17](#_Toc499400582)

[输入描述: 18](#_Toc499400583)

[输出描述: 18](#_Toc499400584)

[输入例子: 18](#_Toc499400585)

[输出例子: 18](#_Toc499400586)

[1010.月饼 (25) 18](#_Toc499400587)

[题目描述 19](#_Toc499400588)

[输入描述: 19](#_Toc499400589)

[输出描述: 19](#_Toc499400590)

[输入例子: 19](#_Toc499400591)

[输出例子: 20](#_Toc499400592)

[1011.个位数统计 (15) 20](#_Toc499400593)

[题目描述 20](#_Toc499400594)

[输入描述: 20](#_Toc499400595)

[输出描述: 20](#_Toc499400596)

[输入例子: 21](#_Toc499400597)

[输出例子: 21](#_Toc499400598)

[1012.D进制的A+B (20) 21](#_Toc499400599)

[题目描述 21](#_Toc499400600)

[输入描述: 21](#_Toc499400601)

[输出描述: 21](#_Toc499400602)

[输入例子: 22](#_Toc499400603)

[输出例子: 22](#_Toc499400604)

[1013.组个最小数 (20) 22](#_Toc499400605)

[题目描述 22](#_Toc499400606)

[输入描述: 22](#_Toc499400607)

[输出描述: 23](#_Toc499400608)

[输入例子: 23](#_Toc499400609)

[输出例子: 23](#_Toc499400610)

[1014.科学计数法 (20) 23](#_Toc499400611)

[题目描述 23](#_Toc499400612)

[输入描述: 24](#_Toc499400613)

[输出描述: 24](#_Toc499400614)

[输入例子: 24](#_Toc499400615)

[输出例子: 24](#_Toc499400616)

[1015.反转链表 (25) 24](#_Toc499400617)

[题目描述 24](#_Toc499400618)

[输入描述: 25](#_Toc499400619)

[输出描述: 25](#_Toc499400620)

[输入例子: 25](#_Toc499400621)

[输出例子: 26](#_Toc499400622)

[1016.程序运行时间(15) 26](#_Toc499400623)

[题目描述 26](#_Toc499400624)

[输入描述: 27](#_Toc499400625)

[输出描述: 27](#_Toc499400626)

[输入例子: 27](#_Toc499400627)

[输出例子: 27](#_Toc499400628)

[1017.打印沙漏(20) 27](#_Toc499400629)

[题目描述 28](#_Toc499400630)

[输入描述: 28](#_Toc499400631)

[输出描述: 28](#_Toc499400632)

[输入例子: 28](#_Toc499400633)

[输出例子: 28](#_Toc499400634)

[1018.人口普查(20) 29](#_Toc499400635)

[题目描述 29](#_Toc499400636)

[输入描述: 29](#_Toc499400637)

[输出描述: 30](#_Toc499400638)

[输入例子: 30](#_Toc499400639)

[输出例子: 30](#_Toc499400640)

[1019.旧键盘 (20) 30](#_Toc499400641)

[题目描述 30](#_Toc499400642)

[输入描述: 31](#_Toc499400643)

[输出描述: 31](#_Toc499400644)

[输入例子: 31](#_Toc499400645)

[输出例子: 31](#_Toc499400646)

[1020.完美数列(25) 31](#_Toc499400647)

[题目描述 32](#_Toc499400648)

[输入描述: 32](#_Toc499400649)

[输出描述: 32](#_Toc499400650)

[输入例子: 32](#_Toc499400651)

[输出例子: 32](#_Toc499400652)

[1021. 查验身份证(15) 32](#_Toc499400653)

[题目描述 33](#_Toc499400654)

[输入描述: 33](#_Toc499400655)

[输出描述: 34](#_Toc499400656)

[输入例子: 34](#_Toc499400657)

[输出例子: 34](#_Toc499400658)

[1022.挖掘机技术哪家强(20) 34](#_Toc499400659)

[题目描述 35](#_Toc499400660)

[输入描述: 35](#_Toc499400661)

[输出描述: 35](#_Toc499400662)

[输入例子: 35](#_Toc499400663)

[输出例子: 36](#_Toc499400664)

[1023.旧键盘打字(20) 36](#_Toc499400665)

[题目描述 36](#_Toc499400666)

[输入描述: 36](#_Toc499400667)

[输出描述: 37](#_Toc499400668)

[输入例子: 37](#_Toc499400669)

[输出例子: 37](#_Toc499400670)

[1024.有理数四则运算(20) 37](#_Toc499400671)

[题目描述 37](#_Toc499400672)

[输入描述: 37](#_Toc499400673)

[输出描述: 38](#_Toc499400674)

[输入例子: 38](#_Toc499400675)

[输出例子: 38](#_Toc499400676)

[1025.插入与归并(25) 38](#_Toc499400677)

[题目描述 39](#_Toc499400678)

[输入描述: 39](#_Toc499400679)

[输出描述: 39](#_Toc499400680)

[输入例子: 40](#_Toc499400681)

[输出例子: 40](#_Toc499400682)

[1026.跟奥巴马一起编程(15) 40](#_Toc499400683)

[题目描述 40](#_Toc499400684)

[输入描述: 40](#_Toc499400685)

[输出描述: 41](#_Toc499400686)

[输入例子: 41](#_Toc499400687)

[输出例子: 41](#_Toc499400688)

[1027.在霍格沃茨找零钱（20） 41](#_Toc499400689)

[题目描述 41](#_Toc499400690)

[输入描述: 42](#_Toc499400691)

[输出描述: 42](#_Toc499400692)

[输入例子: 42](#_Toc499400693)

[输出例子: 42](#_Toc499400694)

[1028.统计同成绩学生(20) 42](#_Toc499400695)

[题目描述 43](#_Toc499400696)

[输入描述: 43](#_Toc499400697)

[输出描述: 43](#_Toc499400698)

[输入例子: 43](#_Toc499400699)

[输出例子: 43](#_Toc499400700)

[1029.到底买不买（20） 43](#_Toc499400701)

[题目描述 44](#_Toc499400702)

[输入描述: 44](#_Toc499400703)

[输出描述: 44](#_Toc499400704)

[输入例子: 44](#_Toc499400705)

[输出例子: 45](#_Toc499400706)

[1030.有几个PAT（25） 45](#_Toc499400707)

[题目描述 45](#_Toc499400708)

[输入描述: 45](#_Toc499400709)

[输出描述: 45](#_Toc499400710)

[输入例子: 45](#_Toc499400711)

[输出例子: 46](#_Toc499400712)

# 1001. A+B和C (15)

### 题目描述

给定区间[-2的31次方, 2的31次方]内的3个整数A、B和C，请判断A+B是否大于C。

### 输入描述:

输入第1行给出正整数T(<=10)，是测试用例的个数。随后给出T组测试用例，每组占一行，顺序给出A、B和C。整数间以空格分隔。

### 输出描述:

对每组测试用例，在一行中输出“Case #X: true”如果A+B>C，否则输出“Case #X: false”，其中X是测试用例的编号（从1开始）。

### 输入例子:

4  
  
1 2 3  
  
2 3 4  
  
2147483647 0 2147483646  
  
0 -2147483648 -2147483647

### 输出例子:

Case #1: false  
  
Case #2: true  
  
Case #3: true  
  
Case #4: false

# 1002.数字分类 (20)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

给定一系列正整数，请按要求对数字进行分类，并输出以下5个数字：  
  
  
  
A1 = 能被5整除的数字中所有偶数的和；  
  
A2 = 将被5除后余1的数字按给出顺序进行交错求和，即计算n1-n2+n3-n4...；  
  
A3 = 被5除后余2的数字的个数；  
  
A4 = 被5除后余3的数字的平均数，精确到小数点后1位；  
  
A5 = 被5除后余4的数字中最大数字。

### 输入描述:

每个输入包含1个测试用例。每个测试用例先给出一个不超过1000的正整数N，随后给出N个不超过1000的待分类的正整数。数字间以空格分隔。

### 输出描述:

对给定的N个正整数，按题目要求计算A1~A5并在一行中顺序输出。数字间以空格分隔，但行末不得有多余空格。  
  
若其中某一类数字不存在，则在相应位置输出“N”。

### 输入例子:

13 1 2 3 4 5 6 7 8 9 10 20 16 18

### 输出例子:

30 11 2 9.7 9

# 1003.数素数 (20)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

令Pi表示第i个素数。现任给两个正整数M <= N <= 10000，请输出PM到PN的所有素数。

### 输入描述:

输入在一行中给出M和N，其间以空格分隔。

### 输出描述:

输出从PM到PN的所有素数，每10个数字占1行，其间以空格分隔，但行末不得有多余空格。

### 输入例子:

5 27

### 输出例子:

11 13 17 19 23 29 31 37 41 43  
  
47 53 59 61 67 71 73 79 83 89  
  
97 101 103

# 1004.福尔摩斯的约会 (20)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

大侦探福尔摩斯接到一张奇怪的字条：“我们约会吧！ 3485djDkxh4hhGE 2984akDfkkkkggEdsb s&hgsfdk d&Hyscvnm”。大侦探很

快就明白了，字条上奇怪的乱码实际上就是约会的时间“星期四 14:04”，因为前面两字符串中第1对相同的大写英文字母（大小写有区分）是

第4个字母'D'，代表星期四；第2对相同的字符是'E'，那是第5个英文字母，代表一天里的第14个钟头（于是一天的0点到23点由数字0到9、

以及大写字母A到N表示）；后面两字符串第1对相同的英文字母's'出现在第4个位置（从0开始计数）上，代表第4分钟。现给定两对字符串，

请帮助福尔摩斯解码得到约会的时间。

### 输入描述:

输入在4行中分别给出4个非空、不包含空格、且长度不超过60的字符串。

### 输出描述:

在一行中输出约会的时间，格式为“DAY HH:MM”，其中“DAY”是某星期的3字符缩写，即MON表示星期一，TUE表示星期二，WED表示星期三，THU表示星期  
  
四，FRI表示星期五，SAT表示星期六，SUN表示星期日。题目输入保证每个测试存在唯一解。

### 输入例子:

3485djDkxh4hhGE  
  
2984akDfkkkkggEdsb  
  
s&hgsfdk  
  
d&Hyscvnm

### 输出例子:

THU 14:04

# 1005.德才论 (25)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

宋代史学家司马光在《资治通鉴》中有一段著名的“德才论”：“是故才德全尽谓之圣人，才德兼亡谓之愚人，德胜才谓之君子，才胜德谓之  
  
小人。凡取人之术，苟不得圣人，君子而与之，与其得小人，不若得愚人。”  
  
  
  
现给出一批考生的德才分数，请根据司马光的理论给出录取排名。

### 输入描述:

输入第1行给出3个正整数，分别为：N（<=105），即考生总数；L（>=60），为录取最低分数线，即德分和才分均不低于L的考生才有资格  
  
被考虑录取；H（<100），为优先录取线——德分和才分均不低于此线的被定义为“才德全尽”，此类考生按德才总分从高到低排序；才分不到  
  
但德分到线的一类考生属于“德胜才”，也按总分排序，但排在第一类考生之后；德才分均低于H，但是德分不低于才分的考生属于“才德兼  
  
亡”但尚有“德胜才”者，按总分排序，但排在第二类考生之后；其他达到最低线L的考生也按总分排序，但排在第三类考生之后。  
  
  
随后N行，每行给出一位考生的信息，包括：准考证号、德分、才分，其中准考证号为8位整数，德才分为区间[0, 100]内的整数。数字间以空格分隔。

### 输出描述:

输出第1行首先给出达到最低分数线的考生人数M，随后M行，每行按照输入格式输出一位考生的信息，考生按输入中说明的规则从高到低排序。当某类考生中有多人  
  
总分相同时，按其德分降序排列；若德分也并列，则按准考证号的升序输出。

### 输入例子:

14 60 80  
  
10000001 64 90  
  
10000002 90 60  
  
10000011 85 80  
  
10000003 85 80  
  
10000004 80 85  
  
10000005 82 77  
  
10000006 83 76  
  
10000007 90 78  
  
10000008 75 79  
  
10000009 59 90  
  
10000010 88 45  
  
10000012 80 100  
  
10000013 90 99  
  
10000014 66 60

### 输出例子:

12  
  
10000013 90 99  
  
10000012 80 100  
  
10000003 85 80  
  
10000011 85 80  
  
10000004 80 85  
  
10000007 90 78  
  
10000006 83 76  
  
10000005 82 77  
  
10000002 90 60  
  
10000014 66 60  
  
10000008 75 79  
  
10000001 64 90

# 1006.部分A+B (15)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

正整数A的“DA（为1位整数）部分”定义为由A中所有DA组成的新整数PA。例如：给定A = 3862767，DA = 6，则A的“6部分”PA是66，因为A中有2个6。  
   
 现给定A、DA、B、DB，请编写程序计算PA + PB。

### 输入描述:

输入在一行中依次给出A、DA、B、DB，中间以空格分隔，其中0 < A, B < 1010。

### 输出描述:

在一行中输出PA + PB的值。

### 输入例子:

3862767 6 13530293 3

### 输出例子:

399

# 1007.A除以B (20)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

本题要求计算A/B，其中A是不超过1000位的正整数，B是1位正整数。你需要输出商数Q和余数R，使得A = B \* Q + R成立。

### 输入描述:

输入在1行中依次给出A和B，中间以1空格分隔。

### 输出描述:

在1行中依次输出Q和R，中间以1空格分隔。

### 输入例子:

123456789050987654321 7

### 输出例子:

17636684150141093474 3

# 1008.锤子剪刀布 (20)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

大家应该都会玩“锤子剪刀布”的游戏：  
  
现给出两人的交锋记录，请统计双方的胜、平、负次数，并且给出双方分别出什么手势的胜算最大。

### 输入描述:

输入第1行给出正整数N（<=105），即双方交锋的次数。随后N行，每行给出一次交锋的信息，即甲、乙双方同时给出的的手势。C代表“锤子”、J代表“剪刀”、B代  
  
表“布”，第1个字母代表甲方，第2个代表乙方，中间有1个空格。

### 输出描述:

输出第1、2行分别给出甲、乙的胜、平、负次数，数字间以1个空格分隔。第3行给出两个字母，分别代表甲、乙获胜次数最多的手势，中间有1个空格。如果解不唯  
  
一，则输出按字母序最小的解。

### 输入例子:

10  
  
C J  
  
J B  
  
C B  
  
B B  
  
B C  
  
C C  
  
C B  
  
J B  
  
B C  
  
J J

### 输出例子:

5 3 2  
  
2 3 5  
  
B B

# 1009.数字黑洞 (20)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

给定任一个各位数字不完全相同的4位正整数，如果我们先把4个数字按非递增排序，再按非递减排序，然后用第1个数字减第2个数字，将得到  
 一个新的数字。一直重复这样做，我们很快会停在有“数字黑洞”之称的6174，这个神奇的数字也叫Kaprekar常数。  
   
 例如，我们从6767开始，将得到  
   
 7766 - 6677 = 1089  
 9810 - 0189 = 9621  
 9621 - 1269 = 8352  
 8532 - 2358 = 6174  
 7641 - 1467 = 6174  
 ... ...  
   
 现给定任意4位正整数，请编写程序演示到达黑洞的过程。

### 输入描述:

输入给出一个(0, 10000)区间内的正整数N。

### 输出描述:

如果N的4位数字全相等，则在一行内输出“N - N = 0000”；否则将计算的每一步在一行内输出，直到6174作为差出现，输出格式见样例,每行中间没有空行。注意每个数字按4位数格  
 式输出。

### 输入例子:

6767

### 输出例子:

7766 - 6677 = 1089  
 9810 - 0189 = 9621  
 9621 - 1269 = 8352  
 8532 - 2358 = 6174

# 1010.月饼 (25)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

月饼是中国人在中秋佳节时吃的一种传统食品，不同地区有许多不同风味的月饼。现给定所有种类月饼的库存量、总售价、以及市场的最大需  
  
求量，请你计算可以获得的最大收益是多少。  
  
  
  
注意：销售时允许取出一部分库存。样例给出的情形是这样的：假如我们有3种月饼，其库存量分别为18、15、10万吨，总售价分别为75、  
  
72、45亿元。如果市场的最大需求量只有20万吨，那么我们最大收益策略应该是卖出全部15万吨第2种月饼、以及5万吨第3种月饼，获得  
  
 72 + 45/2 = 94.5（亿元）。

### 输入描述:

每个输入包含1个测试用例。每个测试用例先给出一个不超过1000的正整数N表示月饼的种类数、以及不超过500（以万吨为单位）的正整数  
  
D表示市场最大需求量。随后一行给出N个正数表示每种月饼的库存量（以万吨为单位）；最后一行给出N个正数表示每种月饼的总售价（以亿  
  
元为单位）。数字间以空格分隔。

### 输出描述:

对每组测试用例，在一行中输出最大收益，以亿元为单位并精确到小数点后2位。

### 输入例子:

3 20  
  
18 15 10  
  
75 72 45

### 输出例子:

94.50

# 1011.个位数统计 (15)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

给定一个k位整数N = dk-1\*10k-1 + ... + d1\*101 + d0 (0<=di<=9, i=0,...,k-1, dk-1>0)，请编写程序统计每种不同的个位数字出现的次数。例如：给定N = 100311，则有2个0，3个1，和1个3。

### 输入描述:

每个输入包含1个测试用例，即一个不超过1000位的正整数N。

### 输出描述:

对N中每一种不同的个位数字，以D:M的格式在一行中输出该位数字D及其在N中出现的次数M。要求按D的升序输出。

### 输入例子:

100311

### 输出例子:

0:2  
  
1:3  
  
3:1

# 1012.D进制的A+B (20)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

输入两个非负10进制整数A和B(<=230-1)，输出A+B的D (1 < D <= 10)进制数。

### 输入描述:

输入在一行中依次给出3个整数A、B和D。

### 输出描述:

输出A+B的D进制数。

### 输入例子:

123 456 8

### 输出例子:

1103

# 1013.组个最小数 (20)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

给定数字0-9各若干个。你可以以任意顺序排列这些数字，但必须全部使用。目标是使得最后得到的数尽可能小（注意0不能做首位）。例如：  
  
给定两个0，两个1，三个5，一个8，我们得到的最小的数就是10015558。  
  
  
  
现给定数字，请编写程序输出能够组成的最小的数。

### 输入描述:

每个输入包含1个测试用例。每个测试用例在一行中给出10个非负整数，顺序表示我们拥有数字0、数字1、……数字9的个数。整数间用一个空  
  
格分隔。10个数字的总个数不超过50，且至少拥有1个非0的数字。

### 输出描述:

在一行中输出能够组成的最小的数。

### 输入例子:

2 2 0 0 0 3 0 0 1 0

### 输出例子:

10015558

# 1014.科学计数法 (20)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

科学计数法是科学家用来表示很大或很小的数字的一种方便的方法，其满足正则表达式[+-][1-9]"."[0-9]+E[+-][0-9]+，即数字的整数部分  
  
只有1位，小数部分至少有1位，该数字及其指数部分的正负号即使对正数也必定明确给出。  
  
  
  
现以科学计数法的格式给出实数A，请编写程序按普通数字表示法输出A，并保证所有有效位都被保留。

### 输入描述:

每个输入包含1个测试用例，即一个以科学计数法表示的实数A。该数字的存储长度不超过9999字节，且其指数的绝对值不超过9999。

### 输出描述:

对每个测试用例，在一行中按普通数字表示法输出A，并保证所有有效位都被保留，包括末尾的0。

### 输入例子:

+1.23400E-03

### 输出例子:

0.00123400

# 1015.反转链表 (25)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

给定一个常数K以及一个单链表L，请编写程序将L中每K个结点反转。例如：给定L为1→2→3→4→5→6，K为3，则输出应该为  
  
3→2→1→6→5→4；如果K为4，则输出应该为4→3→2→1→5→6，即最后不到K个元素不反转。

### 输入描述:

每个输入包含1个测试用例。每个测试用例第1行给出第1个结点的地址、结点总个数正整数N(<= 105)、以及正整数K(<=N)，即要求反转的  
  
子链结点的个数。结点的地址是5位非负整数，NULL地址用-1表示。  
  
  
  
接下来有N行，每行格式为：  
  
  
  
Address Data Next  
  
  
  
其中Address是结点地址，Data是该结点保存的整数数据，Next是下一结点的地址。

### 输出描述:

对每个测试用例，顺序输出反转后的链表，其上每个结点占一行，格式与输入相同。

### 输入例子:

00100 6 4  
  
00000 4 99999  
  
00100 1 12309  
  
68237 6 -1  
  
33218 3 00000  
  
99999 5 68237  
  
12309 2 33218

### 输出例子:

00000 4 33218  
  
33218 3 12309  
  
12309 2 00100  
  
00100 1 99999  
  
99999 5 68237  
  
68237 6 -1

# 1016.程序运行时间(15)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

要获得一个C语言程序的运行时间，常用的方法是调用头文件time.h，其中提供了clock()函数，可以捕捉从程序开始运行到clock()被调用时所  
  
耗费的时间。这个时间单位是clock tick，即“时钟打点”。同时还有一个常数CLK\_TCK，给出了机器时钟每秒所走的时钟打点数。于是为了获  
  
得一个函数f的运行时间，我们只要在调用f之前先调用clock()，获得一个时钟打点数C1；在f执行完成后再调用clock()，获得另一个时钟打点  
  
数C2；两次获得的时钟打点数之差(C2-C1)就是f运行所消耗的时钟打点数，再除以常数CLK\_TCK，就得到了以秒为单位的运行时间。  
  
  
  
这里不妨简单假设常数CLK\_TCK为100。现给定被测函数前后两次获得的时钟打点数，请你给出被测函数运行的时间。

### 输入描述:

输入在一行中顺序给出2个整数C1和C1。注意两次获得的时钟打点数肯定不相同，即C1 < C2，并且取值在[0, 107]

### 输出描述:

在一行中输出被测函数运行的时间。运行时间必须按照“hh:mm:ss”（即2位的“时:分:秒”）格式输出；不足1秒的时间四舍五入到秒。

### 输入例子:

123 4577973

### 输出例子:

12:42:59

# 1017.打印沙漏(20)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

本题要求你写个程序把给定的符号打印成沙漏的形状。例如给定17个“\*”，要求按下列格式打印  
   
 \*\*\*\*\*  
  \*\*\*  
   \*  
  \*\*\*  
 \*\*\*\*\*  
 所谓“沙漏形状”，是指每行输出奇数个符号；各行符号中心对齐；相邻两行符号数差2；符号数先从大到小顺序递减到1，再从小到大顺序递  
 增；首尾符号数相等。  
   
 给定任意N个符号，不一定能正好组成一个沙漏。要求打印出的沙漏能用掉尽可能多的符号。

### 输入描述:

输入在一行给出1个正整数N（<=1000）和一个符号，中间以空格分隔。

### 输出描述:

首先打印出由给定符号组成的最大的沙漏形状，最后在一行中输出剩下没用掉的符号数。

### 输入例子:

19 \*

### 输出例子:

\*\*\*\*\*  
  
 \*\*\*  
  
 \*  
  
 \*\*\*  
  
\*\*\*\*\*  
  
2

# 1018.人口普查(20)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

某城镇进行人口普查，得到了全体居民的生日。现请你写个程序，找出镇上最年长和最年轻的人。  
  
  
  
这里确保每个输入的日期都是合法的，但不一定是合理的——假设已知镇上没有超过200岁的老人，而今天是2014年9月6日，所以超过200  
  
岁的生日和未出生的生日都是不合理的，应该被过滤掉。

### 输入描述:

输入在第一行给出正整数N，取值在(0, 105]；随后N行，每行给出1个人的姓名（由不超过5个英文字母组成的字符串）、以及  
  
按“yyyy/mm/dd”（即年/月/日）格式给出的生日。题目保证最年长和最年轻的人没有并列。

### 输出描述:

在一行中顺序输出有效生日的个数、最年长人和最年轻人的姓名，其间以空格分隔。

### 输入例子:

5  
  
John 2001/05/12  
  
Tom 1814/09/06  
  
Ann 2121/01/30  
  
James 1814/09/05  
  
Steve 1967/11/20

### 输出例子:

3 Tom John

# 1019.旧键盘 (20)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

旧键盘上坏了几个键，于是在敲一段文字的时候，对应的字符就不会出现。现在给出应该输入的一段文字、以及实际被输入的文字，请你列出  
  
肯定坏掉的那些键。

### 输入描述:

输入在2行中分别给出应该输入的文字、以及实际被输入的文字。每段文字是不超过80个字符的串，由字母A-Z（包括大、小写）、数字0-9、  
  
以及下划线“\_”（代表空格）组成。题目保证2个字符串均非空。

### 输出描述:

按照发现顺序，在一行中输出坏掉的键。其中英文字母只输出大写，每个坏键只输出一次。题目保证至少有1个坏键。

### 输入例子:

7\_This\_is\_a\_test  
  
\_hs\_s\_a\_es

### 输出例子:

7TI

# 1020.完美数列(25)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

给定一个正整数数列，和正整数p，设这个数列中的最大值是M，最小值是m，如果M <= m \* p，则称这个数列是完美数列。  
  
  
  
现在给定参数p和一些正整数，请你从中选择尽可能多的数构成一个完美数列。

### 输入描述:

输入第一行给出两个正整数N和p，其中N（<= 105）是输入的正整数的个数，p（<= 109）是给定的参数。第二行给出N个正整数，每个数  
  
不超过109。

### 输出描述:

在一行中输出最多可以选择多少个数可以用它们组成一个完美数列。

### 输入例子:

10 8  
  
2 3 20 4 5 1 6 7 8 9

### 输出例子:

8

# 1021. 查验身份证(15)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

一个合法的身份证号码由17位地区、日期编号和顺序编号加1位校验码组成。校验码的计算规则如下：

首先对前17位数字加权求和，权重分配为：{7，9，10，5，8，4，2，1，6，3，7，9，10，5，8，4，2}；然后将计算的和对11取模得

到值Z；最后按照以下关系对应Z值与校验码M的值：

Z：0 1 2 3 4 5 6 7 8 9 10

M：1 0 X 9 8 7 6 5 4 3 2

现在给定一些身份证号码，请你验证校验码的有效性，并输出有问题的号码。

### 输入描述:

输入第一行给出正整数N（<= 100）是输入的身份证号码的个数。随后N行，每行给出1个18位身份证号码。

### 输出描述:

按照输入的顺序每行输出1个有问题的身份证号码。这里并不检验前17位是否合理，只检查前17位是否全为数字且最后1位校验码计算准确。如果所有号码都正常，

则输出“All passed”。

### 输入例子:

4

320124198808240056

12010X198901011234

110108196711301866

37070419881216001X

### 输出例子:

12010X198901011234

110108196711301866

37070419881216001X

# 1022.挖掘机技术哪家强(20)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

为了用事实说明挖掘机技术到底哪家强，PAT组织了一场挖掘机技能大赛。现请你根据比赛结果统计出技术最强的那个学校。

### 输入描述:

输入在第1行给出不超过105的正整数N，即参赛人数。随后N行，每行给出一位参赛者的信息和成绩，包括其所代表的学校的编号（从1开始  
  
连续编号）、及其比赛成绩（百分制），中间以空格分隔。

### 输出描述:

在一行中给出总得分最高的学校的编号、及其总分，中间以空格分隔。题目保证答案唯一，没有并列。

### 输入例子:

6  
  
3 65  
  
2 80  
  
1 100  
  
2 70  
  
3 40  
  
3 0

### 输出例子:

2 150

# 1023.旧键盘打字(20)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

旧键盘上坏了几个键，于是在敲一段文字的时候，对应的字符就不会出现。现在给出应该输入的一段文字、以及坏掉的那些键，打出的结果文

字会是怎样？

### 输入描述:

输入在2行中分别给出坏掉的那些键、以及应该输入的文字。其中对应英文字母的坏键以大写给出；每段文字是不超过10^5个字符的串。可用的

字符包括字母[a-z, A-Z]、数字0-9、以及下划线“\_”（代表空格）、“,”、“.”、“-”、“+”（代表上档键）。题目保证第2行输入的文字串非空。

注意：如果上档键坏掉了，那么大写的英文字母无法被打出。

### 输出描述:

在一行中输出能够被打出的结果文字。如果没有一个字符能被打出，则输出空行。

### 输入例子:

7+IE.

7\_This\_is\_a\_test.

### 输出例子:

\_hs\_s\_a\_tst

# 1024.有理数四则运算(20)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

本题要求编写程序，计算2个有理数的和、差、积、商。

### 输入描述:

输入在一行中按照“a1/b1 a2/b2”的格式给出两个分数形式的有理数，其中分子和分母全是整型范围内的整数，负号只可能出现在分子前，分  
  
母不为0。

### 输出描述:

分别在4行中按照“有理数1 运算符 有理数2 = 结果”的格式顺序输出2个有理数的和、差、积、商。注意输出的每个有理数必须是该有理数的  
  
最简形式“k a/b”，其中k是整数部分，a/b是最简分数部分；若为负数，则须加括号；若除法分母为0，则输出“Inf”。题目保证正确的输出中  
  
没有超过整型范围的整数。

### 输入例子:

5/3 0/6

### 输出例子:

1 2/3 + 0 = 1 2/3  
  
1 2/3 - 0 = 1 2/3  
  
1 2/3 \* 0 = 0  
  
1 2/3 / 0 = Inf

# 1025.插入与归并(25)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

根据维基百科的定义：  
  
  
  
插入排序是迭代算法，逐一获得输入数据，逐步产生有序的输出序列。每步迭代中，算法从输入序列中取出一元素，将之插入有序序列中正确  
  
的位置。如此迭代直到全部元素有序。  
  
  
  
归并排序进行如下迭代操作：首先将原始序列看成N个只包含1个元素的有序子序列，然后每次迭代归并两个相邻的有序子序列，直到最后只剩  
  
下1个有序的序列。  
  
  
  
现给定原始序列和由某排序算法产生的中间序列，请你判断该算法究竟是哪种排序算法？

### 输入描述:

输入在第一行给出正整数N (<=100)；随后一行给出原始序列的N个整数；最后一行给出由某排序算法产生的中间序列。这里假设排序的目标序列是升序。数字间以  
  
空格分隔。

### 输出描述:

首先在第1行中输出“Insertion Sort”表示插入排序、或“Merge Sort”表示归并排序；然后在第2行中输出用该排序算法再迭代一轮的结果序列。题目保证每组测试  
  
的结果是唯一的。数字间以空格分隔，且行末不得有多余空格。

### 输入例子:

10  
  
3 1 2 8 7 5 9 4 6 0  
  
1 2 3 7 8 5 9 4 6 0

### 输出例子:

Insertion Sort  
  
1 2 3 5 7 8 9 4 6 0

# 1026.跟奥巴马一起编程(15)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

美国总统奥巴马不仅呼吁所有人都学习编程，甚至以身作则编写代码，成为美国历史上首位编写计算机代码的总统。2014年底，为庆祝“计算  
  
机科学教育周”正式启动，奥巴马编写了很简单的计算机代码：在屏幕上画一个正方形。现在你也跟他一起画吧！

### 输入描述:

输入在一行中给出正方形边长N（3<=N<=20）和组成正方形边的某种字符C，间隔一个空格。

### 输出描述:

输出由给定字符C画出的正方形。但是注意到行间距比列间距大，所以为了让结果看上去更像正方形，我们输出的行数实际上是列数的50%  
  
（四舍五入取整）。

### 输入例子:

10 a

### 输出例子:

aaaaaaaaaa

a a

a a

a a

aaaaaaaaaa

# 1027.在霍格沃茨找零钱（20）

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

如果你是哈利·波特迷，你会知道魔法世界有它自己的货币系统 —— 就如海格告诉哈利的：“十七个银西可(Sickle)兑一个加隆(Galleon)，二  
  
十九个纳特(Knut)兑一个西可，很容易。”现在，给定哈利应付的价钱P和他实付的钱A，你的任务是写一个程序来计算他应该被找的零钱。

### 输入描述:

输入在1行中分别给出P和A，格式为“Galleon.Sickle.Knut”，其间用1个空格分隔。这里Galleon是[0, 107]]区间内的整数，Sickle是[0,   
  
17)区间内的整数，Knut是[0, 29)区间内的整数。

### 输出描述:

在一行中用与输入同样的格式输出哈利应该被找的零钱。如果他没带够钱，那么输出的应该是负数。

### 输入例子:

10.16.27 14.1.28

### 输出例子:

3.2.1

# 1028.统计同成绩学生(20)

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

本题要求读入N名学生的成绩，将获得某一给定分数的学生人数输出。

### 输入描述:

输入在第1行给出不超过105的正整数N，即学生总人数。随后1行给出N名学生的百分制整数成绩，中间以空格分隔。最后1行给出要查询的分  
  
数个数K（不超过N的正整数），随后是K个分数，中间以空格分隔。

### 输出描述:

在一行中按查询顺序给出得分等于指定分数的学生人数，中间以空格分隔，但行末不得有多余空格。

### 输入例子:

10  
  
60 75 90 55 75 99 82 90 75 50  
  
3 75 90 88

### 输出例子:

3 2 0

# 1029.到底买不买（20）

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

小红想买些珠子做一串自己喜欢的珠串。卖珠子的摊主有很多串五颜六色的珠串，但是不肯把任何一串拆散了卖。于是小红要你帮忙判断一  
  
下，某串珠子里是否包含了全部自己想要的珠子？如果是，那么告诉她有多少多余的珠子；如果不是，那么告诉她缺了多少珠子。  
  
  
  
为方便起见，我们用[0-9]、[a-z]、[A-Z]范围内的字符来表示颜色。例如，YrR8RrY是小红想做的珠串；那么ppRYYGrrYBR2258可以买，因为包含了  
  
全部她想要的珠子，还多了8颗不需要的珠子；ppRYYGrrYB225不能买，因为没有黑色珠子，并且少了一颗红色的珠子。

### 输入描述:

每个输入包含1个测试用例。每个测试用例分别在2行中先后给出摊主的珠串和小红想做的珠串，两串都不超过1000个珠子。

### 输出描述:

如果可以买，则在一行中输出“Yes”以及有多少多余的珠子；如果不可以买，则在一行中输出“No”以及缺了多少珠子。其间以1个空格分隔。

### 输入例子:

ppRYYGrrYBR2258  
  
YrR8RrY

### 输出例子:

Yes 8

# 1030.有几个PAT（25）

时间限制 1000 ms 内存限制 32768 KB 代码长度限制 100 KB 判断程序 Standard (来自 [小小](https://www.nowcoder.com/profile/59))

### 题目描述

字符串APPAPT中包含了两个单词“PAT”，其中第一个PAT是第2位(P),第4位(A),第6位(T)；第二个PAT是第3位(P),第4位(A),第6位(T)。  
  
  
  
现给定字符串，问一共可以形成多少个PAT？

### 输入描述:

输入只有一行，包含一个字符串，长度不超过105，只包含P、A、T三种字母。

### 输出描述:

在一行中输出给定字符串中包含多少个PAT。由于结果可能比较大，只输出对1000000007取余数的结果。

### 输入例子:

APPAPT

### 输出例子:

2