目录

1001 害死人不偿命的(3n+1)猜想 (15)（15 分） 3

1002 写出这个数 (20)（20 分） 3

1003 我要通过！(20)（20 分） 4

1004 成绩排名 (20)（20 分） 4

1005 继续(3n+1)猜想 (25)（25 分） 5

1006 换个格式输出整数 (15)（15 分） 5

1007 素数对猜想 (20)（20 分） 6

1008 数组元素循环右移问题 (20)（20 分） 6

1009 说反话 (20)（20 分） 7

1010 一元多项式求导 (25)（25 分） 7

1011 A+B和C (15)（15 分） 8

1012 数字分类 (20)（20 分）kj 8

1013 数素数 (20)（20 分） 9

1014 福尔摩斯的约会 (20)（20 分） 10

1015 德才论 (25)（25 分） 10

1016 部分A+B (15)（15 分） 11

1017 A除以B (20)（20 分） 12

1018 锤子剪刀布 (20)（20 分） 12

1019 数字黑洞 (20)（20 分） 13

1020 月饼 (25)（25 分） 14

1021 个位数统计 (15)（15 分） 15

1022 D进制的A+B (20)（20 分） 16

1023 组个最小数 (20)（20 分） 16

1024 科学计数法 (20)（20 分） 17

1025 反转链表 (25)（25 分） 17

1026 程序运行时间(15)（15 分） 18

1027 打印沙漏(20)（20 分） 19

1028 人口普查(20)（20 分） 19

1029 旧键盘(20)（20 分） 20

1030 完美数列(25)（25 分） 21

1031 查验身份证(15)（15 分） 21

1032 挖掘机技术哪家强(20)（20 分） 22

1033 旧键盘打字(20)（20 分） 23

1034 有理数四则运算(20)（20 分） 23

1035 插入与归并(25)（25 分） 24

1036 跟奥巴马一起编程(15)（15 分） 25

1037 在霍格沃茨找零钱（20）（20 分） 25

1038 统计同成绩学生(20)（20 分） 26

1039 到底买不买（20）（20 分） 27

1040 有几个PAT（25）（25 分） 27

1041 考试座位号(15)（15 分） 28

1042 字符统计(20)（20 分） 29

1043 输出PATest(20)（20 分） 29

1044 火星数字(20)（20 分） 30

1045 快速排序(25)（25 分） 30

1046 划拳(15)（15 分） 31

1047 编程团体赛(20)（20 分） 32

1048 数字加密(20)（20 分） 32

1049 数列的片段和(20)（20 分） 33

1050 螺旋矩阵(25)（25 分） 33

1051 复数乘法 (15)（15 分） 34

1052 卖个萌 (20)（20 分） 34

1053 住房空置率 (20)（20 分） 35

1054 求平均值 (20)（20 分） 36

1055 集体照 (25)（25 分） 37

1056 组合数的和（15 分） 38 38

1057 数零壹（20 分） 38 39

1058 选择题（20 分） 39 40

1059 C语言竞赛（20 分） 40 41

1060 爱丁顿数（25 分） 41 42

1061 判断题（15 分） 42 42

1062 最简分数（20 分） 43 43

1063 计算谱半径（20 分） 43 44

1064 朋友数（20 分） 44 45

1065 单身狗（25 分） 45 45

1066 图像过滤（15 分） 46 46

1067 试密码（20 分） 46 47

1068 万绿丛中一点红（20 分） 47 49

1069 微博转发抽奖（20 分） 49 50

1070 结绳（25 分） 50

1071 小赌怡情（15 分） 51 52

1072 开学寄语（20 分） 52

1073 多选题常见计分法（20 分） 54 55

1074 宇宙无敌加法器（20 分） 55 56

1075 链表元素分类（25 分） 56 57

1076 Wifi密码（15 分） 57

1077 互评成绩计算（20 分） 58 59

1078 字符串压缩与解压（20 分） 59 60

1079 延迟的回文数（20 分） 60 61

1080 MOOC期终成绩（25 分） 61 62

1081 检查密码（15 分） 62 63

1082 射击比赛（20 分） 63 64

1083 是否存在相等的差（20 分） 64 64

1084 外观数列（20 分） 64 65

1085 PAT单位排行（25 分） 65 66

1001 害死人不偿命的(3n+1)猜想 (15)（15 分）

卡拉兹(Callatz)猜想：

对任何一个自然数n，如果它是偶数，那么把它砍掉一半；如果它是奇数，那么把(3n+1)砍掉一半。这样一直反复砍下去，最后一定在某一步得到n=1。卡拉兹在1950年的世界数学家大会上公布了这个猜想，传说当时耶鲁大学师生齐动员，拼命想证明这个貌似很傻很天真的命题，结果闹得学生们无心学业，一心只证(3n+1)，以至于有人说这是一个阴谋，卡拉兹是在蓄意延缓美国数学界教学与科研的进展……

我们今天的题目不是证明卡拉兹猜想，而是对给定的任一不超过1000的正整数n，简单地数一下，需要多少步（砍几下）才能得到n=1？

**输入格式：**每个测试输入包含1个测试用例，即给出自然数n的值。

**输出格式：**输出从n计算到1需要的步数。

**输入样例：**

3

**输出样例：**

5

1002 写出这个数 (20)（20 分）

读入一个自然数n，计算其各位数字之和，用汉语拼音写出和的每一位数字。

**输入格式：**每个测试输入包含1个测试用例，即给出自然数n的值。这里保证n小于10^100^。

**输出格式：**在一行内输出n的各位数字之和的每一位，拼音数字间有1 空格，但一行中最后一个拼音数字后没有空格。

**输入样例：**

1234567890987654321123456789

**输出样例：**

yi san wu

1003 我要通过！(20)（20 分）

“**答案正确**”是自动判题系统给出的最令人欢喜的回复。本题属于PAT的“**答案正确**”大派送 —— 只要读入的字符串满足下列条件，系统就输出“**答案正确**”，否则输出“**答案错误**”。

得到“**答案正确**”的条件是：

1. 字符串中必须仅有P, A, T这三种字符，不可以包含其它字符；\

1. 任意形如 xPATx 的字符串都可以获得“**答案正确**”，其中 x 或者是空字符串，或者是仅由字母 A 组成的字符串；\
2. 如果 aPbTc 是正确的，那么 aPbATca 也是正确的，其中 a, b, c 均或者是空字符串，或者是仅由字母 A 组成的字符串。

现在就请你为PAT写一个自动裁判程序，判定哪些字符串是可以获得“**答案正确**”的。

**输入格式：** 每个测试输入包含1个测试用例。第1行给出一个自然数n (&lt10)，是需要检测的字符串个数。接下来每个字符串占一行，字符串长度不超过100，且不包含空格。

**输出格式：**每个字符串的检测结果占一行，如果该字符串可以获得“**答案正确**”，则输出YES，否则输出NO。

**输入样例：**

8 PAT PAAT AAPATAA AAPAATAAAA xPATx PT Whatever APAAATAA

**输出样例：**

YES YES YES YES NO NO NO NO

1004 成绩排名 (20)（20 分）

读入n名学生的姓名、学号、成绩，分别输出成绩最高和成绩最低学生的姓名和学号。

**输入格式：**每个测试输入包含1个测试用例，格式为\

第1行：正整数n 第2行：第1个学生的姓名 学号 成绩 第3行：第2个学生的姓名 学号 成绩 ... ... ... 第n+1行：第n个学生的姓名 学号 成绩

其中姓名和学号均为不超过10个字符的字符串，成绩为0到100之间的一个整数，这里保证在一组测试用例中没有两个学生的成绩是相同的。

**输出格式：**对每个测试用例输出2行，第1行是成绩最高学生的姓名和学号，第2行是成绩最低学生的姓名和学号，字符串间有1空格。

**输入样例：**

3 Joe Math990112 89 Mike CS991301 100 Mary EE990830 95

**输出样例：**

Mike CS991301 Joe Math990112

1005 继续(3n+1)猜想 (25)（25 分）

卡拉兹(Callatz)猜想已经在1001中给出了描述。在这个题目里，情况稍微有些复杂。

当我们验证卡拉兹猜想的时候，为了避免重复计算，可以记录下递推过程中遇到的每一个数。例如对n=3进行验证的时候，我们需要计算3、5、8、4、2、1，则当我们对n=5、8、4、2进行验证的时候，就可以直接判定卡拉兹猜想的真伪，而不需要重复计算，因为这4个数已经在验证3的时候遇到过了，我们称5、8、4、2是被3“覆盖”的数。我们称一个数列中的某个数n为“关键数”，如果n不能被数列中的其他数字所覆盖。

现在给定一系列待验证的数字，我们只需要验证其中的几个关键数，就可以不必再重复验证余下的数字。你的任务就是找出这些关键数字，并按从大到小的顺序输出它们。

**输入格式：**每个测试输入包含1个测试用例，第1行给出一个正整数K(&lt100)，第2行给出K个互不相同的待验证的正整数n(1&ltn<=100)的值，数字间用空格隔开。

**输出格式：**每个测试用例的输出占一行，按从大到小的顺序输出关键数字。数字间用1个空格隔开，但一行中最后一个数字后没有空格。

**输入样例：**

6 3 5 6 7 8 11

**输出样例：**

7 6

1006 换个格式输出整数 (15)（15 分）

让我们用字母B来表示“百”、字母S表示“十”，用“12...n”来表示个位数字n（&lt10），换个格式来输出任一个不超过3位的正整数。例如234应该被输出为BBSSS1234，因为它有2个“百”、3个“十”、以及个位的4。

**输入格式：**每个测试输入包含1个测试用例，给出正整数n（&lt1000）。

**输出格式：**每个测试用例的输出占一行，用规定的格式输出n。

**输入样例1：**

234

**输出样例1：**

BBSSS1234

**输入样例2：**

23

**输出样例2：**

SS12

1007 素数对猜想 (20)（20 分）

让我们定义 d~n~ 为：d~n~ = p~n+1~ - p~n~，其中 p~i~ 是第i个素数。显然有 d~1~=1 且对于n&gt1有 d~n~ 是偶数。“素数对猜想”认为“存在无穷多对相邻且差为2的素数”。

现给定任意正整数N (< 10^5^)，请计算不超过N的满足猜想的素数对的个数。

**输入格式：**每个测试输入包含1个测试用例，给出正整数N。

**输出格式：**每个测试用例的输出占一行，不超过N的满足猜想的素数对的个数。

**输入样例：**

20

**输出样例：**

4

1008 数组元素循环右移问题 (20)（20 分）

一个数组A中存有N（N&gt0）个整数，在不允许使用另外数组的前提下，将每个整数循环向右移M（M>=0）个位置，即将A中的数据由（A~0~ A~1~……A~N-1~）变换为（A~N-M~ …… A~N-1~ A~0~ A~1~……A~N-M-1~）（最后M个数循环移至最前面的M个位置）。如果需要考虑程序移动数据的次数尽量少，要如何设计移动的方法？

**输入格式：**每个输入包含一个测试用例，第1行输入N ( 1<=N<=100)、M（M>=0）；第2行输入N个整数，之间用空格分隔。

**输出格式：**在一行中输出循环右移M位以后的整数序列，之间用空格分隔，序列结尾不能有多余空格。

**输入样例：**

6 2 1 2 3 4 5 6

**输出样例：**

5 6 1 2 3 4

1009 说反话 (20)（20 分）

给定一句英语，要求你编写程序，将句中所有单词的顺序颠倒输出。

**输入格式：**测试输入包含一个测试用例，在一行内给出总长度不超过80的字符串。字符串由若干单词和若干空格组成，其中单词是由英文字母（大小写有区分）组成的字符串，单词之间用1个空格分开，输入保证句子末尾没有多余的空格。

**输出格式：**每个测试用例的输出占一行，输出倒序后的句子。

**输入样例：**

Hello World Here I Come

**输出样例：**

Come I Here World Hello

1010 一元多项式求导 (25)（25 分）

设计函数求一元多项式的导数。（注：x^n^（n为整数）的一阶导数为n\*x^n-1^。）

**输入格式：**以指数递降方式输入多项式非零项系数和指数（绝对值均为不超过1000的整数）。数字间以空格分隔。

**输出格式：**以与输入相同的格式输出导数多项式非零项的系数和指数。数字间以空格分隔，但结尾不能有多余空格。注意“零多项式”的指数和系数都是0，但是表示为“0 0”。

**输入样例：**

3 4 -5 2 6 1 -2 0

**输出样例：**

12 3 -10 1 6 0

1011 A+B和C (15)（15 分）

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

**输入格式：**

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。

**输出格式：**

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）。

**输入样例：**

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

**输出样例：**

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

1012 数字分类 (20)（20 分）kj

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。

**输入格式：**

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。

**输出格式：**

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”。

**输入样例1：**

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

**输出样例1：**

30 11 2 9.7 9

**输入样例2：**

8 1 2 4 5 6 7 9 16

**输出样例2：**

N 11 2 N 9

1013 数素数 (20)（20 分）

令P~i~表示第i个素数。现任给两个正整数M <= N <= 10^4^，请输出P~M~到P~N~的所有素数。

**输入格式：**

输入在一行中给出M和N，其间以空格分隔。

**输出格式：**

输出从P~M~到P~N~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。

**输入样例：**

5 27

**输出样例：**

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

1014 福尔摩斯的约会 (20)（20 分）

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。

**输入格式：**

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。

**输出格式：**

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。

**输入样例：**

3485djDkxh4hhGE 2984akDfkkkkggEdsb s&hgsfdk d&Hyscvnm

**输出样例：**

THU 14:04

1015 德才论 (25)（25 分）

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。

**输入格式：**

输入第1行给出3个正整数，分别为：N（<=10^5^），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。

**输出格式：**

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。

**输入样例：**

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

**输出样例：**

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

1016 部分A+B (15)（15 分）

正整数A的“D~A~（为1位整数）部分”定义为由A中所有D~A~组成的新整数P~A~。例如：给定A = 3862767，D~A~ = 6，则A的“6部分”P~A~是66，因为A中有2个6。

现给定A、D~A~、B、D~B~，请编写程序计算P~A~ + P~B~。

**输入格式：**

输入在一行中依次给出A、D~A~、B、D~B~，中间以空格分隔，其中0 < A, B < 10^10^。

**输出格式：**

在一行中输出P~A~ + P~B~的值。

**输入样例1：**

3862767 6 13530293 3

**输出样例1：**

399

**输入样例2：**

3862767 1 13530293 8

**输出样例2：**

0

1017 A除以B (20)（20 分）

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。

**输入格式：**

输入在1行中依次给出A和B，中间以1空格分隔。

**输出格式：**

在1行中依次输出Q和R，中间以1空格分隔。

**输入样例：**

123456789050987654321 7

**输出样例：**

17636684150141093474 3

1018 锤子剪刀布 (20)（20 分）

大家应该都会玩“锤子剪刀布”的游戏：两人同时给出手势，胜负规则如图所示：

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。

**输入格式：**

输入第1行给出正整数N（<=10^5^），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。

**输出格式：**

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。

**输入样例：**

10

C J

J B

C B

B B

B C

C C

C B

J B

B C

J J

**输出样例：**

5 3 2

2 3 5

B B

1019 数字黑洞 (20)（20 分）

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。

**输入格式：**

输入给出一个(0, 10000)区间内的正整数N。

**输出格式：**

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。注意每个数字按4位数格式输出。

**输入样例1：**

6767

**输出样例1：**

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

**输入样例2：**

2222

**输出样例2：**

2222 - 2222 = 0000

1020 月饼 (25)（25 分）

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）。

**输入格式：**

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。

**输出格式：**

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。

**输入样例：**

3 20 18 15 10 75 72 45

**输出样例：**

94.50

1021 个位数统计 (15)（15 分）

给定一个k位整数N = d~k-1~\*10^k-1^ + ... + d~1~\*10^1^ + d~0~ (0<=d~i~<=9, i=0,...,k-1, d~k-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。

**输入格式：**

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。

**输出格式：**

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。

**输入样例：**

100311

**输出样例：**

0:2 1:3 3:1

1022 D进制的A+B (20)（20 分）

输入两个非负10进制整数A和B(<=2^30^-1)，输出A+B的D (1 < D <= 10)进制数。

**输入格式：**

输入在一行中依次给出3个整数A、B和D。

**输出格式：**

输出A+B的D进制数。

**输入样例：**

123 456 8

**输出样例：**

1103

1023 组个最小数 (20)（20 分）

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。

**输入格式：**

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。

**输出格式：**

在一行中输出能够组成的最小的数。

**输入样例：**

2 2 0 0 0 3 0 0 1 0

**输出样例：**

10015558

1024 科学计数法 (20)（20 分）

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。

**输入格式：**

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。

**输出格式：**

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。

**输入样例1：**

+1.23400E-03

**输出样例1：**

0.00123400

**输入样例2：**

-1.2E+10

**输出样例2：**

-12000000000

1025 反转链表 (25)（25 分）

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。

**输入格式：**

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(<= 10^5^)、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。

**输出格式：**

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。

**输入样例：**

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

**输出样例：**

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

1026 程序运行时间(15)（15 分）

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。

**输入格式：**

输入在一行中顺序给出2个整数C1和C2。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, 10^7^]。

**输出格式：**

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。

**输入样例：**

123 4577973

**输出样例：**

12:42:59

1027 打印沙漏(20)（20 分）

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。

**输入格式：**

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。

**输出格式：**

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。

**输入样例：**

19 \*

**输出样例：**

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

1028 人口普查(20)（20 分）

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。

**输入格式：**

输入在第一行给出正整数N，取值在(0, 10^5^]；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。

**输出格式：**

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。

**输入样例：**

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

**输出样例：**

3 Tom John

1029 旧键盘(20)（20 分）

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。

**输入格式：**

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。

**输出格式：**

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。

**输入样例：**

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

**输出样例：**

7TI

1030 完美数列(25)（25 分）

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。

**输入格式：**

输入第一行给出两个正整数N和p，其中N（<= 10^5^）是输入的正整数的个数，p（<= 10^9^）是给定的参数。第二行给出N个正整数，每个数不超过10^9^。

**输出格式：**

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。

**输入样例：**

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

**输出样例：**

8

1031 查验身份证(15)（15 分）

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。

**输入格式：**

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。

**输出格式：**

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”。

**输入样例1：**

4 320124198808240056 12010X198901011234 110108196711301866 37070419881216001X

**输出样例1：**

12010X198901011234 110108196711301866 37070419881216001X

**输入样例2：**

2 320124198808240056 110108196711301862

**输出样例2：**

All passed

1032 挖掘机技术哪家强(20)（20 分）

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。

**输入格式：**

输入在第1行给出不超过10^5^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。

**输出格式：**

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。

**输入样例：**

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

**输出样例：**

2 150

1033 旧键盘打字(20)（20 分）

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？

**输入格式：**

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。

**输出格式：**

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。

**输入样例：**

7+IE. 7\_This\_is\_a\_test.

**输出样例：**

\_hs\_s\_a\_tst

1034 有理数四则运算(20)（20 分）

本题要求编写程序，计算2个有理数的和、差、积、商。

**输入格式：**

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。

**输出格式：**

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。

**输入样例1：**

2/3 -4/2

**输出样例1：**

2/3 + (-2) = (-1 1/3) 2/3 - (-2) = 2 2/3 2/3 \* (-2) = (-1 1/3) 2/3 / (-2) = (-1/3)

**输入样例2：**

5/3 0/6

**输出样例2：**

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

1035 插入与归并(25)（25 分）

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？

**输入格式：**

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。

**输出格式：**

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。

**输入样例1：**

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

**输出样例1：**

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

**输入样例2：**

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

**输出样例2：**

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

1036 跟奥巴马一起编程(15)（15 分）

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！

**输入格式：**

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。

**输出格式：**

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）。

**输入样例：**

10 a

**输出样例：**

aaaaaaaaaa a a a a a a aaaaaaaaaa

1037 在霍格沃茨找零钱（20）（20 分）

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。

**输入格式：**

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, 10^7^]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。

**输出格式：**

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。

**输入样例1：**

10.16.27 14.1.28

**输出样例1：**

3.2.1

**输入样例2：**

14.1.28 10.16.27

**输出样例2：**

-3.2.1

1038 统计同成绩学生(20)（20 分）

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。

**输入格式：**

输入在第1行给出不超过10^5^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。

**输出格式：**

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。

**输入样例：**

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

**输出样例：**

3 2 0

1039 到底买不买（20）（20 分）

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。例如在图1中，第3串是小红想做的珠串；那么第1串可以买，因为包含了全部她想要的珠子，还多了8颗不需要的珠子；第2串不能买，因为没有黑色珠子，并且少了一颗红色的珠子。

**输入格式：**

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。

**输出格式：**

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。

**输入样例1：**

ppRYYGrrYBR2258

YrR8RrY

**输出样例1：**

Yes 8

**输入样例2：**

ppRYYGrrYB225

YrR8RrY

**输出样例2：**

##### No 2

1040 有几个PAT（25）（25 分）

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？

**输入格式：**

输入只有一行，包含一个字符串，长度不超过10^5^，只包含P、A、T三种字母。

**输出格式：**

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。

**输入样例：**

APPAPT

**输出样例：**

2

1041 考试座位号(15)（15 分）

每个PAT考生在参加考试时都会被分配两个座位号，一个是试机座位，一个是考试座位。正常情况下，考生在入场时先得到试机座位号码，入座进入试机状态后，系统会显示该考生的考试座位号码，考试时考生需要换到考试座位就座。但有些考生迟到了，试机已经结束，他们只能拿着领到的试机座位号码求助于你，从后台查出他们的考试座位号码。

**输入格式：**

输入第一行给出一个正整数N（<=1000），随后N行，每行给出一个考生的信息：“准考证号 试机座位号 考试座位号”。其中准考证号由14位数字组成，座位从1到N编号。输入保证每个人的准考证号都不同，并且任何时候都不会把两个人分配到同一个座位上。

考生信息之后，给出一个正整数M（<=N），随后一行中给出M个待查询的试机座位号码，以空格分隔。

**输出格式：**

对应每个需要查询的试机座位号码，在一行中输出对应考生的准考证号和考试座位号码，中间用1个空格分隔。

**输入样例：**

4 10120150912233 2 4 10120150912119 4 1 10120150912126 1 3 10120150912002 3 2 2 3 4

**输出样例：**

10120150912002 2 10120150912119 1

1042 字符统计(20)（20 分）

请编写程序，找出一段给定文字中出现最频繁的那个英文字母。

**输入格式：**

输入在一行中给出一个长度不超过1000的字符串。字符串由ASCII码表中任意可见字符及空格组成，至少包含1个英文字母，以回车结束（回车不算在内）。

**输出格式：**

在一行中输出出现频率最高的那个英文字母及其出现次数，其间以空格分隔。如果有并列，则输出按字母序最小的那个字母。统计时不区分大小写，输出小写字母。

**输入样例：**

This is a simple TEST. There ARE numbers and other symbols 1&2&3...........

**输出样例：**

e 7

1043 输出PATest(20)（20 分）

给定一个长度不超过10000的、仅由英文字母构成的字符串。请将字符重新调整顺序，按“PATestPATest....”这样的顺序输出，并忽略其它字符。当然，六种字符的个数不一定是一样多的，若某种字符已经输出完，则余下的字符仍按PATest的顺序打印，直到所有字符都被输出。

**输入格式：**

输入在一行中给出一个长度不超过10000的、仅由英文字母构成的非空字符串。

**输出格式：**

在一行中按题目要求输出排序后的字符串。题目保证输出非空。

**输入样例：**

redlesPayBestPATTopTeePHPereatitAPPT

**输出样例：**

PATestPATestPTetPTePePee

1044 火星数字(20)（20 分）

火星人是以13进制计数的：

* 地球人的0被火星人称为tret。
* 地球人数字1到12的火星文分别为：jan, feb, mar, apr, may, jun, jly, aug, sep, oct, nov, dec。
* 火星人将进位以后的12个高位数字分别称为：tam, hel, maa, huh, tou, kes, hei, elo, syy, lok, mer, jou。

例如地球人的数字“29”翻译成火星文就是“hel mar”；而火星文“elo nov”对应地球数字“115”。为了方便交流，请你编写程序实现地球和火星数字之间的互译。

**输入格式：**

输入第一行给出一个正整数N（<100），随后N行，每行给出一个[0, 169)区间内的数字 —— 或者是地球文，或者是火星文。

**输出格式：**

对应输入的每一行，在一行中输出翻译后的另一种语言的数字。

**输入样例：**

4 29 5 elo nov tam

**输出样例：**

hel mar may 115 13

1045 快速排序(25)（25 分）

著名的快速排序算法里有一个经典的划分过程：我们通常采用某种方法取一个元素作为主元，通过交换，把比主元小的元素放到它的左边，比主元大的元素放到它的右边。 给定划分后的N个互不相同的正整数的排列，请问有多少个元素可能是划分前选取的主元？

例如给定N = 5, 排列是1、3、2、4、5。则：

1的左边没有元素，右边的元素都比它大，所以它可能是主元；\

尽管3的左边元素都比它小，但是它右边的2它小，所以它不能是主元；\

尽管2的右边元素都比它大，但其左边的3比它大，所以它不能是主元；\

类似原因，4和5都可能是主元。

因此，有3个元素可能是主元。

**输入格式：**

输入在第1行中给出一个正整数N（<= 10^5^）； 第2行是空格分隔的N个不同的正整数，每个数不超过10^9^。

**输出格式：**

在第1行中输出有可能是主元的元素个数；在第2行中按递增顺序输出这些元素，其间以1个空格分隔，行末不得有多余空格。

**输入样例：**

5 1 3 2 4 5

**输出样例：**

3 1 4 5

1046 划拳(15)（15 分）

划拳是古老中国酒文化的一个有趣的组成部分。酒桌上两人划拳的方法为：每人口中喊出一个数字，同时用手比划出一个数字。如果谁比划出的数字正好等于两人喊出的数字之和，谁就赢了，输家罚一杯酒。两人同赢或两人同输则继续下一轮，直到唯一的赢家出现。

下面给出甲、乙两人的划拳记录，请你统计他们最后分别喝了多少杯酒。

**输入格式：**

输入第一行先给出一个正整数N（<=100），随后N行，每行给出一轮划拳的记录，格式为：

甲喊 甲划 乙喊 乙划

其中“喊”是喊出的数字，“划”是划出的数字，均为不超过100的正整数（两只手一起划）。

**输出格式：**

在一行中先后输出甲、乙两人喝酒的杯数，其间以一个空格分隔。

**输入样例：**

5 8 10 9 12 5 10 5 10 3 8 5 12 12 18 1 13 4 16 12 15

**输出样例：**

1 2

1047 编程团体赛(20)（20 分）

编程团体赛的规则为：每个参赛队由若干队员组成；所有队员独立比赛；参赛队的成绩为所有队员的成绩和；成绩最高的队获胜。

现给定所有队员的比赛成绩，请你编写程序找出冠军队。

**输入格式：**

输入第一行给出一个正整数N（<=10000），即所有参赛队员总数。随后N行，每行给出一位队员的成绩，格式为：“队伍编号-队员编号 成绩”，其中“队伍编号”为1到1000的正整数，“队员编号”为1到10的正整数，“成绩”为0到100的整数。

**输出格式：**

在一行中输出冠军队的编号和总成绩，其间以一个空格分隔。注意：题目保证冠军队是唯一的。

**输入样例：**

6 3-10 99 11-5 87 102-1 0 102-3 100 11-9 89 3-2 61

**输出样例：**

11 176

1048 数字加密(20)（20 分）

本题要求实现一种数字加密方法。首先固定一个加密用正整数A，对任一正整数B，将其每1位数字与A的对应位置上的数字进行以下运算：对奇数位，对应位的数字相加后对13取余——这里用J代表10、Q代表11、K代表12；对偶数位，用B的数字减去A的数字，若结果为负数，则再加10。这里令个位为第1位。

**输入格式：**

输入在一行中依次给出A和B，均为不超过100位的正整数，其间以空格分隔。

**输出格式：**

在一行中输出加密后的结果。

**输入样例：**

1234567 368782971

**输出样例：**

3695Q8118

1049 数列的片段和(20)（20 分）

给定一个正数数列，我们可以从中截取任意的连续的几个数，称为片段。例如，给定数列{0.1, 0.2, 0.3, 0.4}，我们有(0.1) (0.1, 0.2) (0.1, 0.2, 0.3) (0.1, 0.2, 0.3, 0.4) (0.2) (0.2, 0.3) (0.2, 0.3, 0.4) (0.3) (0.3, 0.4) (0.4) 这10个片段。

给定正整数数列，求出全部片段包含的所有的数之和。如本例中10个片段总和是0.1

* 0.3 + 0.6 + 1.0 + 0.2 + 0.5 + 0.9 + 0.3 + 0.7 + 0.4 = 5.0。

**输入格式：**

输入第一行给出一个不超过10^5^的正整数N，表示数列中数的个数，第二行给出N个不超过1.0的正数，是数列中的数，其间以空格分隔。

**输出格式：**

在一行中输出该序列所有片段包含的数之和，精确到小数点后2位。

**输入样例：**

4 0.1 0.2 0.3 0.4

**输出样例：**

5.00

1050 螺旋矩阵(25)（25 分）

本题要求将给定的N个正整数按非递增的顺序，填入“螺旋矩阵”。所谓“螺旋矩阵”，是指从左上角第1个格子开始，按顺时针螺旋方向填充。要求矩阵的规模为m行n列，满足条件：m\*n等于N；m>=n；且m-n取所有可能值中的最小值。

**输入格式：**

输入在第1行中给出一个正整数N，第2行给出N个待填充的正整数。所有数字不超过10^4^，相邻数字以空格分隔。

**输出格式：**

输出螺旋矩阵。每行n个数字，共m行。相邻数字以1个空格分隔，行末不得有多余空格。

**输入样例：**

12 37 76 20 98 76 42 53 95 60 81 58 93

**输出样例：**

98 95 93 42 37 81 53 20 76 58 60 76

1051 复数乘法 (15)（15 分）

复数可以写成(A + B**i**)的常规形式，其中A是实部，B是虚部，**i**是虚数单位，满足**i**^2^ = -1；也可以写成极坐标下的指数形式(R\*e^(P**i**)^)，其中R是复数模，P是辐角，**i**是虚数单位，其等价于三角形式 R(cos(P) + **i**sin(P))。

现给定两个复数的R和P，要求输出两数乘积的常规形式。

**输入格式：**

输入在一行中依次给出两个复数的R1, P1, R2, P2，数字间以空格分隔。

**输出格式：**

在一行中按照“A+Bi”的格式输出两数乘积的常规形式，实部和虚部均保留2位小数。注意：如果B是负数，则应该写成“A-|B|i”的形式。

**输入样例：**

2.3 3.5 5.2 0.4

**输出样例：**

-8.68-8.23i

1052 卖个萌 (20)（20 分）

萌萌哒表情符号通常由“手”、“眼”、“口”三个主要部分组成。简单起见，我们假设一个表情符号是按下列格式输出的：

[左手]([左眼][口][右眼])[右手]

现给出可选用的符号集合，请你按用户的要求输出表情。

**输入格式：**

输入首先在前三行顺序对应给出手、眼、口的可选符号集。每个符号括在一对方括号[]内。题目保证每个集合都至少有一个符号，并不超过10个符号；每个符号包含1到4个非空字符。

之后一行给出一个正整数K，为用户请求的个数。随后K行，每行给出一个用户的符号选择，顺序为左手、左眼、口、右眼、右手——这里只给出符号在相应集合中的序号（从1开始），数字间以空格分隔。

**输出格式：**

对每个用户请求，在一行中输出生成的表情。若用户选择的序号不存在，则输出“Are you kidding me? @\/@”。

**输入样例：**

[╮][╭][o][~\][/~] [<][>] [╯][╰][^][-][=][>][<][@][⊙] [Д][▽][\_][ε][^] ... 4 1 1 2 2 2 6 8 1 5 5 3 3 4 3 3 2 10 3 9 3

**输出样例：**

╮(╯▽╰)╭ <(@Д=)/~ o(^ε^)o Are you kidding me? @\/@

1053 住房空置率 (20)（20 分）

在不打扰居民的前提下，统计住房空置率的一种方法是根据每户用电量的连续变化规律进行判断。判断方法如下：

* 在观察期内，若存在超过一半的日子用电量低于某给定的阈值e，则该住房为“可能空置”；
* 若观察期超过某给定阈值D天，且满足上一个条件，则该住房为“空置”。

现给定某居民区的住户用电量数据，请你统计“可能空置”的比率和“空置”比率，即以上两种状态的住房占居民区住房总套数的百分比。

**输入格式：**

输入第一行给出正整数N（<=1000），为居民区住房总套数；正实数e，即低电量阈值；正整数D，即观察期阈值。随后N行，每行按以下格式给出一套住房的用电量数据：

K E~1~ E~2~ ... E~K~

其中K为观察的天数，E~i~为第i天的用电量。

**输出格式：**

在一行中输出“可能空置”的比率和“空置”比率的百分比值，其间以一个空格分隔，保留小数点后1位。

**输入样例：**

5 0.5 10 6 0.3 0.4 0.5 0.2 0.8 0.6 10 0.0 0.1 0.2 0.3 0.0 0.8 0.6 0.7 0.0 0.5 5 0.4 0.3 0.5 0.1 0.7 11 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 0.1 11 2 2 2 1 1 0.1 1 0.1 0.1 0.1 0.1

**输出样例：**

40.0% 20.0%

（样例解释：第2、3户为“可能空置”，第4户为“空置”，其他户不是空置。）

1054 求平均值 (20)（20 分）

本题的基本要求非常简单：给定N个实数，计算它们的平均值。但复杂的是有些输入数据可能是非法的。一个“合法”的输入是[-1000，1000]区间内的实数，并且最多精确到小数点后2位。当你计算平均值的时候，不能把那些非法的数据算在内。

**输入格式：**

输入第一行给出正整数N（<=100）。随后一行给出N个实数，数字间以一个空格分隔。

**输出格式：**

对每个非法输入，在一行中输出“ERROR: X is not a legal number”，其中X是输入。最后在一行中输出结果：“The average of K numbers is Y”，其中K是合法输入的个数，Y是它们的平均值，精确到小数点后2位。如果平均值无法计算，则用“Undefined”替换Y。如果K为1，则输出“The average of 1 number is Y”。

**输入样例1：**

7 5 -3.2 aaa 9999 2.3.4 7.123 2.35

**输出样例1：**

ERROR: aaa is not a legal number ERROR: 9999 is not a legal number ERROR: 2.3.4 is not a legal number ERROR: 7.123 is not a legal number The average of 3 numbers is 1.38

**输入样例2：**

2 aaa -9999

1055 集体照 (25)（25 分）

拍集体照时队形很重要，这里对给定的N个人K排的队形设计排队规则如下：

* 每排人数为N/K（向下取整），多出来的人全部站在最后一排；
* 后排所有人的个子都不比前排任何人矮；
* 每排中最高者站中间（中间位置为m/2+1，其中m为该排人数，除法向下取整）；
* 每排其他人以中间人为轴，按身高非增序，先右后左交替入队站在中间人的两侧（例如5人身高为190、188、186、175、170，则队形为175、188、190、186、170。这里假设你面对拍照者，所以你的左边是中间人的右边）；
* 若多人身高相同，则按名字的字典序升序排列。这里保证无重名。

现给定一组拍照人，请编写程序输出他们的队形。

**输入格式：**

每个输入包含1个测试用例。每个测试用例第1行给出两个正整数N（<=10000，总人数）和K（<=10，总排数）。随后N行，每行给出一个人的名字（不包含空格、长度不超过8个英文字母）和身高（[30, 300]区间内的整数）。

**输出格式：**

输出拍照的队形。即K排人名，其间以空格分隔，行末不得有多余空格。注意：假设你面对拍照者，后排的人输出在上方，前排输出在下方。

**输入样例：**

10 3 Tom 188 Mike 170 Eva 168 Tim 160 Joe 190 Ann 168 Bob 175 Nick 186 Amy 160 John 159

**输出样例：**

Bob Tom Joe Nick Ann Mike Eva Tim Amy John

1056 组合数的和（15 分）

给定 N 个非 0 的个位数字，用其中任意 2 个数字都可以组合成 1 个 2 位的数字。要求所有可能组合出来的 2 位数字的和。例如给定 2、5、8，则可以组合出：25、28、52、58、82、85，它们的和为330。

### 输入格式：

输入在第一行中给出 N（1 < N < 10），随后一行给出 N 个不同的非 0 个位数字。数字间以空格分隔。

### 输出格式：

输出所有可能组合出来的2位数字的和。

### 输入样例：

3 2 8 5

### 输出样例：

330

1057 数零壹（20 分）

给定一串长度不超过 10​5​​ 的字符串，本题要求你将其中所有英文字母的序号（字母 a-z 对应序号 1-26，不分大小写）相加，得到整数 N，然后再分析一下 N 的二进制表示中有多少 0、多少 1。例如给定字符串 PAT (Basic)，其字母序号之和为：16+1+20+2+1+19+9+3=71，而 71 的二进制是 1000111，即有 3 个 0、4 个 1。

### 输入格式：

输入在一行中给出长度不超过 10​5​​、以回车结束的字符串。

### 输出格式：

在一行中先后输出 0 的个数和 1 的个数，其间以空格分隔。

### 输入样例：

PAT (Basic)

### 输出样例：

3 4

1058 选择题（20 分）

批改多选题是比较麻烦的事情，本题就请你写个程序帮助老师批改多选题，并且指出哪道题错的人最多。

### 输入格式：

输入在第一行给出两个正整数 N（≤ 1000）和 M（≤ 100），分别是学生人数和多选题的个数。随后 M 行，每行顺次给出一道题的满分值（不超过 5 的正整数）、选项个数（不少于 2 且不超过 5 的正整数）、正确选项个数（不超过选项个数的正整数）、所有正确选项。注意每题的选项从小写英文字母 a 开始顺次排列。各项间以 1 个空格分隔。最后 N 行，每行给出一个学生的答题情况，其每题答案格式为 (选中的选项个数 选项1 ……)，按题目顺序给出。注意：题目保证学生的答题情况是合法的，即不存在选中的选项数超过实际选项数的情况。

### 输出格式：

按照输入的顺序给出每个学生的得分，每个分数占一行。注意判题时只有选择全部正确才能得到该题的分数。最后一行输出错得最多的题目的错误次数和编号（题目按照输入的顺序从 1 开始编号）。如果有并列，则按编号递增顺序输出。数字间用空格分隔，行首尾不得有多余空格。如果所有题目都没有人错，则在最后一行输出 Too simple。

### 输入样例：

3 4 3 4 2 a c 2 5 1 b 5 3 2 b c 1 5 4 a b d e (2 a c) (2 b d) (2 a c) (3 a b e) (2 a c) (1 b) (2 a b) (4 a b d e) (2 b d) (1 e) (2 b c) (4 a b c d)

### 输出样例：

3 6 5 2 2 3 4

1059 C语言竞赛（20 分）

C 语言竞赛是浙江大学计算机学院主持的一个欢乐的竞赛。既然竞赛主旨是为了好玩，颁奖规则也就制定得很滑稽：

* 0、冠军将赢得一份“神秘大奖”（比如很巨大的一本学生研究论文集……）。
* 1、排名为素数的学生将赢得最好的奖品 —— 小黄人玩偶！
* 2、其他人将得到巧克力。

给定比赛的最终排名以及一系列参赛者的 ID，你要给出这些参赛者应该获得的奖品。

### 输入格式：

输入第一行给出一个正整数 *N*（≤10​4​​），是参赛者人数。随后 *N* 行给出最终排名，每行按排名顺序给出一位参赛者的 ID（4 位数字组成）。接下来给出一个正整数 *K* 以及 *K* 个需要查询的 ID。

### 输出格式：

对每个要查询的 ID，在一行中输出 ID: 奖品，其中奖品或者是 Mystery Award（神秘大奖）、或者是 Minion（小黄人）、或者是 Chocolate（巧克力）。如果所查 ID 根本不在排名里，打印 Are you kidding?（耍我呢？）。如果该 ID 已经查过了（即奖品已经领过了），打印 ID: Checked（不能多吃多占）。

### 输入样例：

6 1111 6666 8888 1234 5555 0001 6 8888 0001 1111 2222 8888 2222

### 输出样例：

8888: Minion 0001: Chocolate 1111: Mystery Award 2222: Are you kidding? 8888: Checked 2222: Are you kidding?

1060 爱丁顿数（25 分）

英国天文学家爱丁顿很喜欢骑车。据说他为了炫耀自己的骑车功力，还定义了一个“爱丁顿数” *E* ，即满足有 *E* 天骑车超过 *E* 英里的最大整数 *E*。据说爱丁顿自己的 *E* 等于87。

现给定某人 *N* 天的骑车距离，请你算出对应的爱丁顿数 *E*（≤*N*）。

### 输入格式：

输入第一行给出一个正整数 *N* (≤10​5​​)，即连续骑车的天数；第二行给出 *N* 个非负整数，代表每天的骑车距离。

### 输出格式：

在一行中给出 *N* 天的爱丁顿数。

### 输入样例：

10 6 7 6 9 3 10 8 2 7 8

### 输出样例：

6

1061 判断题（15 分）

判断题的评判很简单，本题就要求你写个简单的程序帮助老师判题并统计学生们判断题的得分。

### 输入格式：

输入在第一行给出两个不超过 100 的正整数 N 和 M，分别是学生人数和判断题数量。第二行给出 M 个不超过 5 的正整数，是每道题的满分值。第三行给出每道题对应的正确答案，0 代表“非”，1 代表“是”。随后 N 行，每行给出一个学生的解答。数字间均以空格分隔。

### 输出格式：

按照输入的顺序输出每个学生的得分，每个分数占一行。

### 输入样例：

3 6 2 1 3 3 4 5 0 0 1 0 1 1 0 1 1 0 0 1 1 0 1 0 1 0 1 1 0 0 1 1

### 输出样例：

13 11 12

1062 最简分数（20 分）

一个分数一般写成两个整数相除的形式：*N*/*M*，其中 *M* 不为0。最简分数是指分子和分母没有公约数的分数表示形式。

现给定两个不相等的正分数 *N*​1​​/*M*​1​​ 和 *N*​2​​/*M*​2​​，要求你按从小到大的顺序列出它们之间分母为 *K* 的最简分数。

### 输入格式：

输入在一行中按 *N*/*M* 的格式给出两个正分数，随后是一个正整数分母 *K*，其间以空格分隔。题目保证给出的所有整数都不超过 1000。

### 输出格式：

在一行中按 *N*/*M* 的格式列出两个给定分数之间分母为 *K* 的所有最简分数，按从小到大的顺序，其间以 1 个空格分隔。行首尾不得有多余空格。题目保证至少有 1 个输出。

### 输入样例：

7/18 13/20 12

### 输出样例：

5/12 7/12

1063 计算谱半径（20 分）

在数学中，矩阵的“谱半径”是指其特征值的模集合的上确界。换言之，对于给定的 *n* 个复数空间的特征值 { *a*​1​​+*b*​1​​*i*,⋯,*a*​*n*​​+*b*​*n*​​*i*}，它们的模为实部与虚部的平方和的开方，而“谱半径”就是最大模。

现在给定一些复数空间的特征值，请你计算并输出这些特征值的谱半径。

### 输入格式：

输入第一行给出正整数 N（≤ 10 000）是输入的特征值的个数。随后 N 行，每行给出 1 个特征值的实部和虚部，其间以空格分隔。注意：题目保证实部和虚部均为绝对值不超过 1000 的整数。

### 输出格式：

在一行中输出谱半径，四舍五入保留小数点后 2 位。

### 输入样例：

5 0 1 2 0 -1 0 3 3 0 -3

### 输出样例：

4.24

1064 朋友数（20 分）

如果两个整数各位数字的和是一样的，则被称为是“朋友数”，而那个公共的和就是它们的“朋友证号”。例如 123 和 51 就是朋友数，因为 1+2+3 = 5+1 = 6，而 6 就是它们的朋友证号。给定一些整数，要求你统计一下它们中有多少个不同的朋友证号。

### 输入格式：

输入第一行给出正整数 N。随后一行给出 N 个正整数，数字间以空格分隔。题目保证所有数字小于 10​4​​。

### 输出格式：

首先第一行输出给定数字中不同的朋友证号的个数；随后一行按递增顺序输出这些朋友证号，数字间隔一个空格，且行末不得有多余空格。

### 输入样例：

8 123 899 51 998 27 33 36 12

### 输出样例：

4 3 6 9 26

1065 单身狗（25 分）

“单身狗”是中文对于单身人士的一种爱称。本题请你从上万人的大型派对中找出落单的客人，以便给予特殊关爱。

### 输入格式：

输入第一行给出一个正整数 N（≤ 50 000），是已知夫妻/伴侣的对数；随后 N 行，每行给出一对夫妻/伴侣——为方便起见，每人对应一个 ID 号，为 5 位数字（从 00000 到 99999），ID 间以空格分隔；之后给出一个正整数 M（≤ 10 000），为参加派对的总人数；随后一行给出这 M 位客人的 ID，以空格分隔。题目保证无人重婚或脚踩两条船。

### 输出格式：

首先第一行输出落单客人的总人数；随后第二行按 ID 递增顺序列出落单的客人。ID 间用 1 个空格分隔，行的首尾不得有多余空格。

### 输入样例：

3 11111 22222 33333 44444 55555 66666 7 55555 44444 10000 88888 22222 11111 23333

### 输出样例：

5 10000 23333 44444 55555 88888

1066 图像过滤（15 分）

图像过滤是把图像中不重要的像素都染成背景色，使得重要部分被凸显出来。现给定一幅黑白图像，要求你将灰度值位于某指定区间内的所有像素颜色都用一种指定的颜色替换。

### 输入格式：

输入在第一行给出一幅图像的分辨率，即两个正整数 *M* 和 *N*（0<*M*,*N*≤500），另外是待过滤的灰度值区间端点 *A* 和 *B*（0≤*A*<*B*≤255）、以及指定的替换灰度值。随后 *M* 行，每行给出 *N* 个像素点的灰度值，其间以空格分隔。所有灰度值都在 [0, 255] 区间内。

### 输出格式：

输出按要求过滤后的图像。即输出 *M* 行，每行 *N* 个像素灰度值，每个灰度值占 3 位（例如黑色要显示为 000），其间以一个空格分隔。行首尾不得有多余空格。

### 输入样例：

3 5 100 150 0 3 189 254 101 119 150 233 151 99 100 88 123 149 0 255

### 输出样例：

003 189 254 000 000 000 233 151 099 000 088 000 000 000 255

1067 试密码（20 分）

当你试图登录某个系统却忘了密码时，系统一般只会允许你尝试有限多次，当超出允许次数时，账号就会被锁死。本题就请你实现这个小功能。

### 输入格式：

输入在第一行给出一个密码（长度不超过 20 的、不包含空格、Tab、回车的非空字符串）和一个正整数 N（≤ 10），分别是正确的密码和系统允许尝试的次数。随后每行给出一个以回车结束的非空字符串，是用户尝试输入的密码。输入保证至少有一次尝试。当读到一行只有单个 # 字符时，输入结束，并且这一行不是用户的输入。

### 输出格式：

对用户的每个输入，如果是正确的密码且尝试次数不超过 N，则在一行中输出 Welcome in，并结束程序；如果是错误的，则在一行中按格式输出 Wrong password: 用户输入的错误密码；当错误尝试达到 N 次时，再输出一行 Account locked，并结束程序。

### 输入样例 1：

Correct%pw 3 correct%pw Correct@PW whatisthepassword! Correct%pw #

### 输出样例 1：

Wrong password: correct%pw Wrong password: Correct@PW Wrong password: whatisthepassword! Account locked

### 输入样例 2：

cool@gplt 3 coolman@gplt coollady@gplt cool@gplt try again #

### 输出样例 2：

Wrong password: coolman@gplt Wrong password: coollady@gplt Welcome in

1068 万绿丛中一点红（20 分）

对于计算机而言，颜色不过是像素点对应的一个 24 位的数值。现给定一幅分辨率为 *M*×*N* 的画，要求你找出万绿丛中的一点红，即有独一无二颜色的那个像素点，并且该点的颜色与其周围 8 个相邻像素的颜色差充分大。

### 输入格式：

输入第一行给出三个正整数，分别是 *M* 和 *N*（≤ 1000），即图像的分辨率；以及 TOL，是所求像素点与相邻点的颜色差阈值，色差超过 TOL 的点才被考虑。随后 *N* 行，每行给出 *M* 个像素的颜色值，范围在 [0,2​24​​) 内。所有同行数字间用空格或 TAB 分开。

### 输出格式：

在一行中按照 (x, y): color 的格式输出所求像素点的位置以及颜色值，其中位置 x 和 y 分别是该像素在图像矩阵中的列、行编号（从 1 开始编号）。如果这样的点不唯一，则输出 Not Unique；如果这样的点不存在，则输出 Not Exist。

### 输入样例 1：

8 6 200 0 0 0 0 0 0 0 0 65280 65280 65280 16711479 65280 65280 65280 65280 16711479 65280 65280 65280 16711680 65280 65280 65280 65280 65280 65280 65280 65280 65280 165280 165280 65280 65280 16777015 65280 65280 165280 65480 165280 16777215 16777215 16777215 16777215 16777215 16777215 16777215 16777215

### 输出样例 1：

(5, 3): 16711680

### 输入样例 2：

4 5 2 0 0 0 0 0 0 3 0 0 0 0 0 0 5 0 0 0 0 0 0

### 输出样例 2：

Not Unique

### 输入样例 3：

3 3 5 1 2 3 3 4 5 5 6 7

### 输出样例 3：

Not Exist

1069 微博转发抽奖（20 分）

小明 PAT 考了满分，高兴之余决定发起微博转发抽奖活动，从转发的网友中按顺序每隔 N 个人就发出一个红包。请你编写程序帮助他确定中奖名单。

### 输入格式：

输入第一行给出三个正整数 M（≤ 1000）、N 和 S，分别是转发的总量、小明决定的中奖间隔、以及第一位中奖者的序号（编号从 1 开始）。随后 M 行，顺序给出转发微博的网友的昵称（不超过 20 个字符、不包含空格回车的非空字符串）。

### 输出格式：

按照输入的顺序输出中奖名单，每个昵称占一行。如果没有人中奖，则输出 Keep going...。

### 输入样例 1：

9 3 2 Imgonnawin! PickMe PickMeMeMeee LookHere Imgonnawin! TryAgainAgain TryAgainAgain Imgonnawin! TryAgainAgain

### 输出样例 1：

PickMe Imgonnawin! TryAgainAgain

### 输入样例 2：

2 3 5 Imgonnawin! PickMe

### 输出样例 2：

Keep going...

1070 结绳（25 分）

给定一段一段的绳子，你需要把它们串成一条绳。每次串连的时候，是把两段绳子对折，再如下图所示套接在一起。这样得到的绳子又被当成是另一段绳子，可以再次对折去跟另一段绳子串连。每次串连后，原来两段绳子的长度就会减半。



给定 *N* 段绳子的长度，你需要找出它们能串成的绳子的最大长度。

输入格式：

每个输入包含 1 个测试用例。每个测试用例第 1 行给出正整数 *N* (2≤*N*≤10

4

)；第 2 行给出 *N* 个正整数，即原始绳段的长度，数字间以空格分隔。所有整数都不超过10

4

。

输出格式：

在一行中输出能够串成的绳子的最大长度。结果向下取整，即取为不超过最大长度的最近整数。

输入样例：

8

10 15 12 3 4 13 1 15

输出样例：

14

1071 小赌怡情（15 分）

常言道“小赌怡情”。这是一个很简单的小游戏：首先由计算机给出第一个整数；然后玩家下注赌第二个整数将会比第一个数大还是小；玩家下注 t 个筹码后，计算机给出第二个数。若玩家猜对了，则系统奖励玩家 t 个筹码；否则扣除玩家 t 个筹码。

注意：玩家下注的筹码数不能超过自己帐户上拥有的筹码数。当玩家输光了全部筹码后，游戏就结束。

### 输入格式：

输入在第一行给出 2 个正整数 T 和 K（≤ 100），分别是系统在初始状态下赠送给玩家的筹码数、以及需要处理的游戏次数。随后 K 行，每行对应一次游戏，顺序给出 4 个数字：

n1 b t n2

其中 n1 和 n2 是计算机先后给出的两个[0, 9]内的整数，保证两个数字不相等。b 为 0 表示玩家赌小，为 1 表示玩家赌大。t表示玩家下注的筹码数，保证在整型范围内。

### 输出格式：

对每一次游戏，根据下列情况对应输出（其中 t 是玩家下注量，x 是玩家当前持有的筹码量）：

* 玩家赢，输出 Win t! Total = x.；
* 玩家输，输出 Lose t. Total = x.；
* 玩家下注超过持有的筹码量，输出 Not enough tokens. Total = x.；
* 玩家输光后，输出 Game Over. 并结束程序。

### 输入样例 1：

100 4 8 0 100 2 3 1 50 1 5 1 200 6 7 0 200 8

### 输出样例 1：

Win 100! Total = 200. Lose 50. Total = 150. Not enough tokens. Total = 150. Not enough tokens. Total = 150.

### 输入样例 2：

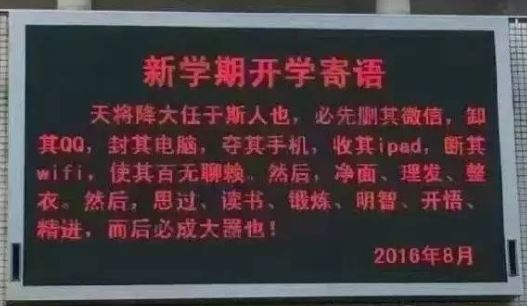
100 4 8 0 100 2 3 1 200 1 5 1 200 6 7 0 200 8

### 输出样例 2：

Win 100! Total = 200. Lose 200. Total = 0. Game Over.

1072 开学寄语（20 分）

下图是上海某校的新学期开学寄语：天将降大任于斯人也，必先删其微博，卸其 QQ，封其电脑，夺其手机，收其 ipad，断其 wifi，使其百无聊赖，然后，净面、理发、整衣，然后思过、读书、锻炼、明智、开悟、精进。而后必成大器也！



本题要求你写个程序帮助这所学校的老师检查所有学生的物品，以助其成大器。

输入格式：

输入第一行给出两个正整数 N（≤ 1000）和 M（≤ 6），分别是学生人数和需要被查缴的物品种类数。第二行给出 M 个需要被查缴的物品编号，其中编号为 4 位数字。随后 N 行，每行给出一位学生的姓名缩写（由 1-4 个大写英文字母组成）、个人物品数量 K（0 ≤ K ≤ 10）、以及 K 个物品的编号。

输出格式：

顺次检查每个学生携带的物品，如果有需要被查缴的物品存在，则按以下格式输出该生的信息和其需要被查缴的物品的信息（注意行末不得有多余空格）：

姓名缩写: 物品编号1 物品编号2 ……

最后一行输出存在问题的学生的总人数和被查缴物品的总数。

输入样例：

4 2

2333 6666

CYLL 3 1234 2345 3456

U 4 9966 6666 8888 6666

GG 2 2333 7777

JJ 3 0012 6666 2333

输出样例：

U: 6666 6666

GG: 2333

JJ: 6666 2333

3 5

1073 多选题常见计分法（20 分）

批改多选题是比较麻烦的事情，有很多不同的计分方法。有一种最常见的计分方法是：如果考生选择了部分正确选项，并且没有选择任何错误选项，则得到 50% 分数；如果考生选择了任何一个错误的选项，则不能得分。本题就请你写个程序帮助老师批改多选题，并且指出哪道题的哪个选项错的人最多。

### 输入格式：

输入在第一行给出两个正整数 N（≤1000）和 M（≤100），分别是学生人数和多选题的个数。随后 M 行，每行顺次给出一道题的满分值（不超过 5 的正整数）、选项个数（不少于 2 且不超过 5 的正整数）、正确选项个数（不超过选项个数的正整数）、所有正确选项。注意每题的选项从小写英文字母 a 开始顺次排列。各项间以 1 个空格分隔。最后 N 行，每行给出一个学生的答题情况，其每题答案格式为 (选中的选项个数 选项1 ……)，按题目顺序给出。注意：题目保证学生的答题情况是合法的，即不存在选中的选项数超过实际选项数的情况。

### 输出格式：

按照输入的顺序给出每个学生的得分，每个分数占一行，输出小数点后 1 位。最后输出错得最多的题目选项的信息，格式为：错误次数 题目编号（题目按照输入的顺序从1开始编号）-选项号。如果有并列，则每行一个选项，按题目编号递增顺序输出；再并列则按选项号递增顺序输出。行首尾不得有多余空格。如果所有题目都没有人错，则在最后一行输出 Too simple。

### 输入样例 1：

3 4 3 4 2 a c 2 5 1 b 5 3 2 b c 1 5 4 a b d e (2 a c) (3 b d e) (2 a c) (3 a b e) (2 a c) (1 b) (2 a b) (4 a b d e) (2 b d) (1 e) (1 c) (4 a b c d)

### 输出样例 1：

3.5 6.0 2.5 2 2-e 2 3-a 2 3-b

1074 宇宙无敌加法器（20 分）

地球人习惯使用十进制数，并且默认一个数字的每一位都是十进制的。而在 PAT 星人开挂的世界里，每个数字的每一位都是不同进制的，这种神奇的数字称为“PAT数”。每个 PAT 星人都必须熟记各位数字的进制表，例如“……0527”就表示最低位是 7 进制数、第 2 位是 2 进制数、第 3 位是 5 进制数、第 4 位是 10 进制数，等等。每一位的进制 d 或者是 0（表示十进制）、或者是 [2，9] 区间内的整数。理论上这个进制表应该包含无穷多位数字，但从实际应用出发，PAT 星人通常只需要记住前 20 位就够用了，以后各位默认为 10 进制。

在这样的数字系统中，即使是简单的加法运算也变得不简单。例如对应进制表“0527”，该如何计算“6203 + 415”呢？我们得首先计算最低位：3 + 5 = 8；因为最低位是 7 进制的，所以我们得到 1 和 1 个进位。第 2 位是：0 + 1 + 1（进位）= 2；因为此位是 2 进制的，所以我们得到 0 和 1 个进位。第 3 位是：2 + 4 + 1（进位）= 7；因为此位是 5 进制的，所以我们得到 2 和 1 个进位。第 4 位是：6 + 1（进位）= 7；因为此位是 10 进制的，所以我们就得到 7。最后我们得到：6203 + 415 = 7201。

### 输入格式：

输入首先在第一行给出一个 N 位的进制表（0 < N ≤ 20），以回车结束。 随后两行，每行给出一个不超过 N 位的非负的 PAT 数。

### 输出格式：

在一行中输出两个 PAT 数之和。

### 输入样例：

30527 06203 415

### 输出样例：

7201

1075 链表元素分类（25 分）

给定一个单链表，请编写程序将链表元素进行分类排列，使得所有负值元素都排在非负值元素的前面，而 [0, K] 区间内的元素都排在大于 K 的元素前面。但每一类内部元素的顺序是不能改变的。例如：给定链表为 18→7→-4→0→5→-6→10→11→-2，K 为 10，则输出应该为 -4→-6→-2→7→0→5→10→18→11。

### 输入格式：

每个输入包含一个测试用例。每个测试用例第 1 行给出：第 1 个结点的地址；结点总个数，即正整数N (≤10​5​​)；以及正整数K (≤10​3​​)。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 是该结点保存的数据，为 [−10​5​​,10​5​​] 区间内的整数；Next 是下一结点的地址。题目保证给出的链表不为空。

### 输出格式：

对每个测试用例，按链表从头到尾的顺序输出重排后的结果链表，其上每个结点占一行，格式与输入相同。

### 输入样例：

00100 9 10 23333 10 27777 00000 0 99999 00100 18 12309 68237 -6 23333 33218 -4 00000 48652 -2 -1 99999 5 68237 27777 11 48652 12309 7 33218

### 输出样例：

33218 -4 68237 68237 -6 48652 48652 -2 12309 12309 7 00000 00000 0 99999 99999 5 23333 23333 10 00100 00100 18 27777 27777 11 -1

1076 Wifi密码（15 分）

下面是微博上流传的一张照片：“各位亲爱的同学们，鉴于大家有时需要使用 wifi，又怕耽误亲们的学习，现将 wifi 密码设置为下列数学题答案：A-1；B-2；C-3；D-4；请同学们自己作答，每两日一换。谢谢合作！！~”—— 老师们为了促进学生学习也是拼了…… 本题就要求你写程序把一系列题目的答案按照卷子上给出的对应关系翻译成 wifi 的密码。这里简单假设每道选择题都有 4 个选项，有且只有 1 个正确答案。

输入格式：

输入第一行给出一个正整数 N（≤ 100），随后 N 行，每行按照 编号-答案 的格式给出一道题的 4 个选项，T 表示正确选项，F 表示错误选项。选项间用空格分隔。

输出格式：

在一行中输出 wifi 密码。

输入样例：

8

A-T B-F C-F D-F

C-T B-F A-F D-F

A-F D-F C-F B-T

B-T A-F C-F D-F

B-F D-T A-F C-F

A-T C-F B-F D-F

D-T B-F C-F A-F

C-T A-F B-F D-F

输出样例：

13224143

1077 互评成绩计算（20 分）

在浙大的计算机专业课中，经常有互评分组报告这个环节。一个组上台介绍自己的工作，其他组在台下为其表现评分。最后这个组的互评成绩是这样计算的：所有其他组的评分中，去掉一个最高分和一个最低分，剩下的分数取平均分记为 *G*​1​​；老师给这个组的评分记为 *G*​2​​。该组得分为 (*G*​1​​+*G*​2​​)/2，最后结果四舍五入后保留整数分。本题就要求你写个程序帮助老师计算每个组的互评成绩。

### 输入格式：

输入第一行给出两个正整数 *N*（> 3）和 *M*，分别是分组数和满分，均不超过 100。随后 *N* 行，每行给出该组得到的 *N* 个分数（均保证为整型范围内的整数），其中第 1 个是老师给出的评分，后面 *N*−1 个是其他组给的评分。合法的输入应该是 [0,*M*] 区间内的整数，若不在合法区间内，则该分数须被忽略。题目保证老师的评分都是合法的，并且每个组至少会有 3 个来自同学的合法评分。

### 输出格式：

为每个组输出其最终得分。每个得分占一行。

### 输入样例：

6 50 42 49 49 35 38 41 36 51 50 28 -1 30 40 36 41 33 47 49 30 250 -25 27 45 31 48 0 0 50 50 1234 43 41 36 29 42 29

### 输出样例：

42 33 41 31 37 39

1078 字符串压缩与解压（20 分）

文本压缩有很多种方法，这里我们只考虑最简单的一种：把由相同字符组成的一个连续的片段用这个字符和片段中含有这个字符的个数来表示。例如 ccccc 就用 5c 来表示。如果字符没有重复，就原样输出。例如 aba 压缩后仍然是 aba。

解压方法就是反过来，把形如 5c 这样的表示恢复为 ccccc。

本题需要你根据压缩或解压的要求，对给定字符串进行处理。这里我们简单地假设原始字符串是完全由英文字母和空格组成的非空字符串。

### 输入格式：

输入第一行给出一个字符，如果是 C 就表示下面的字符串需要被压缩；如果是 D 就表示下面的字符串需要被解压。第二行给出需要被压缩或解压的不超过 1000 个字符的字符串，以回车结尾。题目保证字符重复个数在整型范围内，且输出文件不超过 1MB。

### 输出格式：

根据要求压缩或解压字符串，并在一行中输出结果。

### 输入样例 1：

C TTTTThhiiiis isssss a tesssst CAaaa as

### 输出样例 1：

5T2h4is i5s a3 te4st CA3a as

### 输入样例 2：

D 5T2h4is i5s a3 te4st CA3a as10Z

### 输出样例 2：

TTTTThhiiiis isssss a tesssst CAaaa asZZZZZZZZZZ

1079 延迟的回文数（20 分）

给定一个 *k*+1 位的正整数 *N*，写成 *a*​*k*​​⋯*a*​1​​*a*​0​​ 的形式，其中对所有 *i* 有 0≤*a*​*i*​​<10 且 *a*​*k*​​>0。*N* 被称为一个**回文数**，当且仅当对所有 *i* 有 *a*​*i*​​=*a*​*k*−*i*​​。零也被定义为一个回文数。

非回文数也可以通过一系列操作变出回文数。首先将该数字逆转，再将逆转数与该数相加，如果和还不是一个回文数，就重复这个逆转再相加的操作，直到一个回文数出现。如果一个非回文数可以变出回文数，就称这个数为**延迟的回文数**。（定义翻译自 <https://en.wikipedia.org/wiki/Palindromic_number> ）

给定任意一个正整数，本题要求你找到其变出的那个回文数。

### 输入格式：

输入在一行中给出一个不超过1000位的正整数。

### 输出格式：

对给定的整数，一行一行输出其变出回文数的过程。每行格式如下

A + B = C

其中 A 是原始的数字，B 是 A 的逆转数，C 是它们的和。A 从输入的整数开始。重复操作直到 C 在 10 步以内变成回文数，这时在一行中输出 C is a palindromic number.；或者如果 10 步都没能得到回文数，最后就在一行中输出 Not found in 10 iterations.。

### 输入样例 1：

97152

### 输出样例 1：

97152 + 25179 = 122331 122331 + 133221 = 255552 255552 is a palindromic number.

1080 MOOC期终成绩（25 分）

对于在中国大学MOOC（<http://www.icourse163.org/> ）学习“数据结构”课程的学生，想要获得一张合格证书，必须首先获得不少于200分的在线编程作业分，然后总评获得不少于60分（满分100）。总评成绩的计算公式为 *G*=(*G*​*mid*−*term*​​×40%+*G*​*final*​​×60%)，如果 *G*​*mid*−*term*​​>*G*​*final*​​；否则总评 *G* 就是 *G*​*final*​​。这里 *G*​*mid*−*term*​​ 和 *G*​*final*​​分别为学生的期中和期末成绩。

现在的问题是，每次考试都产生一张独立的成绩单。本题就请你编写程序，把不同的成绩单合为一张。

### 输入格式：

输入在第一行给出3个整数，分别是 P（做了在线编程作业的学生数）、M（参加了期中考试的学生数）、N（参加了期末考试的学生数）。每个数都不超过10000。

接下来有三块输入。第一块包含 P 个在线编程成绩 *G*​*p*​​；第二块包含 M 个期中考试成绩 *G*​*mid*−*term*​​；第三块包含 N 个期末考试成绩 *G*​*final*​​。每个成绩占一行，格式为：学生学号 分数。其中学生学号为不超过20个字符的英文字母和数字；分数是非负整数（编程总分最高为900分，期中和期末的最高分为100分）。

### 输出格式：

打印出获得合格证书的学生名单。每个学生占一行，格式为：

学生学号 *G*​*p*​​ *G*​*mid*−*term*​​ *G*​*final*​​ *G*

如果有的成绩不存在（例如某人没参加期中考试），则在相应的位置输出“−1”。输出顺序为按照总评分数（四舍五入精确到整数）递减。若有并列，则按学号递增。题目保证学号没有重复，且至少存在1个合格的学生。

### 输入样例：

6 6 7 01234 880 a1903 199 ydjh2 200 wehu8 300 dx86w 220 missing 400 ydhfu77 99 wehu8 55 ydjh2 98 dx86w 88 a1903 86 01234 39 ydhfu77 88 a1903 66 01234 58 wehu8 84 ydjh2 82 missing 99 dx86w 81

### 输出样例：

missing 400 -1 99 99 ydjh2 200 98 82 88 dx86w 220 88 81 84 wehu8 300 55 84 84

1081 检查密码（15 分）

本题要求你帮助某网站的用户注册模块写一个密码合法性检查的小功能。该网站要求用户设置的密码必须由不少于6个字符组成，并且只能有英文字母、数字和小数点 .，还必须既有字母也有数字。

### 输入格式：

输入第一行给出一个正整数 N（≤ 100），随后 N 行，每行给出一个用户设置的密码，为不超过 80 个字符的非空字符串，以回车结束。

### 输出格式：

对每个用户的密码，在一行中输出系统反馈信息，分以下5种：

* 如果密码合法，输出Your password is wan mei.；
* 如果密码太短，不论合法与否，都输出Your password is tai duan le.；
* 如果密码长度合法，但存在不合法字符，则输出Your password is tai luan le.；
* 如果密码长度合法，但只有字母没有数字，则输出Your password needs shu zi.；
* 如果密码长度合法，但只有数字没有字母，则输出Your password needs zi mu.。

### 输入样例：

5 123s zheshi.wodepw 1234.5678 WanMei23333 pass\*word.6

### 输出样例：

Your password is tai duan le. Your password needs shu zi. Your password needs zi mu. Your password is wan mei. Your password is tai luan le.

1082 射击比赛（20 分）

本题目给出的射击比赛的规则非常简单，谁打的弹洞距离靶心最近，谁就是冠军；谁差得最远，谁就是菜鸟。本题给出一系列弹洞的平面坐标(x,y)，请你编写程序找出冠军和菜鸟。我们假设靶心在原点(0,0)。

### 输入格式：

输入在第一行中给出一个正整数 N（≤ 10 000）。随后 N 行，每行按下列格式给出：

ID x y

其中 ID 是运动员的编号（由 4 位数字组成）；x 和 y 是其打出的弹洞的平面坐标(x,y)，均为整数，且 0 ≤ |x|, |y| ≤100。题目保证每个运动员的编号不重复，且每人只打 1 枪。

### 输出格式：

输出冠军和菜鸟的编号，中间空 1 格。题目保证他们是唯一的。

### 输入样例：

3 0001 5 7 1020 -1 3 0233 0 -1

### 输出样例：

0233 0001

1083 是否存在相等的差（20 分）

给定 N 张卡片，正面分别写上 1、2、……、N，然后全部翻面，洗牌，在背面分别写上 1、2、……、N。将每张牌的正反两面数字相减（大减小），得到 N 个非负差值，其中是否存在相等的差？

### 输入格式：

输入第一行给出一个正整数 N（2 ≤ N ≤ 10 000），随后一行给出 1 到 N 的一个洗牌后的排列，第 i 个数表示正面写了 i 的那张卡片背面的数字。

### 输出格式：

按照“差值 重复次数”的格式从大到小输出重复的差值及其重复的次数，每行输出一个结果。

### 输入样例：

8 3 5 8 6 2 1 4 7

### 输出样例：

5 2 3 3 2 2

1084 外观数列（20 分）

外观数列是指具有以下特点的整数序列：

d, d1, d111, d113, d11231, d112213111, ...

它从不等于 1 的数字 d 开始，序列的第 n+1 项是对第 n 项的描述。比如第 2 项表示第 1 项有 1 个 d，所以就是 d1；第 2 项是 1 个 d（对应 d1）和 1 个 1（对应 11），所以第 3 项就是 d111。又比如第 4 项是 d113，其描述就是 1 个 d，2 个 1，1 个 3，所以下一项就是 d11231。当然这个定义对 d = 1 也成立。本题要求你推算任意给定数字 d 的外观数列的第 N 项。

### 输入格式：

输入第一行给出 [0,9] 范围内的一个整数 d、以及一个正整数 N（≤ 40），用空格分隔。

### 输出格式：

在一行中给出数字 d 的外观数列的第 N 项。

### 输入样例：

1 8

### 输出样例：

1123123111

1085 PAT单位排行（25 分）

每次 PAT 考试结束后，考试中心都会发布一个考生单位排行榜。本题就请你实现这个功能。

### 输入格式：

输入第一行给出一个正整数 N（≤10​5​​），即考生人数。随后 N 行，每行按下列格式给出一个考生的信息：

准考证号 得分 学校

其中准考证号是由 6 个字符组成的字符串，其首字母表示考试的级别：B代表乙级，A代表甲级，T代表顶级；得分是 [0, 100] 区间内的整数；学校是由不超过 6 个英文字母组成的单位码（大小写无关）。注意：题目保证每个考生的准考证号是不同的。

### 输出格式：

首先在一行中输出单位个数。随后按以下格式非降序输出单位的排行榜：

排名 学校 加权总分 考生人数

其中排名是该单位的排名（从 1 开始）；学校是全部按小写字母输出的单位码；加权总分定义为乙级总分/1.5 + 甲级总分 + 顶级总分\*1.5的**整数部分**；考生人数是该属于单位的考生的总人数。

学校首先按加权总分排行。如有并列，则应对应相同的排名，并按考生人数升序输出。如果仍然并列，则按单位码的字典序输出。

### 输入样例：

10 A57908 85 Au B57908 54 LanX A37487 60 au T28374 67 CMU T32486 24 hypu A66734 92 cmu B76378 71 AU A47780 45 lanx A72809 100 pku A03274 45 hypu

### 输出样例：

5 1 cmu 192 2 1 au 192 3 3 pku 100 1 4 hypu 81 2 4 lanx 81 2