**Problem A: 字符的距离（I）**

Time Limit: 1 Sec  Memory Limit: 128 MB  
Submit: 1794  Solved: 1241  
[[Submit](http://192.168.119.211/JudgeOnline/submitpage.php?cid=4639&pid=0&langmask=1022)][[Status](http://192.168.119.211/JudgeOnline/problemstatus.php?id=1906)]

**Description**

现有2个小写英文字母，请编程求出它们之间的距离。距离定义为两个字母之间的包含的其他字母的个数。如：a、c的距离就是1，因为a、c之间只有1个字母b。

**Input**

两个小写英文字母，中间用一个空格隔开。假定前一个字母小于后一个字母。

**Output**

输出一个不小于0的整数，表示两个字母之间的距离。

**Sample Input**

a c

**Sample Output**

1

**HINT**

**Problem B: 除法的差异**

Time Limit: 1 Sec  Memory Limit: 128 MB  
Submit: 2018  Solved: 1485  
[[Submit](http://192.168.119.211/JudgeOnline/submitpage.php?cid=4639&pid=1&langmask=1022)][[Status](http://192.168.119.211/JudgeOnline/problemstatus.php?id=1907)]

**Description**

我们知道，在C语言中，两个相同的操作数参与除法（“/”）运算，对于整数和实数的运算结果是不同的。现请编程求输入的两个整数进行整除和实数除法时的结果。

**Input**

输入只有一行。是两个int类型范围内的整数a和b。用一个空格隔开。

**Output**

输出有2行。

第1行是按照整除进行运算时，b/a的结果。

第2行是将a、b作为实数参与除法运算时，b/a的结果。保留3位小数。

**Sample Input**

3 10

**Sample Output**

3

3.333

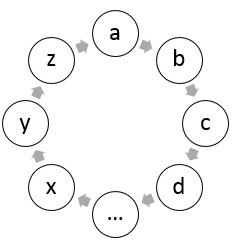
**HINT**

**Problem C: 字符的距离（II）**

Time Limit: 1 Sec  Memory Limit: 128 MB  
Submit: 4765  Solved: 1269  
[[Submit](http://192.168.119.211/JudgeOnline/submitpage.php?cid=4639&pid=2&langmask=1022)][[Status](http://192.168.119.211/JudgeOnline/problemstatus.php?id=1909)]

**Description**

       定义两个字母之间的距离是它们之间的字母的个数。将26个英文字母排成一个环，那么两个字母之间的间距既可以按照顺时针计算，也可以按照逆时针计算。如：字母a和c之间，如果按照顺时针，它们的距离就是1；如果按照逆时针计算，距离是23。定义同一个字母的顺时针和逆时针距离均为0。



        现在读取输入的两个小写英文字母，分别计算它们顺时针距离和逆时针距离。

**Input**

两个小写英文字母，中间用一个空格隔开。

**Output**

输出两个字母之间的顺时针距离和逆时针距离，中间用一个空格隔开。

**Sample Input**

a c

**Sample Output**

1 23

**HINT**

**Problem D: 第几个字母**

Time Limit: 1 Sec  Memory Limit: 16 MB  
Submit: 1601  Solved: 1044  
[[Submit](http://192.168.119.211/JudgeOnline/submitpage.php?cid=4639&pid=3&langmask=1022)][[Status](http://192.168.119.211/JudgeOnline/problemstatus.php?id=2411)]

**Description**

判断英文字母（A-Z）是第几个，A是第1个，至Z是第26个。

**Input**

输入只包含若干大写或小写英文字母，至EOF结束，无回车换行空格等其它字符。

**Output**

输出每个字母是第几个英文字母。

**Sample Input**

HelloWorld

**Sample Output**

8

5

12

12

15

23

15

18

12

4

**HINT**

**Problem E: Print Graphics Problerm (II)**

Time Limit: 1 Sec  Memory Limit: 2 MB  
Submit: 15513  Solved: 6572  
[[Submit](http://192.168.119.211/JudgeOnline/submitpage.php?cid=4639&pid=4&langmask=1022)][[Status](http://192.168.119.211/JudgeOnline/problemstatus.php?id=1096)]

**Description**

向标准输出上打印一些用ASCII字符组成的图形。

**Input**

输入为多个整数n，0<n<100。当n为0时结束输入。

**Output**

若n为偶数，则输出一个正向的n层等腰三角形；n为奇数，则输出一个倒向的n层等腰三角形。三角形由“+”组成。任意两个图形之间有一个空行分隔，格式见sample。

**Sample Input**

5

4

0

**Sample Output**

+++++++++

+++++++

+++++

+++

+

+

+++

+++++

+++++++

**HINT**

**Problem F: 它满足条件吗？**

Time Limit: 1 Sec  Memory Limit: 128 MB  
Submit: 11904  Solved: 6237  
[[Submit](http://192.168.119.211/JudgeOnline/submitpage.php?cid=4639&pid=5&langmask=1022)][[Status](http://192.168.119.211/JudgeOnline/problemstatus.php?id=1403)]

**Description**

需要判断给定的一个整数是否同时满足如下三个条件：

1. 它是一个完全平方数。

2. 它是一个偶数。

3. 它是一个正数。

注：若一个数能表示成某个自然数的平方的形式，则称这个数为完全平方数。例如：

0,1,4,9,16,25,36,49,64,81,100,121,144,169,196,225,256,289,324,361,400,441,484,529

**Input**

一个int范围内的整数。

**Output**

如果输入的数字满足条件，则输出yes，否则输出no。

**Sample Input**

100

**Sample Output**

yes

**HINT**

注意逻辑表达式判断三个条件的顺序。

如果你不会使用分支语句，同样可以使用条件运算符实现该程序。

库函数sqrt()可以用于求一个数的平方根。

**Problem G: 输出整数的最低两位**

Time Limit: 1 Sec  Memory Limit: 2 MB  
Submit: 14456  Solved: 6625  
[[Submit](http://192.168.119.211/JudgeOnline/submitpage.php?cid=4639&pid=6&langmask=1022)][[Status](http://192.168.119.211/JudgeOnline/problemstatus.php?id=1088)]

**Description**

把一个整数的最低两位打印出来，不输出整数的符号。

**Input**

输入为一个整数n，不会超出int类型的数据范围。

**Output**

输出n的最低两位数字。但是，输入的数字本身不足两位时，不应当补0。如，输入为“1”，则输出为“1”。

**Sample Input**

-102

**Sample Output**

02

**HINT**

printf函数可以完成补0的操作。

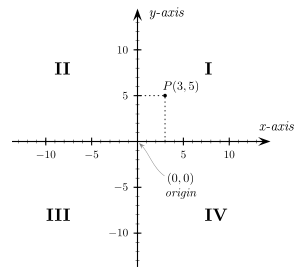
**Problem H: 笛卡尔平面**

Time Limit: 1 Sec  Memory Limit: 16 MB  
Submit: 16774  Solved: 5599  
[[Submit](http://192.168.119.211/JudgeOnline/submitpage.php?cid=4639&pid=7&langmask=1022)][[Status](http://192.168.119.211/JudgeOnline/problemstatus.php?id=1162)]

**Description**

在数学里，笛卡儿坐标系（Cartesian坐标系），也称直角坐标系，是一种正交坐标系。二维的直角坐标系是由两条相互垂直、0 点重合的数轴构成的。在平面内，任何一点的坐标 是根据数轴上对应的点的坐标设定的。

二维的直角坐标系通常由两个互相垂直的坐标轴设定，通常分别称为 x-轴 和 y-轴；两个坐标轴的相交点，称为原点，通常标记为 O ，既有“零”的意思，又是英语“Origin”的首字母。每一个轴都指向一个特定的方向。这两个不同线的坐标轴，决定了一个平面，称为 xy-平面，又称为笛卡儿平面。



以笛卡尔平面上一个点的坐标(x,y)坐标为输入，输出该点是否在坐标轴上或者位于哪一个象限。

**Input**

输入为一行。输入的坐标x、y值都会带有1位小数，括号和逗号都是英文半角符号，逗号后面有一个空格。

**Output**

四个象限输出为“quadrand I”、“quadrand II”、“quadrand III”、“quadrand IV”，x-轴和y-轴输出为“x axis”和“y axis”，原点输出为“origin”

输出为以下三种：

(-1.0, -2.5) is in quadrand III

(0.0, 4.8) is on the y axis

(0.0, 0.0) is origin

**Sample Input**

(-1.0, -2.5)

**Sample Output**

(-1.0, -2.5) is in quadrand III

**HINT**