**Python语言**

**Description**

Python是一种面向对象的解释型计算机程序设计语言，在web编程、图形处理、科学计算等方面有着广泛应用。Python的作者有意的设计限制性很强的语法，使得不好的编程习惯都不能通过编译。其中很重要的一项就是Python的缩进规则。

Python语言利用缩进表示语句块的开始和退出，而非使用花括号或者某种关键字。增加缩进表示语句块的开始，而减少缩进则表示语句块的退出。缩进成为了语法的一部分。例如if语句：

==============================================

if age<18:

    print("DDjing不能买酒。")

    print("不过DDjing能买口香糖。")

print("这句话处于if语句块的外面。")

==============================================

这个缩进表示，前两个print函数是if语句的子语句，属于一个代码块，第三个print则是和if并列的下一条语句，每级缩进使用4个空格来表示。在if、while、def、class等语句后需要缩进⋯⋯说白了，当一行代码的最后是':'时，便说明下一行要缩进了。

DDjing由于写C++时养成了不好的缩进习惯，代码风格就很糟糕，写的python程序都不能通过编译，他希望你能够写一个程序让他的代码自动实现正确缩进。

幸运的是DDjing在每个代码块结束后都打了一个空行，这样你应该感觉简单多了吧（但DDjing依然很懒，程序最后该有的空行他都没有打），而且DDjing希望你同时在他代码中所有','的后面加一个空格，让代码更加美观。

**Input**

输入一段完全没有任何缩进的python代码，不超过100行，每行不超过100个字符（包括完成缩进后），保证每个代码段结束后都有一个空行（程序最后的空行除外，但你也要按格式把它们输出来，最后输出的一行空格数为0），保证每个逗号后面都没有空格。

**Output**

输出正确缩进后的python代码，空行与上一行的缩进相同，每个逗号后面都有一个空格。

**Sample Input**

def getsubset(myset,subtract):

if len(myset)<=1:

return []

result=[]

newsubtract=subtract.copy()

for i in subtract:

result.append(myset-{i})

newsubtract=newsubtract-{i}

result.extend(getsubset(myset-{i},newsubtract))

return result

**Sample Output**

def getsubset(myset, subtract):

if len(myset)<=1:

return []

result=[]

newsubtract=subtract.copy()

for i in subtract:

result.append(myset-{i})

newsubtract=newsubtract-{i}

result.extend(getsubset(myset-{i}, newsubtract))

return result

**HINT**

OJ问题，输出样例的最后一行的空行（空格数为0）没显示出来，注意一下。

**题解**

模拟题。

读一行处理一行即可，根据行末是否有“：”来确定是否增加缩进；根据读入空行来决定是否减少缩进。输出过程中遇到“，”在后面多输出一个空格。最后的空行，每一行减少一段缩进，直至缩进为0。