To prepare for the English exam Little Ho collected many digital reading materials. Unfortunately the materials are messed up by a malware.

It is known that the original text contains only English letters (a-zA-Z), spaces, commas, periods and newlines, conforming to the following format:

1. Each sentence contains at least one word, begins with a letter and ends with a period.

2. In a sentence the only capitalized letter is the first letter.

3. In a sentence the words are separated by a single space or a comma and a space.

4. The sentences are separated by a single space or a single newline.

It is also known the malware changes the text in the following ways:

1. Changing the cases of letters.

2. Adding spaces between words and punctuations.

Given the messed text, can you help Little Ho restore the original text?

输入

A string containing no more than 8192 English letters (a-zA-Z), spaces, commas, periods and newlines which is the messed text.

输出

The original text.

**样例输入**

my Name is Little Hi.

His name IS Little ho , We are friends.

**样例输出**

My name is little hi.

His name is little ho, we are friends.

首先考试的时候脑子有坑啊……

第一个自己挖的坑：

看到输入的时候想的不是单行处理输出而是存起来一起输出，然后就开始纠结怎么判断回车，未果，死亡。

第二个自己挖的坑

分类讨论的结果未完全被题目拉着走，下意识分为了单词+逗号，单词+句号，单词，单逗号，单句号……你他喵的是逗逼么！

后来在补题的时候终于想起来分成逗号，句号，单词并分类讨论，遂ac.

问题：

1. 思路不够灵活，一旦跳进自己的思维惯性陷阱里就没法及时的跳出来。
2. 分类模糊，且没有效率，得更加的明确，凡是都得打个问号。

#include <iostream>

#include<sstream>

#include<iomanip>

#include<string>

#include<vector>

#include<stack>

#include<algorithm>

#define hash 997

#define MAX 1000000000

#define ll long long

using namespace std;

vector<string>text;

string temp;

string store;

string result;

int main()

{

while (getline(cin, store))

{

for (int i = 0; i < store.size(); i++)

{

if (store[i] <= 'Z'&&store[i] >= 'A')

store[i] += 32;

if (store[i] == '.' || store[i] == ',')

{

if (!temp.empty())

text.push\_back(temp);

temp = store[i];

text.push\_back(temp);

temp.clear();

}

else if (store[i] != ' ')

{

temp.push\_back(store[i]);

}

else

{

if (!temp.empty())

{

text.push\_back(temp);

temp.clear();

}

}

}

if (!temp.empty())

{

text.push\_back(temp);

temp.clear();

}

int flag = 1;

for (int i = 0; i < text.size(); i++)

{

if (flag)

{

flag = 0;

text[i][0] -= 32;

result += text[i];

}

else if (text[i] == ".")

{

if (i == text.size() - 1)

result += text[i];

else

{

result += text[i] + ' ';

flag = 1;

}

}

else if (text[i] == ",")

{

result += text[i];

}

else

{

result += (" " + text[i]);

}

}

cout << result << endl;

result.clear();

text.clear();

}

}