#include <iostream>

#include<sstream>

#include<cstdio>

#include<iomanip>

#include<string>

#include<vector>

#include<stack>

#include<queue>

#include<algorithm>

#include<map>

#include<cmath>

#include<climits>

#include<cstring>

#define hash 10007

#define MAX 900005

#define ll long long

using namespace std;

vector<int>next\_store;

vector<int>start;

void reset(string shift)

{

next\_store.clear();

start.clear();

next\_store.resize(shift.size(), 0);

next\_store[0] = 0;

int num = shift.size();

for (int i = 0; i < num-1; i++)

{

if (shift[i + 1] == shift[next\_store[i]])

next\_store[i + 1] = next\_store[i] + 1;

}

}

void output()

{

if (start.size())

cout << start[0];

for (int i = 1; i < start.size(); i++)

cout << " " << start[i];

cout << endl;

}

char shift\_c[MAX], text\_c[MAX];

int main()

{

string shift, text;

while (fgets(shift\_c,MAX,stdin))

{

fgets(text\_c, MAX, stdin);

shift = shift\_c;

text = text\_c;

shift.erase(shift.size() - 1);

text.erase(text.size() - 1);

reset(shift);

int num\_t = text.size();

int num\_s = shift.size();

for (int i = 0; i < num\_t; i++)

{

if (i + num\_s>num\_t)

break;

int flag = 1;

for (int j = 0; j < num\_s; j++)

{

if (shift[j] != text[i + j])

{

if (j == 0)

{

flag = 0;

break;

}

else

{

flag = 0;

i += j - next\_store[j - 1] - 1;

break;

}

}

}

if (flag)

{

start.push\_back(i);

i += num\_s - next\_store[next\_store.size() - 1] - 1;

}

}

output();

}

}