

## KMP

```
#include <bits/stdc++.h>
using namespace std;

int n,k,len1,len2;
int next1[1000001];
char s1[1000001],s2[1000001];
inline void get_next(){ //next数组是从 s[0到i-1]前子串 的前缀后缀最大值
    int i=0,j;
    next1[0]=j=-1;
    while(i<len2){
        if(j==-1 || s2[i]==s2[j]) //类似于KMP的匹配
            next1[++i]=++j;
        else j=next1[j]; //失配
    }
}
inline void KMP(){
    int i=0,j=0; //从0位开始匹配
    while(i<len1){ //临界值
        if(j==-1 || s1[i]==s2[j]) i++,j++;
        else j=next1[j]; //失配
        if(j==len2) printf("%d\n",i-len2+1),j=next1[j]; //t2==lenn2时，匹配成功； i-
len2+1即为第一个字母的位置
    } //匹配成功后，j置为next[j]
}
int main(){
    scanf("%s",s1);
    scanf("%s",s2);
    len1=strlen(s1);
    len2=strlen(s2);
    get_next();
    KMP();
    for(int i=1;i<=len2;++i)
        printf("%d ",next1[i]); //输出next数组
    return 0;
}
```