实验五：2021/4/21

一：

（1）给定一行字符串文本为主串，另一个字符串为模式串，在主串中查找模式串首次出现的位置，并返回其位置下标。

（2）思路分析：在主串中查找与模式串第一个字符相同的位置，然后依次比较，直到之后的每一位都相同，输出这个位置。如果有不同则继续向下移动初始值。

（3）代码实现：

法一：（我的解法）

#include <stdio.h>

#include <string.h>

int BF(char S[],char T[]){

int i,k,m;

int j = strlen(T);

for( i = 0 ; i < strlen(S) - j ; i++ ){

k = i ;

m = 0 ;

while(k <= i + j){

if( S[k++] != T[m++] )

break;

if( k == i+j ){

return i + 1;

}

}

}

printf("error!");

return 0;

}

int main(){

char S[19] = "ababcaababbabccbab";

char T[13] = "ababbabc";

int a = BF(S,T);

printf("%d",a);

return 0;

}

法二：（老师解法）

int BF(char T[],char S[]){

int index = 0;

int i = 0 , j = 0;

while(S[i] != '\0' && T[j] != '\0'){

if(S[i] == T[j]){

i++;

j++;

}else{

j = 0;

index++;

i = index;

}

}

if(T[j] == '\0')

return index+1;

else

return -1;

}

1. 运行结果：



二：

（1）字符串的真前/后缀指不包括字符串自身的前缀/后缀。给定一个字符串，编写函数，求解字符串T所有前缀子串（包括自己）的最大相等真前后缀的长度，并存入next数组中。

（2）思路分析：从长度为1开始判断，长度为1时输出0；长度为n（n>=2），判断前n-1个字符和后n-1个字符是否相同，若不相同则减小比较的长度。知道找出相等真前后缀为止。

（3）代码实现：

#include <stdio.h>

#include <string.h>

void MaxAffixLength(char T[],int next[]){

int i,j,len;

memset(next,0,12\*sizeof(int));

next[0]= 0;

for( i = 1 ; T[i] != '\0' ; i++ ){//计算[0,i]的最大真前后缀的长度

for( len = i-1 ; len >= 1 ; len-- ){

for( j = 0 ; j < len ; j++ ){ //比较[0,len-1]和[i-len+1,i]

if( T[j] != T[i-len+j+1] ) break;

}

if( j == len ){

next[i] = len;

break;

}

}

}

}

void main(){

int i;

char T[13] = "ababaaababaa";

int next[12] = {0};

MaxAffixLength(T,next);

for(i = 0 ; i < 12 ; i++ ){

printf("%d ",next[i]);

}

}

（4）运行结果：

