#include<stdio.h>

#include<string.h>

#define MaxStrSize 256

typedef struct{

char ch[MaxStrSize];

int length;

}SeqString;

int PartPosition(SeqString s1,SeqString s2,int k)

{

int i,j;

i=k-1;

j=0;

while(i<s1.length && j<s2.length)

if(s1.ch[i]==s1.ch[j])

{

i++;

j++;

}

else

{

i=i-j+1;

}

if(j>=s2.length)

return i+1-s2.length;

else return -1;

}

void CreatTextFile()

{

SeqString S;

char fname[10],yn;

FILE \*fp;

printf("请输入要建立的文件名");

scanf("%s",fname);

fp=fopen(fname,"w");

yn='n';

while(yn=='n' || yn=='N'){

printf("输入一行文本：");

getchar();

gets(S.ch);

S.length=strlen(S.ch);

fwrite(&S,sizeof(S),1,fp);

printf("结束输入吗？y or n:");

yn=getchar();

}

fclose(fp);

printf("建立文件结束！\n");

}

void SubStrCount()

{

FILE \*fp;

SeqString S,T;

char fname[10] ;

int i=0,j,k;

printf("输入文本文件名");

scanf("%s",fname);

fp=fopen(fname,"r");

printf("输入要统计计数的单词");

scanf("%s",T.ch);

T.length=strlen(T.ch);

while(!feof(fp)){

fread(&S,sizeof(S),1,fp);

if(feof(fp)) break;

k=1;

while(k<S.length)

{

j=PartPosition(S,T,k);

if(j<0) break;

else{

i++;

k=j+T.length;

}

}

}

printf("\n单词%s在文本文件%s中共出现%d次\n",T.ch,fname,i);

}

void SubStrInd()

{

FILE \*fp;

SeqString S,T;

char fname[10];

int i,j,k,l,m;

int wz[20];

printf("输入文本文件名：");

scanf("%s",fname);

fp=fopen(fname,"r");

printf("输入需要检索的单词：");

scanf("%s",T.ch);

T.length=strlen(T.ch);

l=0;

while(!feof(fp)){

fread(&S,sizeof(S),l,fp);

if(feof(fp))break;

l++;

k=1;

i=0;

while(k<S.length)

{

j=PartPosition(S,T,k);

if(j<0)break;

else{

i++;

wz[i]=j;

k=j+T.length;

}

}

if(i>0){

printf("行号：%d,次数：%d,位置分别为：",l,i);

for(m=1;m<=i;m++) printf("%4d",wz[m]);

printf("\n");

}

}

}

int main(void)

{

void CreatTextFile(),SubStrCount(),SubStrInd();

int xz;

do{

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\*文本文件的检索，子串的统计及定位\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\* 1.建立文本文件 \*\n");

printf("\* 2.单词子串的计数 \*\n");

printf("\* 3.单词子串的定位 \*\n");

printf("\* 4.退出整个程序 \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\* 请选择：[1-4] \*\n");

scanf("%d",&xz);

switch(xz){

case 1:CreatTextFile();break;

case 2:SubStrCount();break;

case 3:SubStrInd();break;

case 4:return 0;

default: printf("选择错误！请重新选择！\n");

}

}while(1);

}

#include<stdio.h>

#include<string.h>

#define MaxStrSize 999

typedef struct{

char ch[MaxStrSize];//ch可容999个字符

int length;

}SeqString;//定义字符串

int Index\_KMP(SeqString s1,SeqString s2,int k)

{

int i,j;

i=k-1;

j=0;

while(i<s1.length && j<s2.length)//两个串均为比较到串未

if(s1.ch[i]==s1.ch[j])//继续比较后继字符

{

i++;

j++;

}

else //模式串向右移动

{

i=i-j+1;

}

if(j>=s2.length) //匹配成功

return i+1-s2.length;

else return -1;//匹配失败

}

void CreatTextFile()

{

SeqString S;

char fname[10],yn;

FILE \*fp;

printf("输入建立文件名");

scanf("%s",fname);

fp=fopen(fname,"w");

yn='n'; //输入结束标准初值

while(yn=='n' || yn=='N'){

printf("输入一行文本：");

getchar();

gets(S.ch);

S.length=strlen(S.ch);

fwrite(&S,sizeof(S),1,fp);

printf("y结束n继续:");

yn=getchar();

}

fclose(fp);//关闭文件

printf("建立文件结束！\n");

}

void SubStrCount()

{

FILE \*fp;

SeqString S,T;

char fname[10] ;

int i=0,j,k;

printf("输入文本文件名");

scanf("%s",fname);

fp=fopen(fname,"r");

printf("输入要统计计数的单词");

scanf("%s",T.ch);

T.length=strlen(T.ch);

while(!feof(fp)){ //扫描文件夹

fread(&S,sizeof(S),1,fp);//读入文本

k=1;

while(k<S.length)//搜索主串S

{

j=Index\_KMP(S,T,k);//调用串匹配函数

if(j<0) break;

else{

i++; //计数+1

k=j+T.length;//继续搜索

}

}

}

printf("\n单词%s在文本文件%s中共出现%d次\n",T.ch,fname,i);

}

void SubStrInd()

{

FILE \*fp;

SeqString S,T;

char fname[10];

int i,j,k,l,m;

int wz[20];//存放一行中子串匹配的多个位置

printf("输入文本文件名：");

scanf("%s",fname);

fp=fopen(fname,"r");

printf("输入需要检索的单词：");

scanf("%s",T.ch);

T.length=strlen(T.ch);

l=0; //计数器至0

while(!feof(fp)){

fread(&S,sizeof(S),l,fp);//读入文本

l++;

k=1;//初始化开始搜索位置

i=0;//初始化单词计数器

while(k<S.length)//搜索整个主串

{

j=Index\_KMP(S,T,k);//调用串函数函数

if(j<0)break;

else{

i++;//计数器+1

wz[i]=j;//记录单词位置

k=j+T.length;//继续下一个串搜索

}

}

if(i>0){

printf("行号：%d,次数：%d,位置分别为：",l,i);

for(m=1;m<=i;m++) printf("%4d",wz[m]);

printf("\n");

}

}

}

int main(void)

{

void CreatTextFile(),SubStrCount(),SubStrInd();

int xz;

do{

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\*文本文件的检索，子串的统计及定位\*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\* 1.建立文本文件 \*\n");

printf("\* 2.单词子串的计数 \*\n");

printf("\* 3.单词子串的定位 \*\n");

printf("\* 4.退出整个程序 \*\n");

printf("\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\n");

printf("\* 请选择：[1-4] \*\n");

scanf("%d",&xz);

switch(xz){

case 1:CreatTextFile();break;

case 2:SubStrCount();break;

case 3:SubStrInd();break;

case 4:return 0;

default: printf("选择错误！请重新选择！\n");

}

}while(1);

}