**题目7：文本文件单词的检索及计数（学时：6）**

要求编程建立一个文本文件，每个单词不包括空格及跨行，单词由字符序列构成且区分大小写，完成以下功能：统计给定单词在文本文件中出现的总次数、检索输出某单词在文本文件中首次出现的行号及位置。

代码：

#include<stdio.h>

#include<stdlib.h>

#include<malloc.h>

#include<string.h>

#define N 250

typedef struct Cstring //定义结构体(字符长串）

{

char string[N];

}Cstring;

typedef struct File //定义结构体（文件）

{

Cstring filename[N];//文件数组

FILE \*in, \*out; //文件指针（入/出）

int line[N], filenum ; //行编号，文件编号

}File;

typedef struct Line //结构体（行）

{

int pos[N],counter;//所在行号，计数器

}Line;

int Length(char\* a)//函数（判断单词是否结束）返回单词长度

{

int i=0;

while(a[i]!='\0'&&a[i]!='\n')

i++;

return i;

}

int Index(Cstring a,Cstring b,Line\* l)//索引（字符总串，单词，要查找的字符总串第几行）

{

int i,k,j,p=1,num=0;

l->counter=0;

Cstring temp;

while(p<=a.string[0])//sring[0]用于存字符串大小

{

i=1;

while(a.string[p]!=' '&&a.string[p]!='\0'&&a.string[p]!=','&&a.string[p]!='.')//将字符总串的一个单词放到temp中，并没进行一次，num++

{

temp.string[i]=a.string[p];

i++;

p++;

}

temp.string[i]='\0';

temp.string[0]=Length(&(temp.string[1]));

num++;

k=1;

j=1;

if(temp.string[0]==b.string[0])

{

while(j<=b.string[0])

{

if(temp.string[k]==b.string[j])

{

k++;

j++;

}

else break;

}

}

if(j>b.string[0])//说明2个单词相同

{

l->pos[l->counter]=num;//该行第l->counter的单词所在位置(即该单词为该行第几个单词)

l->counter++; //行计数器加1

}

while(a.string[p]==' '||a.string[p]==','||a.string[p]=='.')

p++;

}

return 1;

}

int Equal(Cstring a,Cstring b) //是否相同，相同返回1

{

int i=1,j=1;

if(a.string[0]!=b.string[0])

return 0;

while(i<=a.string[0]&&j<=b.string[0])

{

if(a.string[i]==b.string[j])

{

i++; j++;

}

else return 0;

}

return 1;

}

int CreateFile(File\* F) //创建文件

{

Cstring s;

char c;

printf("请输入文件名\n");

scanf("%s",&(F->filename[F->filenum].string[1]));//名字放置区

F->filename[F->filenum].string[0]=Length(&(F->filename[F->filenum].string[1]));//文件名字符串长度

if(!(F->in=fopen(&(F->filename[F->filenum].string[1]),"w")))//以写的方式打开文件，文件指针 F->in

{

printf("不能打开文件\n");

exit(0);

}

F->line[F->filenum]=0;//该文件行数初始为0

printf("请输入文件内容(结束符为'#')\n");

while(gets(&(s.string[1])))//每次输一行时就会执行一次循环

{

s.string[0]=Length(&s.string[1]);

c=s.string[s.string[0]];

if(s.string[0]!=0)

{

if(c=='#')

s.string[s.string[0]]='\n';

fputs(&(s.string[1]),F->in);//将所有字符存到文件中

fputc('\n',F->in);//存入一个换行符

F->line[F->filenum]++;//该文件行数加1

}

if(c=='#') break;

}

fclose(F->in);//关闭文件

F->filenum++;//文件数组下标加1，指向下一个文件编号

return 1;

}

int CountString(File\* F) //计数

{

Cstring b,a,s;

int k,counter=0,line=0,i;

Line l[N];

printf("请输入检索的文件名:");

scanf("%s",&(b.string[1]));

b.string[0]=Length(&(b.string[1]));

for(k=0;k<F->filenum;k++) //找到该文件名

{

if(Equal(F->filename[k],b)) break;

}

if(k==F->filenum)

{

printf("文件不存在\n");

return 0;

}

if(!(F->out=fopen(&F->filename[k].string[1],"r")))

{

printf("不能打开文件\n");

exit(0);

}

//找到了并能正常打开文件编号为k

printf("请输入搜索的单词：");

scanf("%s",&(s.string[1]));

s.string[0]=Length(&(s.string[1]));

while(line<F->line[k])//行数小于k文件的总行数

{

fgets(&(a.string[1]),N,F->out);//从文件中将内容读取到a地址处

a.string[0]=Length(&(a.string[1]));

Index(a,s,&l[line]);//索引

line++;//进入下一行

}

for(i=0;i<line;i++) //通过每行的计数器，算出单词出现的总次数，计入counter

{

if(l[i].counter!=0)

counter+=l[i].counter;

}

printf("文件中出现该单词的个数为：");

printf("%d\n",counter);

fclose(F->out);//关闭文件

return 1;

}

int PotString(File\* F) //单词定位

{

Cstring b,s,a;

Line l[N];

int k,line=0,i,j;

printf("请输入检索的文件名:");

scanf("%s",&(b.string[1]));

b.string[0]=Length(&(b.string[1]));

for(k=0;k<F->filenum;k++) //找到该文件编号

{

if(Equal(F->filename[k],b)) break;

}

if(k==F->filenum)

{

printf("文件不存在\n");

return 0;

}

if(!(F->out=fopen(&(F->filename[k].string[1]),"r")))

{

printf("不能打开文件\n");

exit(0);

}

//找到了并能正常打开文件编号为k

printf("请输入要定位的单词：");

scanf("%s",&(s.string[1]));

s.string[0]=Length(&(s.string[1]));

while(line<F->line[k])//行数小于k文件的总行数

{

fgets(&(a.string[1]),N,F->out);//从文件中将内容读取到a地址处

a.string[0]=Length(&(a.string[1]));

Index(a,s,&l[line]);//索引

line++;

}

printf("该单词首次出现的行号以及位置：\n");

for(i=0;i<line;i++)

{

if(l[i].counter>0)

{

printf("行号：");printf("%-3d",i+1);

printf("所在位置：");

for(j=0;j<l[i].counter;j++)

{

printf("第");

printf("%d",l[i].pos[j]);

printf("个单词 ");

}

printf("\n");

break;

}

}

return 1;

}

int main()

{

int i;

File F;

F.filenum=0;

printf("-----------------文本文件单词的检索及计数------------------- \n");

printf("\n 1. 创建文件\n 2. 单词计数\n 3. 单词定位\n 4. 退出\n");

printf("请选择操作:\n");

while(scanf("%d",&i)&&i!=4)

{

switch(i)

{

case 1:CreateFile(&F);printf("\n请选择操作:\n");break;

case 2:{CountString(&F); printf("\n请选择操作:\n"); break;}

case 3:{PotString(&F); printf("\n请选择操作:\n"); break; }

default:printf("\n操作错误\n"),i=4;

}

}

return 0;

}

运行结果：

