|  |  |
| --- | --- |
| 2 | Write a program to design Lexical Analyzer in C/C++ Language (to recognize any five keywords, identifiers, numbers, operators and punctuations) |

#include<stdio.h>

#include<ctype.h>

#include<string.h>

#include<stdlib.h>

void keyw(char \*p);

int i=0,id=0,kw=0,num=0,op=0,pun=0;

char keys[32][10]={"auto","break","case","char","const","continue","default",

"do","double","else","enum","extern","float","for","goto",

"if","int","long","register","return","short","signed",

"sizeof","static","struct","switch","typedef","union",

"unsigned","void","volatile","while"};

char punc[]=".,?!";

main()

{

char ch,str[25],seps[15]="\t\n (){}[];#\"<>",oper[]="%^&\*-+=~|<>/";

int j;

char fname[50];

FILE \*f1;

printf("enter filename");

scanf("%s",fname);

f1 = fopen(fname,"r");

if(f1==NULL)

{

printf("file not found");

exit(0);

}

while((ch=fgetc(f1))!=EOF)

{

for(j=0;j<=3;j++)

{

if(ch==punc[j])

{

printf("%c is a punctuation\n",ch);

pun++;

str[i]='\0';

break;

}

}

for(j=0;j<=14;j++)

{

if(ch==oper[j])

{

printf("%c is an operator\n",ch);

op++;

str[i]='\0';

keyw(str);

}

}

for(j=0;j<=15;j++)

{

if(i==-1)

break;

if(ch==seps[j])

{

if(ch=='#')

{

while(ch!='>')

{

printf("%c",ch);

ch=fgetc(f1);

}

printf("%c is a header file\n",ch);

i=-1;

break;

}

if(ch=='"')

{

do

{

ch=fgetc(f1);

printf("%c",ch);

}while(ch!='"');

printf("\b is an argument\n");

i=-1;

break;

}

str[i]='\0';

keyw(str);

}

}

if(i!=-1)

{

str[i]=ch;

i++;

}

else

i=0;

}

printf("Keywords: %d\nIdentifiers: %d\nOperators: %d\nNumbers: %d\nPunctuations: %d\n",kw,id,op,num,pun);

}

void keyw(char \*p)

{

int k,flag=0,flag1=0;

for(k=0;k<=31;k++)

{

if(strcmp(keys[k],p)==0)

{

printf("%s is a keyword\n",p);

kw++;

flag=1;

break;

}

}

for(k=0;k<=3;k++)

{

if(punc[k]==p[0])

{

flag1=1;

break;

}

}

if(flag==0)

{

if(isdigit(p[0]))

{

printf("%s is a number\n",p);

num++;

}

else

{

if(p[0]!='\0'&&flag1==0)

{

printf("%s is an identifier\n",p);

id++;

}

}

}

i=-1;

}

