**PRATICAL - 1**

**AIM : Write a program to count number of characters, words and lines from a given input file.**

**Program : -**

#include<iostream>

#include<fstream>

using namespace std;

int main()

{

fstream f("abcd.odt",ios::in);

int c=0,w=0,l=0,s=0,cap=0,sp=0;

char ch;

f.get(ch);

if(ch==' ')

{

cout<<" FILE DOES NOT HAVE ANYTHING";

w=0;

l=0;

}

else

{

while(!f.eof())

{

cout<<ch;

if(ch>='a' && ch<='z')

{

s++;

c++;

}

if(ch>='A' && ch<='Z')

{

cap++;

c++;

}

if(ch=='!' || ch=='@' || ch=='$' || ch=='&')

{

sp++;

c++;

}

if(ch=='\n')

{

l++;

}

if(ch==' ' || ch=='\n')

{

++w;

}

f.get(ch);

}

}

cout<<"chateracters are = "<<c<<endl;

cout<<"NO of lines are = "<<l<<endl;

cout<<"total words are = "<<w<<endl;

cout<<"small characters="<<s<<endl;

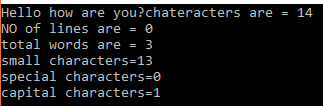
cout<<"special characters="<<sp<<endl;

cout<<"capital characters="<<cap<<endl;

return 0;

}

**OUTPUT :-**



**PRATICAL – 2**

**AIM: Write a program to strip comments from an input “C” program file and store in output file.**

**PROGRAM :-**

#include<iostream>

#include<cstdlib>

#include<fstream>

using namespace std;

int main()

{

char ch;

int count=0;

fstream f1("file.cpp",ios::in|ios::out),f2("file1.cpp",ios::in|ios::out|ios::app);

while(f1)

{

f1.get(ch);

if(ch=='/')

{

f1.get(ch);

if(ch=='/')

{

while(ch!='\n')

{

f1.get(ch);

cout<<ch;

}

}

else

{

f1.get(ch);

while(ch!='/')

{

f1.get(ch);

cout<<ch;

}

}

}

else

{

f2.put(ch);

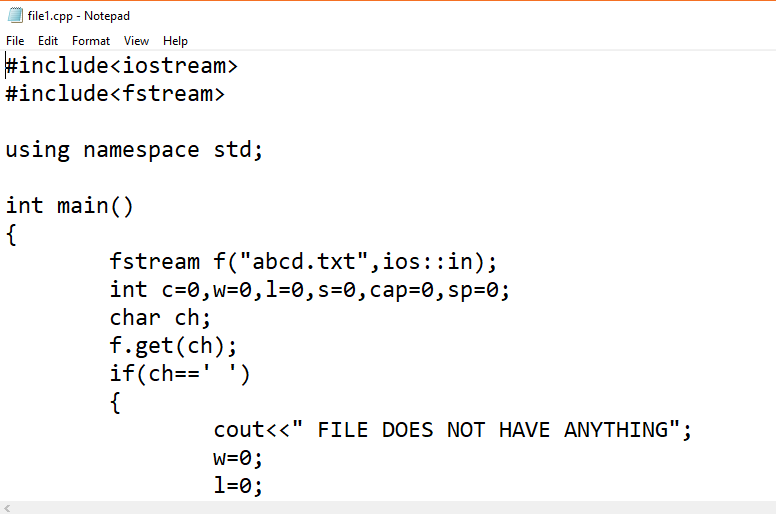
}

}

return 0;

}

**OUTPUT :-**



**PRATICAL – 3**

**AIM :- Write a program to study the use of strtok().**

**PROGRAM :-**

#include <iostream>

#include <cstring>

using namespace std;

int main ()

{

char str[] ="- This, a sample string.";

char \* tok;

cout<<"Splitting string "<<str<<" into tokens:\n"<<endl;

tok = strtok (str," ,.-");

while (tok != NULL)

{

cout<<tok<<endl;

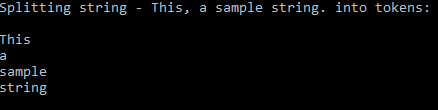
tok = strtok (NULL, " ,.-");

}

return 0;

}

**OUTPUT :-**



**PRATICAL – 4**

**AIM : Write a program to implement Lexical Analyzer.**

**PROGRAM :-**

//Write a program to implement Lexical Analyzer.

#include<iostream>

#include<fstream>

#include<stdlib.h>

#include<string.h>

#include<ctype.h>

using namespace std;

int Keywords(char buf[])

{

char key[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"};

int i, flag = 0;

for(i = 0; i < 32; ++i)

{

if(strcmp(key[i], buf) == 0)

{

flag = 1;

break;

}

}

return flag;

}

int main()

{

char ch, buf[15], operators[] = "+-\*/=%";

ifstream fin("program.txt");

int i,j=0;

cout<<"\nIn Your File Operators Are : ";

while(!fin.eof())

{

ch = fin.get();

for(i = 0; i < 6; ++i)

{

if(ch == operators[i])

cout<<ch<<" ";

}

}

fin.close();

cout<<endl;

fin.open("program.txt");

cout<<"\n";

while(!fin.eof())

{

ch = fin.get();

if(isalnum(ch))

{

buf[j++] = ch;

}

if((ch == ' ' || ch == '\n') && (j != 0))

{

buf[j] = '\0';

j = 0;

if(Keywords(buf) == 1)

cout<<buf<<" is Keyword.\n";

else

cout<<buf<<" is Indentifier.\n";

}

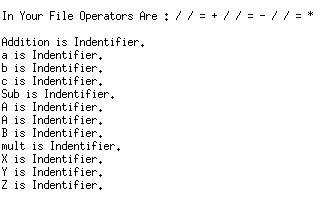
}

fin.close();

return 0;

}

**OUTPUT : -**



**PRATICAL – 6**

**AIM : Write the following programs with Lex Utility Tool.**

1. **Write a LEX program to check whether input symbol is number or character.**

**PROGRAM :-**

%{

#include<stdio.h>

%}

%%

[0-9] printf("It is a number");

[a-zA-Z] printf("It is a character");

. printf("Other");

%%

int yywrap()

{

return 1;

}

int main()

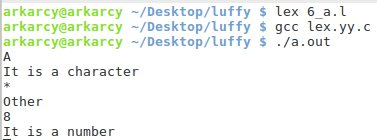
{

yylex();

return 0;

}

**OUTPUT :-**



1. **Write a LEX program to check whether input is multi-digit number or string.**

**PROGRAM :-**

%{

#include<stdio.h>

%}

%%

[0-9][0-9]+ printf("It is a Multi digit number");

[a-zA-Z]+ printf("String");

%%

int yywrap()

{

return 1;

}

int main()

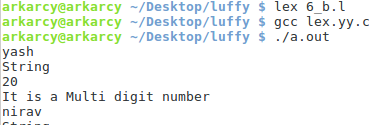
{

yylex();

return 0;

}

**OUTPUT :-**



1. **Write a LEX program to check whether inputted character is vowel or not.**

**PROGRAM :-**

%{

#include<stdio.h>

%}

%%

[aeiouAEIOU]+ printf("Vowel");

. printf("Not Vowel");

%%

int yywrap()

{

return 1;

}

int main()

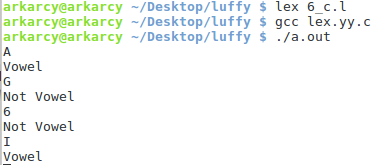
{

yylex();

return 0;

}

**OUTPUT :-**



1. **Write a LEX program to check whether a word starts with vowel or not.**

**PROGRAM :-**

%{

#include<stdio.h>

%}

%%

[aeiouAEIOU][a-zA-Z]+ printf("WORD START WITH VOWEL");

. printf("WORD DOES NOT START WITH VOWEL");

%%

int yywrap()

{

return 1;

}

int main()

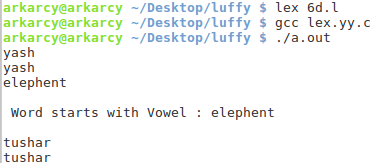
{

yylex();

return 0;

}

**OUTPUT :-**



1. **Write a Lex program to count the number of characters, words and lines from a given input file**.

**PROGRAM :-**

%{

#include<stdio.h>

int wc=0,ch=0,lc=0;

%}

%%

[a-zA-Z]\* {

wc++;

ch+=yyleng;

}

\n lc++;

. ch++;

%%

int yywrap()

{

return 1;

}

int main()

{

yyin=fopen("inp.txt","r");

if(yyin==NULL)

printf("FILE IS NOT THERE \n");

yyout=fopen("out.txt","w");

yylex();

printf("\n WORDS ARE %d",wc);

printf("\n CHARACTERS ARE %d",ch);

printf("\n LINES ARE %d \n",lc);

return 1;

}

**OUTPUT :-**

