```
/**************************************************************

Name  :     Shubham S. Tembhurkar
PRN   :     1641060
Batch :     B1
Class :     L.Y. B.Tech.
Sub   :     CCL
Aim   :     Design a lexical analyzer for given language and the
Lexical analyzer should ignore redundant spaces, tabs and new lines.
**************************************************************/

#include<stdio.h>     //header file
#include<string.h>    //header file
int main()      //main function
{
    int c=1,i,j=0;        //declaring variables
    char file_name[20],s[20];
    scanf("%s",file_name);
    FILE *fp;
    fp=fopen(file_name,"r");
    if(fp == NULL)        //if condition
    {
        printf("Failed to open the file");
    }
    else            //else condition
    {
        do{
            fscanf(fp,"%s",s);
            if(s[0]=='#' && s[7]=='e')
            {
                printf("%s is a Header file\n",s);
            }
            else
            if(!strcmp(s,"+") || !strcmp(s,"-") || !strcmp(s,"*")
|| !strcmp(s,"/") || !strcmp(s,">") || !strcmp(s,"<") ||
               !strcmp(s,"=") || !strcmp(s,"<=") ||
!strcmp(s,">=") || !strcmp(s,"!") || !strcmp(s,"!=") ||
!strcmp(s,"==") ||
               !strcmp(s,"++") || !strcmp(s,"--") ||
!strcmp(s,"&") || !strcmp(s,"|") || !strcmp(s,"||") || !strcmp(s,"&&")
||
               !strcmp(s,"%"))
            {
                printf("%s is an Operator\n",s);
            }
            else
            if(!strcmp(s,"auto") ||!strcmp(s,"break")
||!strcmp(s,"case") || !strcmp(s,"char") || !strcmp(s,"const") ||
               !strcmp(s,"continue") || !strcmp(s,"default") ||
!strcmp(s,"do") || !strcmp(s,"double") || !strcmp(s,"else") ||
               !strcmp(s,"enum") || !strcmp(s,"extern") ||
!strcmp(s,"float") || !strcmp(s,"for") || !strcmp(s,"goto") ||
               !strcmp(s,"if") || !strcmp(s,"int") ||
!strcmp(s,"long") || !strcmp(s,"register") || !strcmp(s,"return") ||
               !strcmp(s,"short") || !strcmp(s,"signed") ||
!strcmp(s,"sizeof") || !strcmp(s,"static") || !strcmp(s,"struct") ||
               !strcmp(s,"switch") || !strcmp(s,"typedef") ||
!strcmp(s,"union") || !strcmp(s,"unsigned") || !strcmp(s,"void") ||
               !strcmp(s,"volatile") || !strcmp(s,"while"))
            {
                printf("%s is a Keyword\n",s);
            }
            else
            if(!strcmp(s,"\""))
            {
                fscanf(fp,"%s",s);
                while(strcmp(s,"\""))
                {
                     printf("%s ",s);
                     fscanf(fp,"%s",s);
                }
                printf("is an Argument\n");
            }
            else
            if(!strcmp(s,"scanf") || !strcmp(s,"printf") ||
!strcmp(s,"main"))
            {
                printf("%s is an Identifier\n",s);
            }
            else
            if(!strcmp(s,",") || !strcmp(s,";") || !strcmp(s,"{")
|| !strcmp(s,"}") || !strcmp(s,"(") || !strcmp(s,")"))
            {
                continue;
            }
            else
            {
                printf("%s is an Identifier\n",s);
            }
      }while(c != EOF);
    }
    fclose(fp);
    return 0;
}

/*******************OUTPUT*********************
abc.c
#include<stdio.h> is a Header file
#include<stdlib.h> is a Header file
int is a Keyword
main is an Identifier
int is a Keyword
```

```
a is an Identifier
b is an Identifier
c is an Identifier
ch is an Identifier
printf is an Identifier
\n ENTER numbers as an argument is an Argument
scanf is an Identifier
%d%d is an Argument
& is an Operator
a is an Identifier
& is an Operator
b is an Identifier
printf is an Identifier
\nMENU\n1.ADD\n2.Sub\n3.MUL is an Argument
scanf is an Identifier
%d is an Argument
& is an Operator
ch is an Identifier
c is an Identifier
= is an Operator
a is an Identifier
+ is an Operator
b is an Identifier
if is a Keyword
ch is an Identifier
== is an Operator
0 is an Identifier
printf is an Identifier
bye is an Argument

*******************************************************/
```

INPUT FILE :

abc.c

```
#include<stdio.h>
#include<stdlib.h>

int main ( )
{
      int a , b , c , ch ;
      printf ( " \n ENTER numbers as an argument " ) ;
      scanf ( " %d%d " , & a , & b ) ;
      printf ( " \nMENU\n1.ADD\n2.Sub\n3.MUL " ) ;
      scanf ( " %d " , & ch ) ;
      c = a + b ;
      if ( ch == 0 )
      {
           printf ( " bye " ) ;
      }
}
```