Practical 3

Definition: Write a program to implement lexical analyzer.

Code:

```
#include <string.h>
#include <stdio.h>
#include <conio.h>
void main(){
	FILE *f1;
	char ch,str[300][30],str1[300][30],sym[300];
	int m=0,j=0,i=0,i1=0,j1=0,flag=0,p=0,q=0;
	clrscr();
	f1=fopen("sample.c","r");
	while((ch=getc(f1))!=EOF){
			if(isalnum(ch) || ch=='_'){
			str[i][j++]=ch;
			flag=1;
		}
		else{
			if(!isspace(ch)){
				sym[m++]=ch;
			}
			if(flag==1){
				str[i][j]='\0';
				i++;
				j=0;
				flag=0;
			}
			if(ch=='"'){
				while((ch=getc(f1))!='"'){
					str1[i1][j1++]=ch;
			}
			 str1[i1][j1]='\0';
			 i1++;
			 j1=0;
		}
  }
  }
  fclose(f1);
	for(p=0;p<i;p++){
```

```
                printf("%s\t",str[p]);
        }
        printf("\n---------------------------------\n");
        for(p=0;p<m;p++){
                printf("%c\t",sym[p]);
        }
                printf("\n---------------------------------\n");
        for(p=0;p<i1;p++){
                printf("%s\t",str1[p]);
        }
        getch();
}
/*
Output
include ctype h      include stdio h      include conio h      include
string h      void    main    char    s1      20      printf  gets    s1
puts    s1      getch
--------------------------------
#       <       .       >       #       <       .       >       #       <
.       >       #       <       .       >       (       )       {       [
]       ;       (       "       )       ;       (       )       ;       (
)       ;       (       )       ;       }
--------------------------------
Enter a string:
*/
```