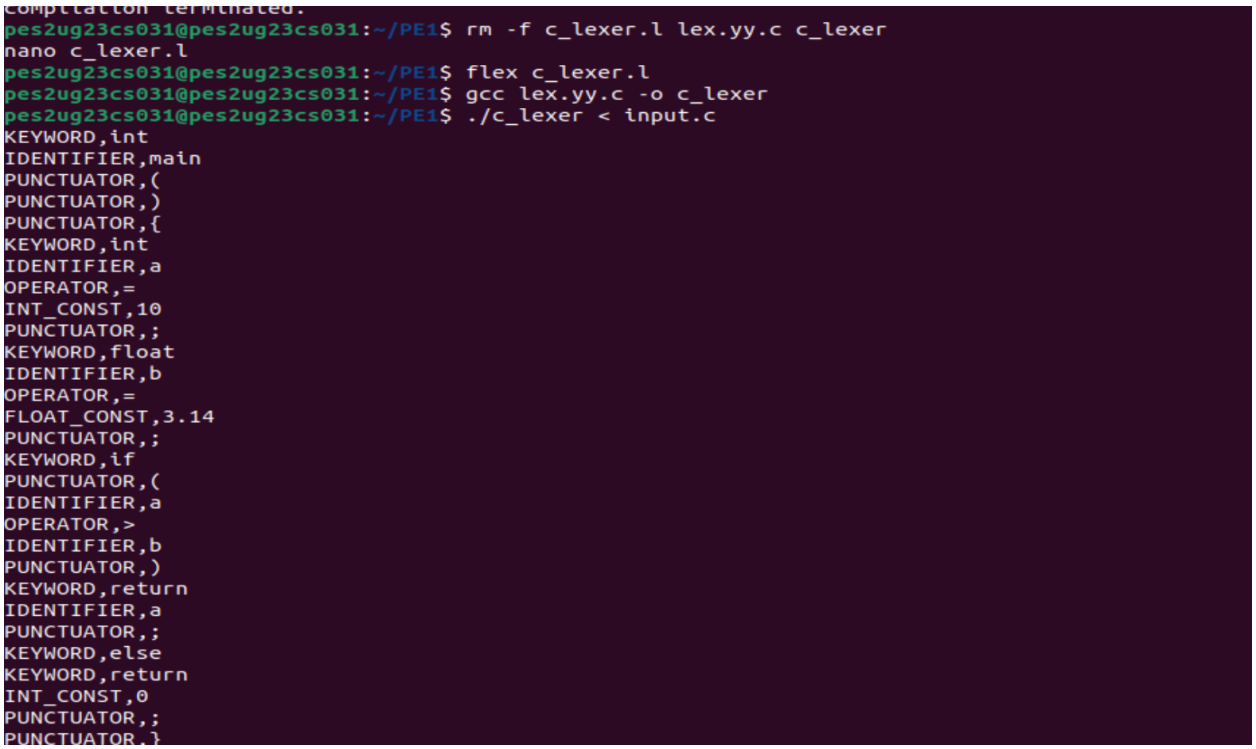
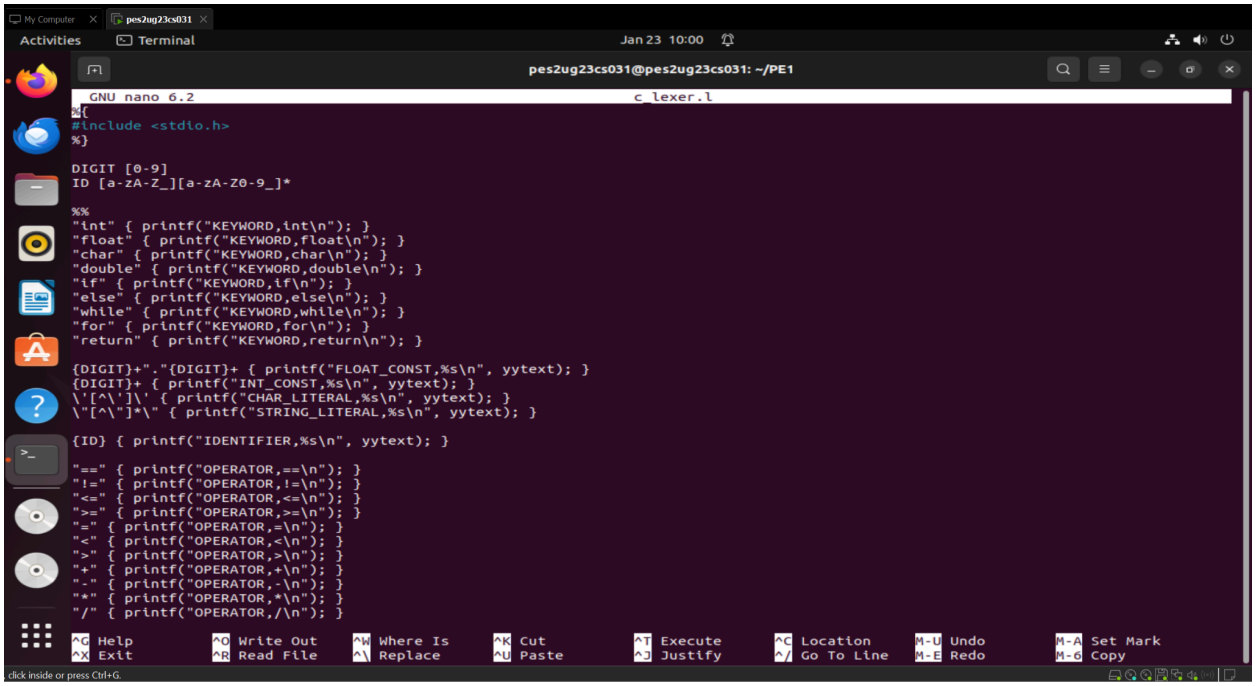
COMPILER DESIGN

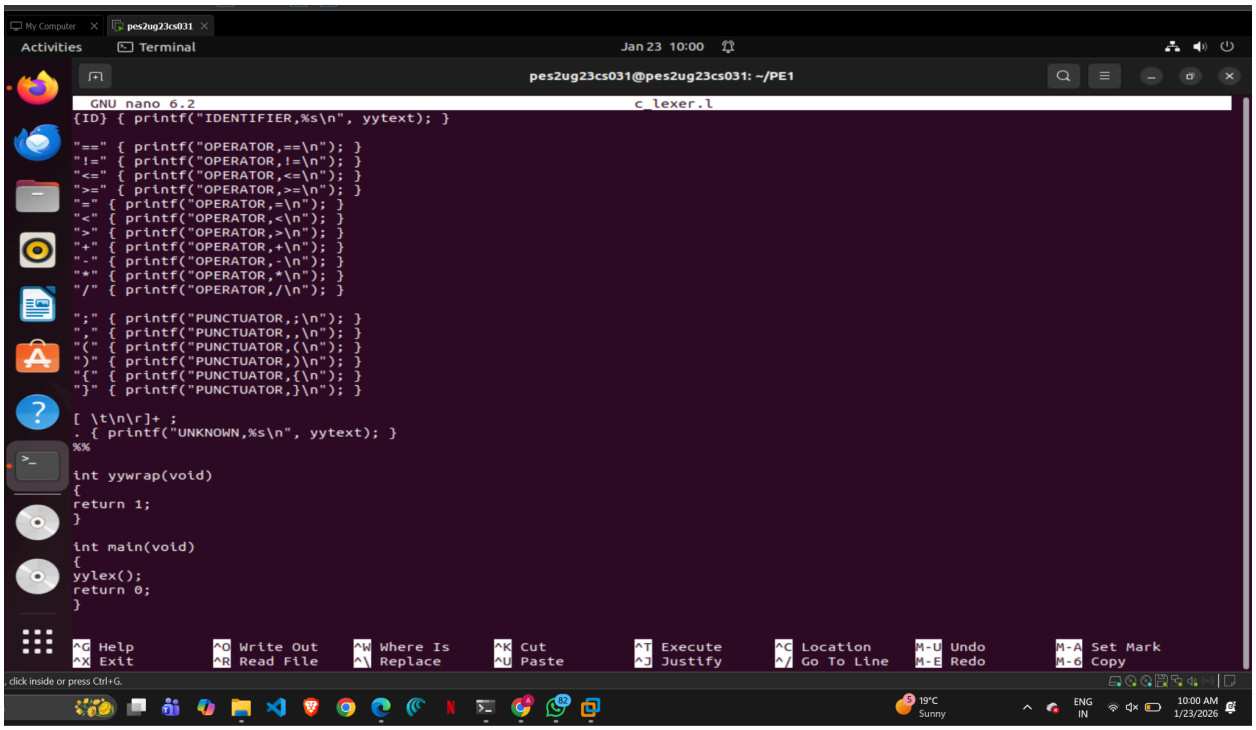
PE 1
NAME:ADITYA D RAO
SRN:PES2UG23CS031
DATE:23RD JAN 2026

GITHUB LINK
https://github.com/AdiXgit/compiler-design/tree/main/PE1



C_LEXER.L


pes2ug23cs031
Activities
Terminal
Jan 23 10:00
pes2ug23cs031@pes2ug23cs031: ~/PE1
GNU nano 6.2
c_lexer.l
%{
#include <stdio.h>
%}
DIGIT [0-9]
ID [a-zA-Z_][a-zA-Z0-9_]*
%%
"int" { printf("KEYWORD,int\n"); }
"float" { printf("KEYWORD,float\n"); }
"char" { printf("KEYWORD,char\n"); }
"double" { printf("KEYWORD,double\n"); }
"if" { printf("KEYWORD,if\n"); }
"else" { printf("KEYWORD,else\n"); }
"while" { printf("KEYWORD,while\n"); }
"for" { printf("KEYWORD,for\n"); }
"return" { printf("KEYWORD,return\n"); }
{DIGIT}+"."{DIGIT}+ { printf("FLOAT_CONST,%s\n", yytext); }
{DIGIT}+ { printf("INT_CONST,%s\n", yytext); }
\'[^\']\' { printf("CHAR_LITERAL,%s\n", yytext); }
\"[^\"]*\" { printf("STRING_LITERAL,%s\n", yytext); }
{ID} { printf("IDENTIFIER,%s\n", yytext); }
"==" { printf("OPERATOR,==\n"); }
"!=" { printf("OPERATOR,!=\n"); }
"<=" { printf("OPERATOR,<=\n"); }
">=" { printf("OPERATOR,>=\n"); }
"=" { printf("OPERATOR,=\n"); }
"<" { printf("OPERATOR,<\n"); }
">" { printf("OPERATOR,>\n"); }
"+" { printf("OPERATOR,+\n"); }
"-" { printf("OPERATOR,-\n"); }
"*" { printf("OPERATOR,*\n"); }
"/" { printf("OPERATOR,/\n"); }
^G Help
^X Exit
^O Write Out
^R Read File
^W Where Is
^\ Replace
^K Cut
^U Paste
^T Execute
^J Justify
^C Location
^/ Go To Line
M-U Undo
M-E Redo
M-A Set Mark
M-6 Copy
click inside or press Ctrl+G.


pes2ug23cs031
Activities
Terminal
Jan 23 10:00
pes2ug23cs031@pes2ug23cs031: ~/PE1
GNU nano 6.2
c_lexer.l
{ID} { printf("IDENTIFIER,%s\n", yytext); }
"==" { printf("OPERATOR,==\n"); }
"!=" { printf("OPERATOR,!=\n"); }
"<=" { printf("OPERATOR,<=\n"); }
">=" { printf("OPERATOR,>=\n"); }
"=" { printf("OPERATOR,=\n"); }
"<" { printf("OPERATOR,<\n"); }
">" { printf("OPERATOR,>\n"); }
"+" { printf("OPERATOR,+\n"); }
"-" { printf("OPERATOR,-\n"); }
"*" { printf("OPERATOR,*\n"); }
"/" { printf("OPERATOR,/\n"); }
";" { printf("PUNCTUATOR,;\n"); }
"," { printf("PUNCTUATOR,,\n"); }
"(" { printf("PUNCTUATOR,(\n"); }
")" { printf("PUNCTUATOR,)\n"); }
"{" { printf("PUNCTUATOR,{\n"); }
"}" { printf("PUNCTUATOR,}\n"); }
[\t\n\r]+ ;
. { printf("UNKNOWN,%s\n", yytext); }
%%
int yywrap(void)
{
return 1;
}
int main(void)
{
yylex();
return 0;
}
^G Help
^X Exit
^O Write Out
^R Read File
^W Where Is
^\ Replace
^K Cut
^U Paste
^T Execute
^J Justify
^C Location
^/ Go To Line
M-U Undo
M-E Redo
M-A Set Mark
M-6 Copy
click inside or press Ctrl+G.
19°C
Sunny
ENG
IN
10:00 AM
1/23/2026

```
pes2ug23cs031@pes2ug23cs031:~/PE1$ nano input.c
pes2ug23cs031@pes2ug23cs031:~/PE1$ ./c_lexer < input.c > tokens.csv
pes2ug23cs031@pes2ug23cs031:~/PE1$ sed -i '1iTOKEN_TYPE,LEXEME' tokens.csv
pes2ug23cs031@pes2ug23cs031:~/PE1$ cat tokens.csv
TOKEN_TYPE,LEXEME
KEYWORD,int
IDENTIFIER,main
PUNCTUATOR,(
PUNCTUATOR,)
PUNCTUATOR,{
KEYWORD,int
IDENTIFIER,a
OPERATOR,=
INT_CONST,10
PUNCTUATOR,;
KEYWORD,float
IDENTIFIER,b
OPERATOR,=
FLOAT_CONST,3.14
PUNCTUATOR,;
KEYWORD,if
PUNCTUATOR,(
IDENTIFIER,a
OPERATOR,>
IDENTIFIER,b
PUNCTUATOR,)
KEYWORD,return
IDENTIFIER,a
PUNCTUATOR,;
KEYWORD,else
KEYWORD,return
INT_CONST,0
PUNCTUATOR,;
PUNCTUATOR,}
pes2ug23cs031@pes2ug23cs031:~/PE1$ nano c_lexer.l
```

THE INPUT FILE I USED