Compiler HW#1 Scanner

2013012148 이재일

# 1. Implementation of C-Scanner using C-code

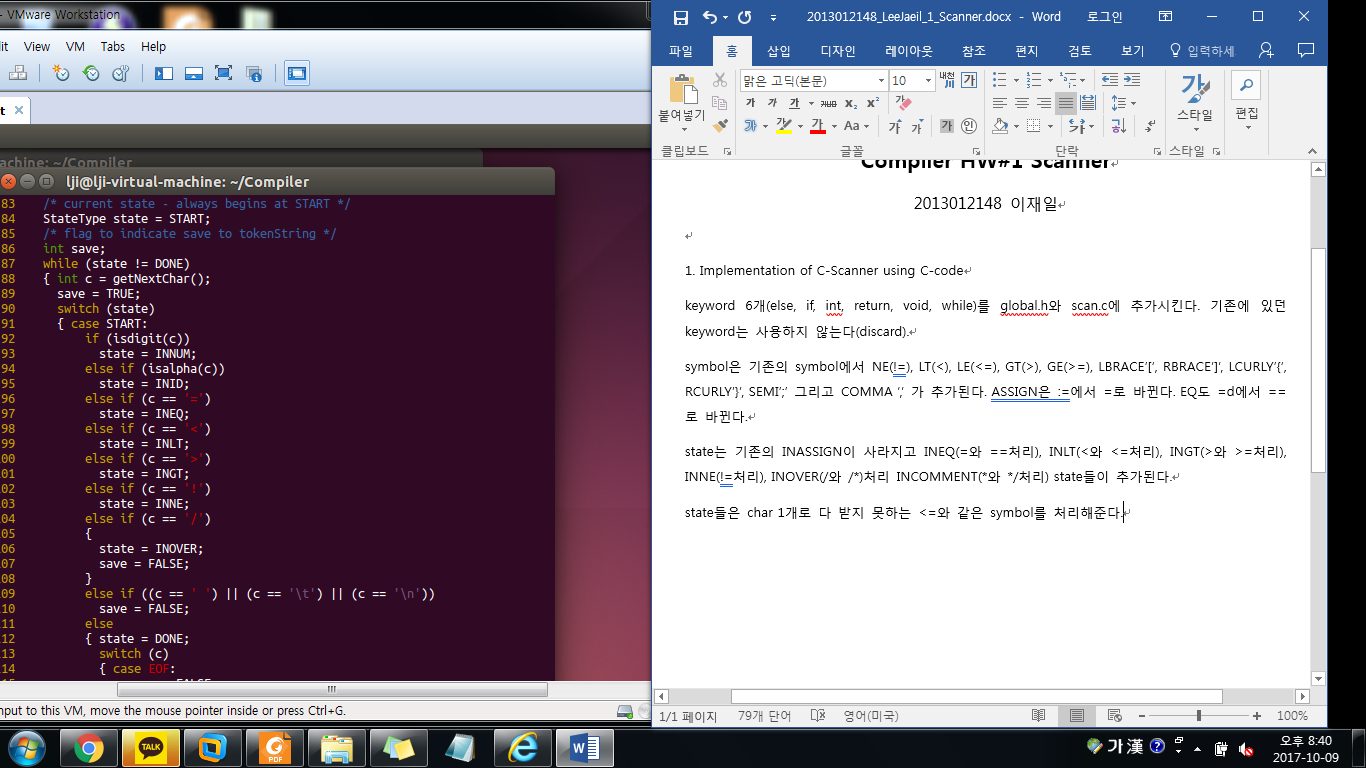
keyword 6개(else, if, int, return, void, while)를 global.h와 scan.c에 추가시킨다. 기존에 있던 keyword는 사용하지 않는다(discard).

symbol은 기존의 symbol에서 NE(!=), LT(<), LE(<=), GT(>), GE(>=), LBRACE’[‘, RBRACE’]’, LCURLY’{‘, RCURLY’}’, SEMI’;’ 그리고 COMMA ’,’ 가 추가된다. ASSIGN은 :=에서 =로 바뀐다. EQ도 =d에서 ==로 바뀐다.

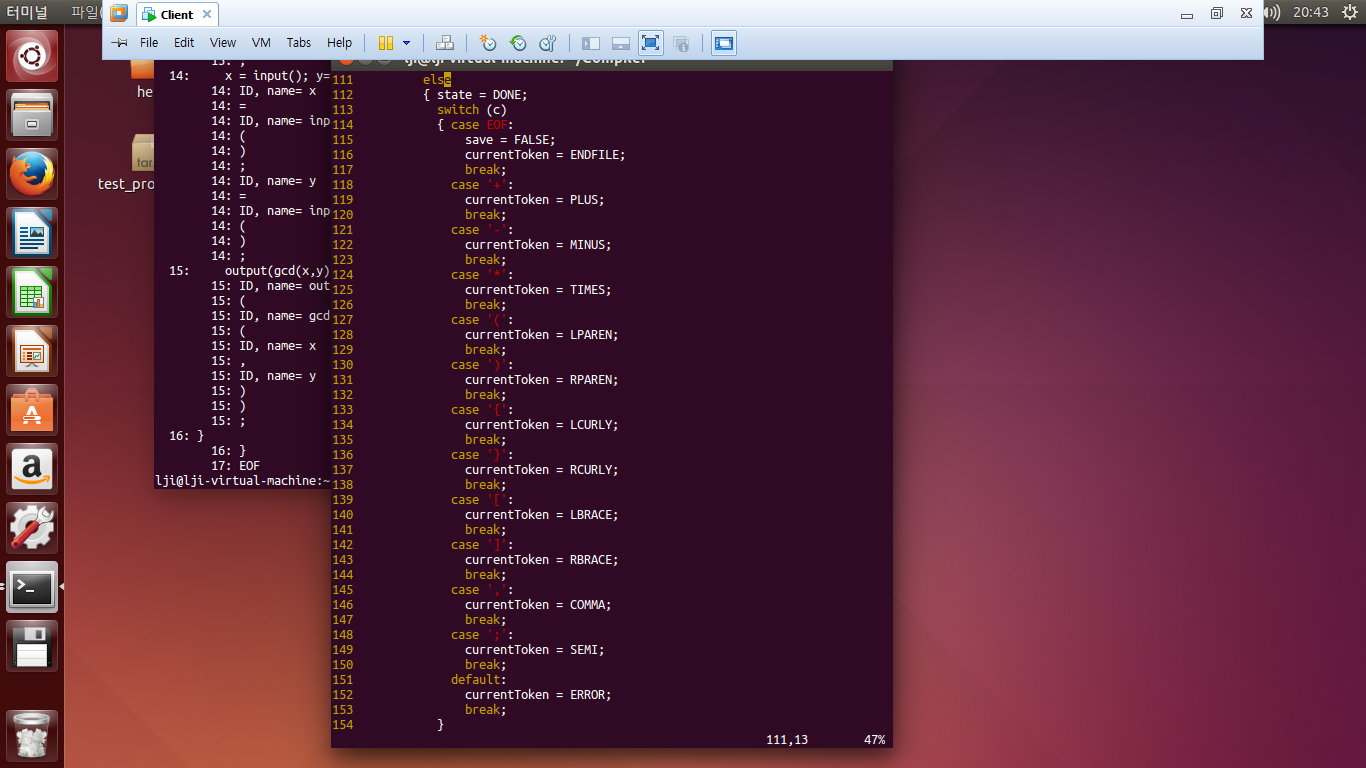
state는 기존의 INASSIGN이 사라지고 INEQ(=와 ==처리), INLT(<와 <=처리), INGT(>와 >=처리), INNE(!=처리), INOVER(/와 /\*)처리 INCOMMENT(\*와 \*/처리) state들이 추가된다.

state들은 char 1개로 다 받지 못하는 <=와 같은 symbol를 처리해준다.

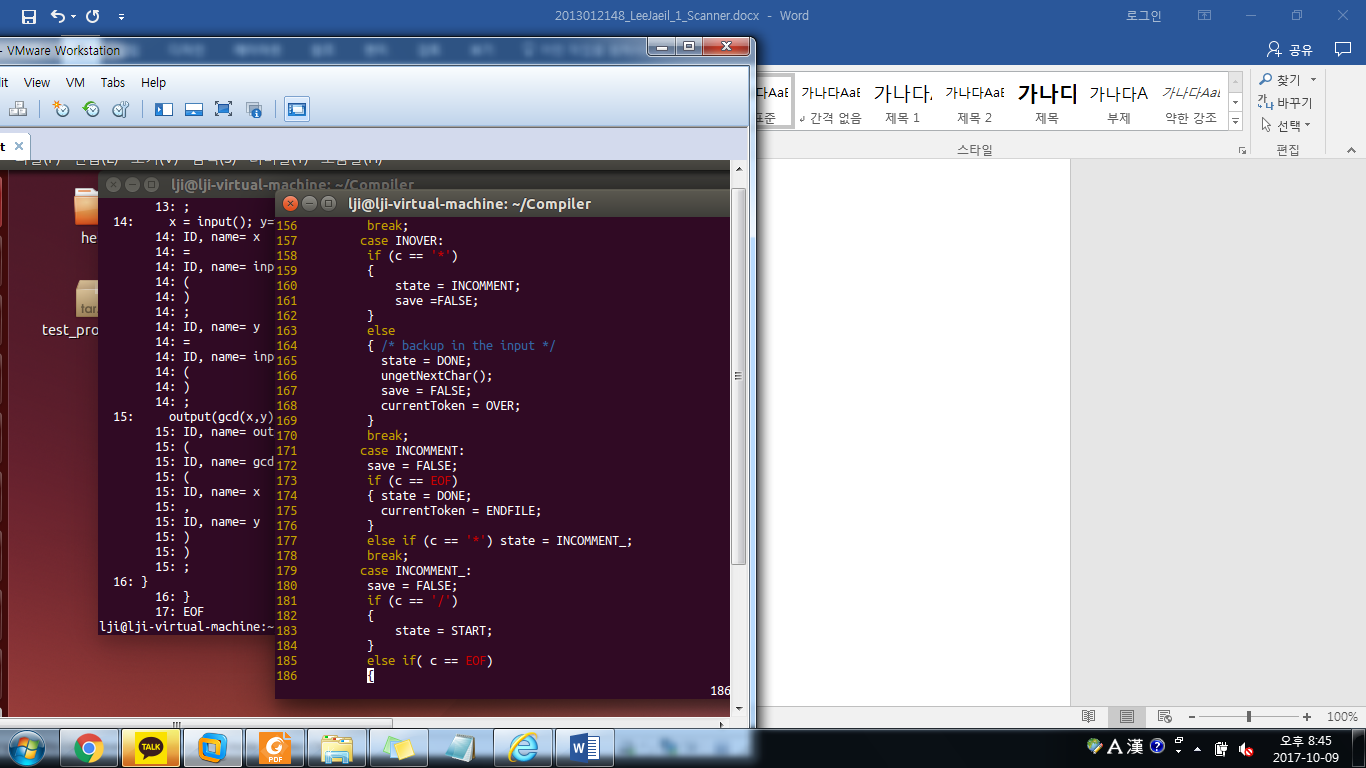
<받은 char에 따라서 해당 char에 맞는 state를 설정>



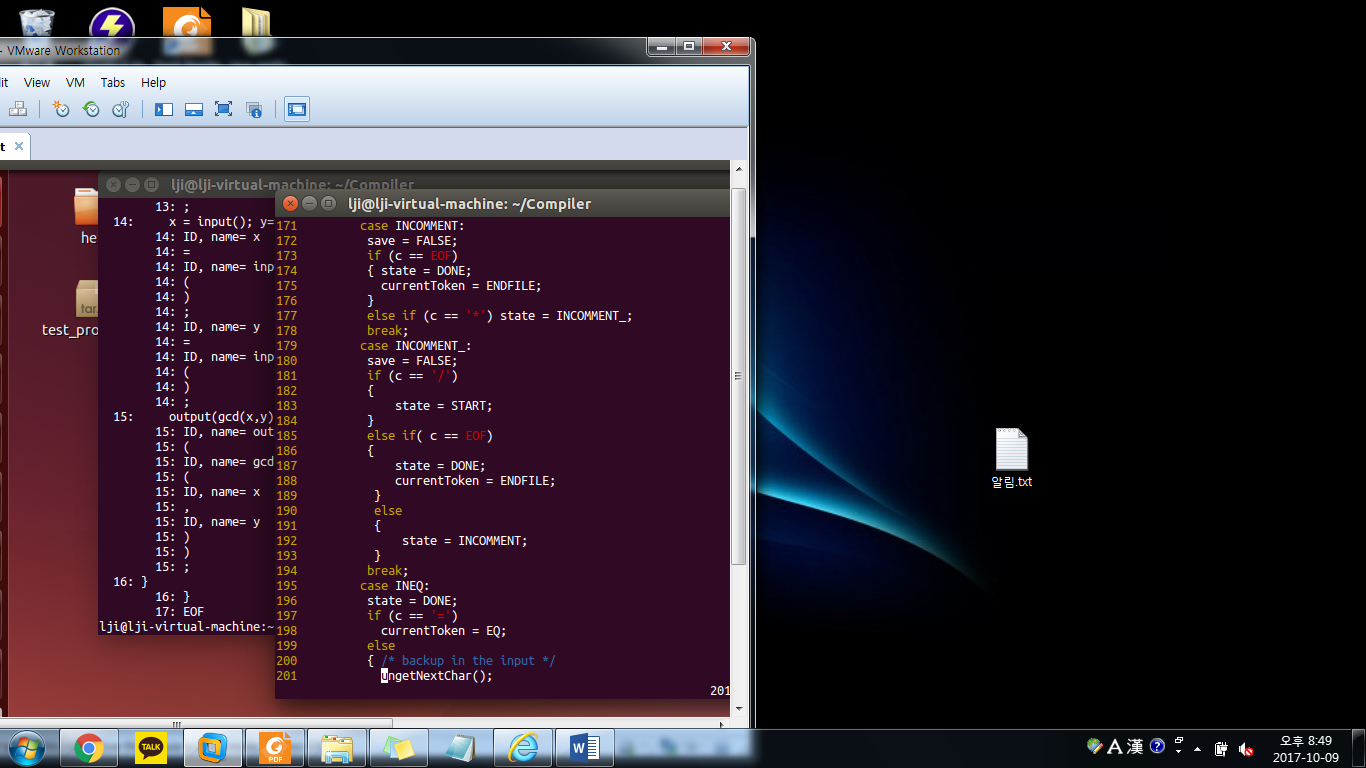
<char형으로 받을 수 있는 경우 && 위의 state를 거친 후 token이 확정되는 경우>



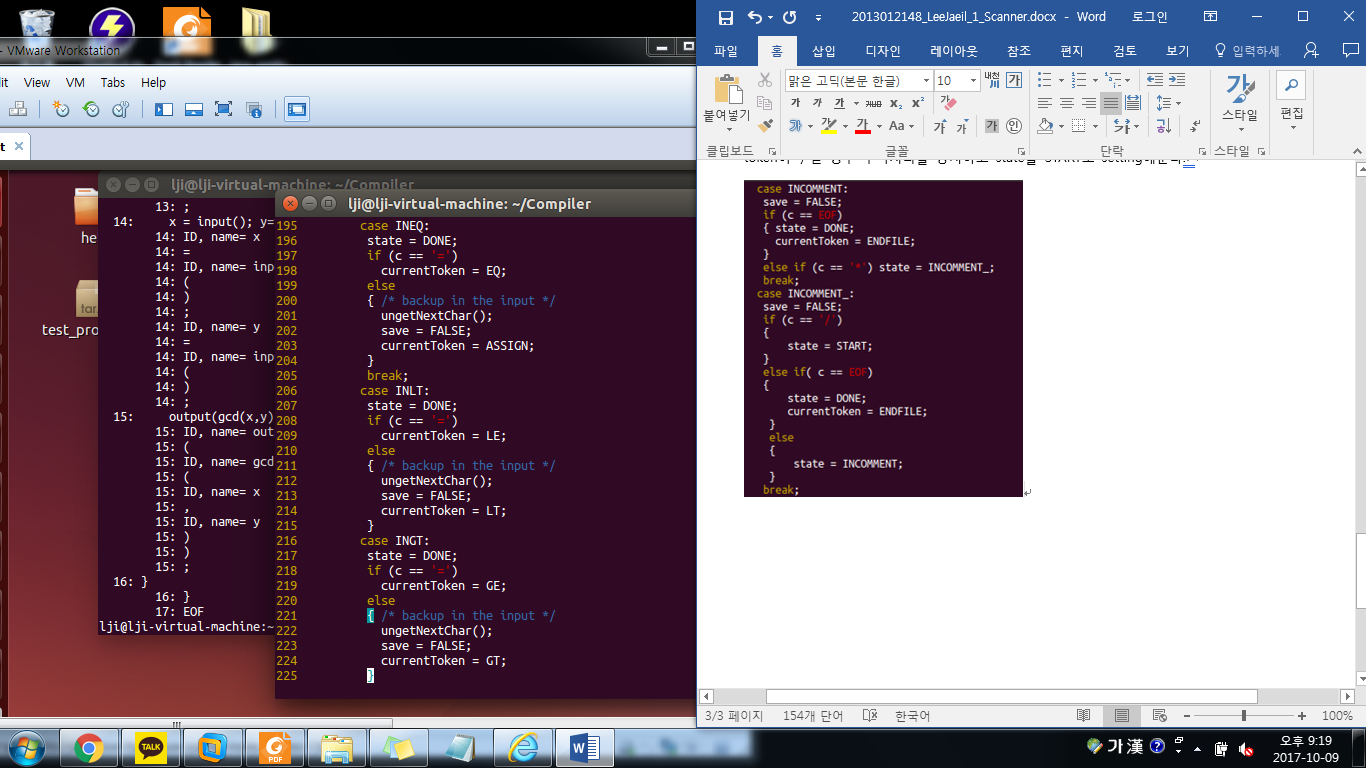
<state가 start인 상태에서 들어온 token이 ‘/’인 경우 state는 INOVER로 바뀐다. INOVER에서 다음 token이 \*인 경우 ‘\*/’를 만날때까지 주석처리한다. 다음 token이 \*아닐 경우 currenToken은 OVER.



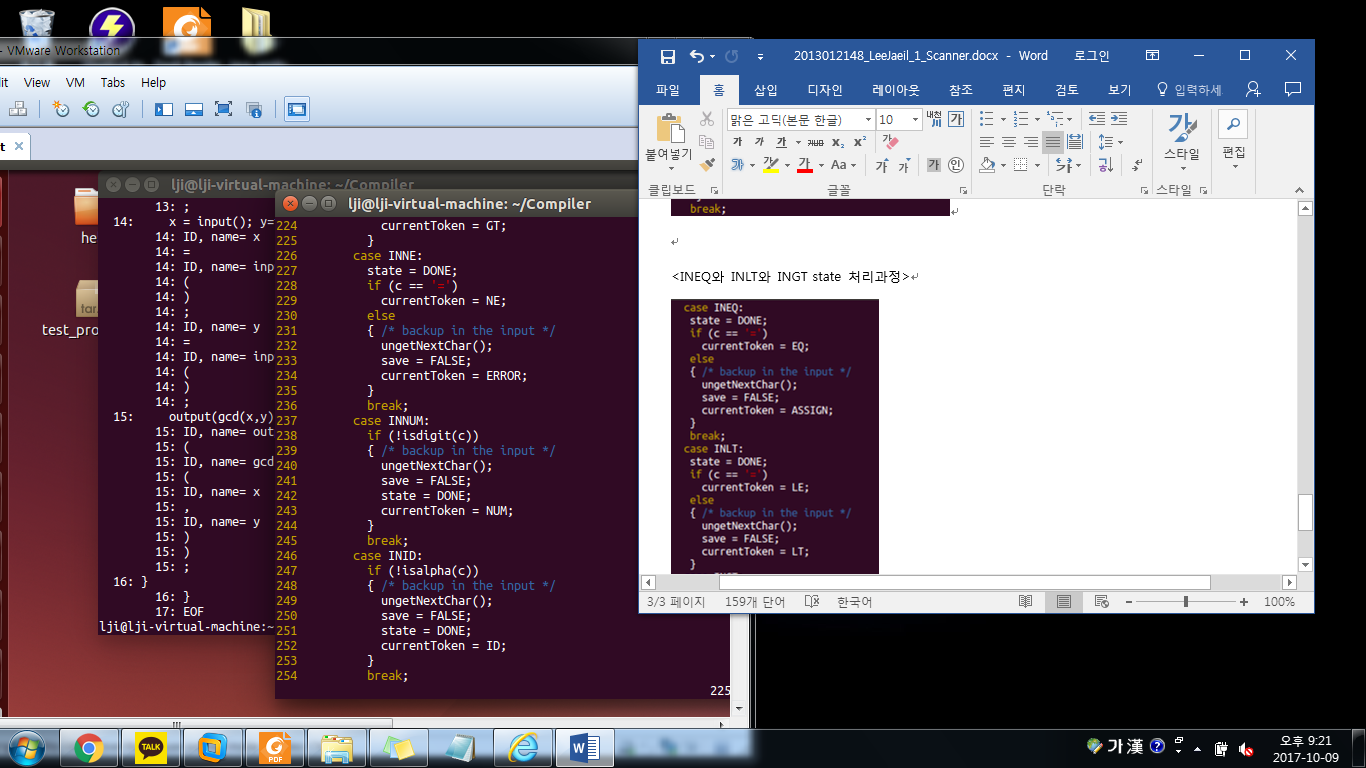
</이후에 \* token을 받을 경우 state는 INCOMMENT로 setting되며, token’\*/’을 만날때까지 주석처리한다. 만약 INCOMMENT state에서 \* token을 받을 경우 INCOMMENT\_ state로 setting하고 다음 token이 ‘/’일 경우 주석처리를 중지하고 state를 START로 setting해준다.>



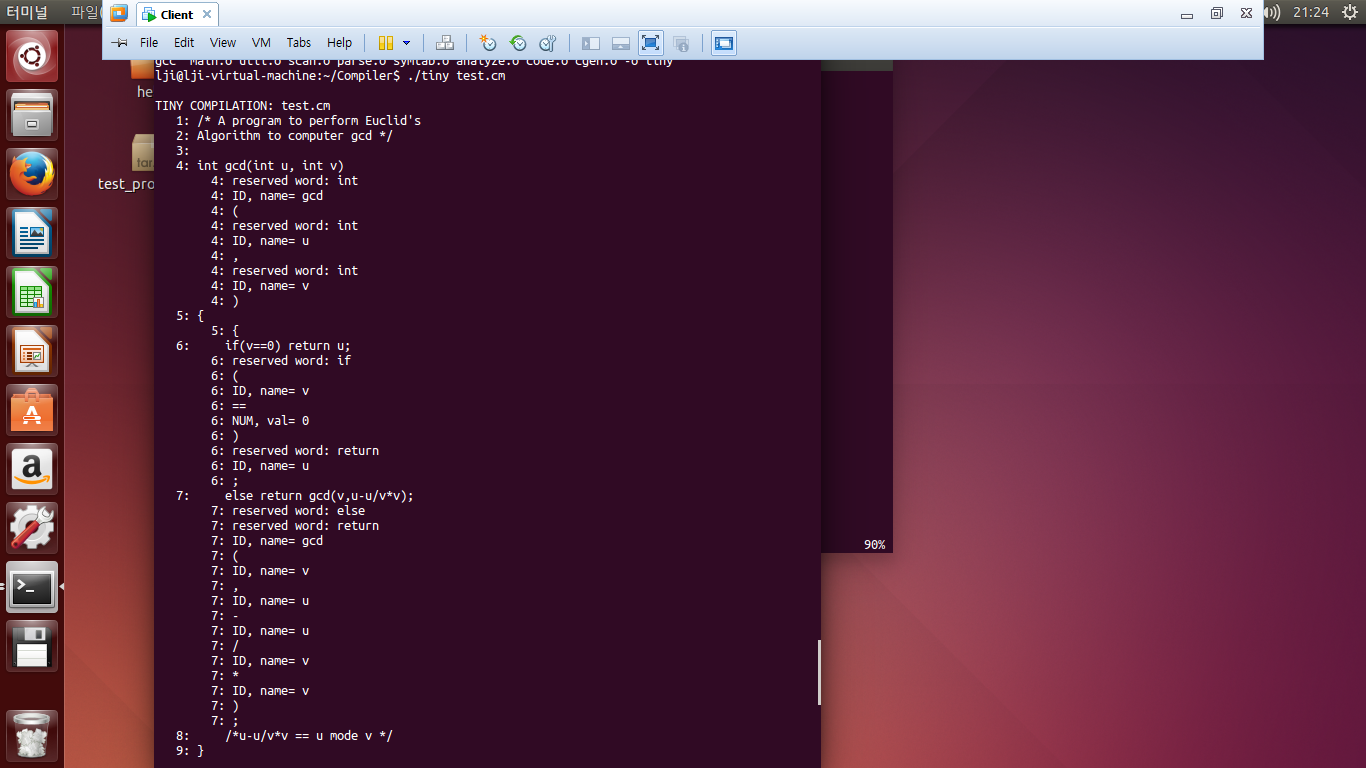
<INEQ와 INLT와 INGT state 처리과정>



<state INNE와 INNUM, INID 처리과정>

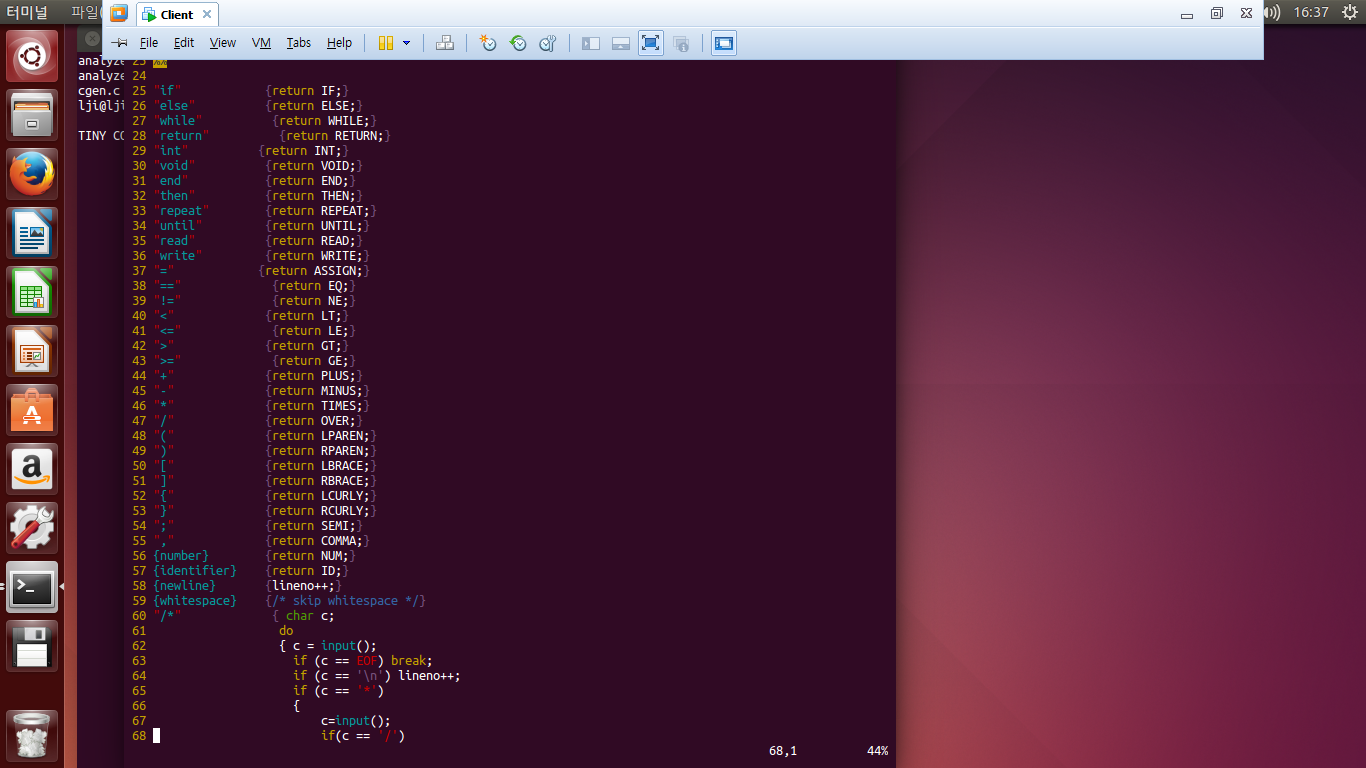


<Makefile로 exe만든 후 실행화면>

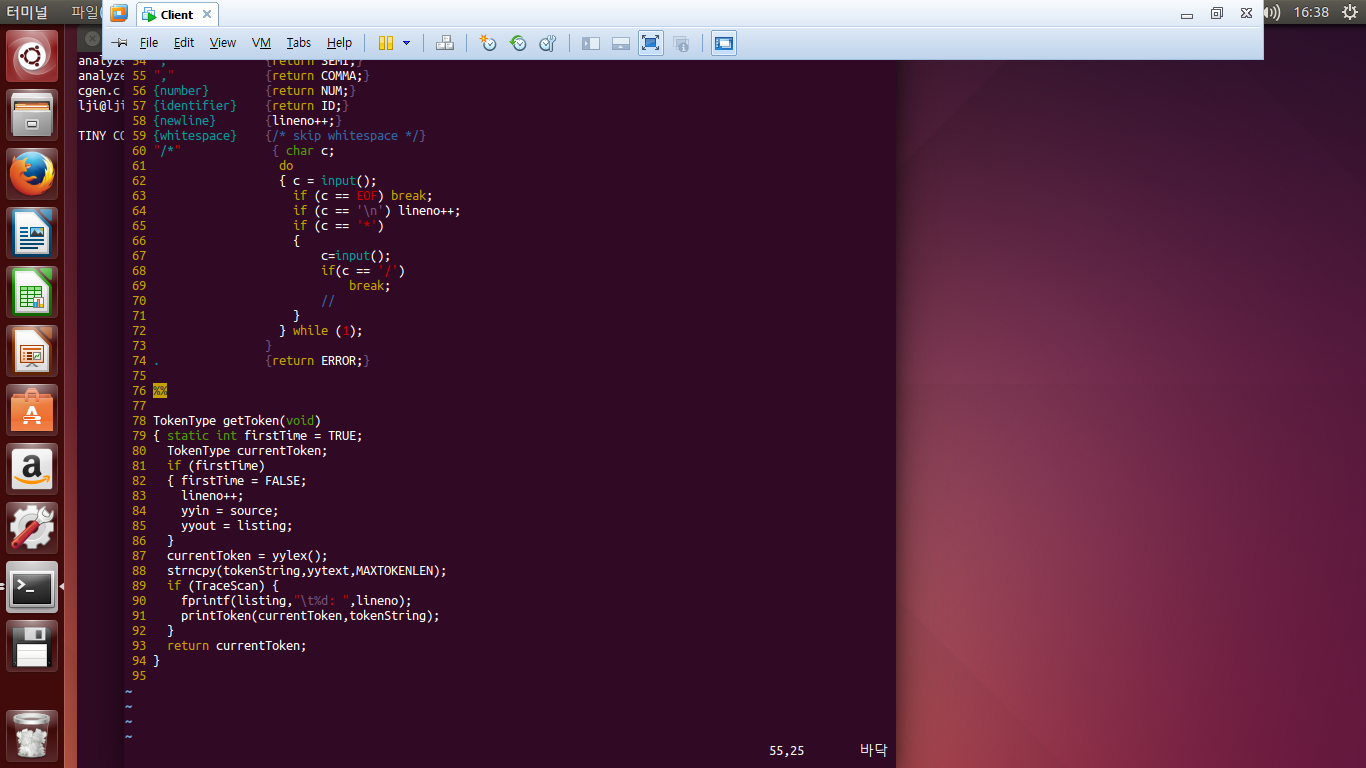


# 2. Implementation of C-Scanner using lex(flex) by Tiny.I modification

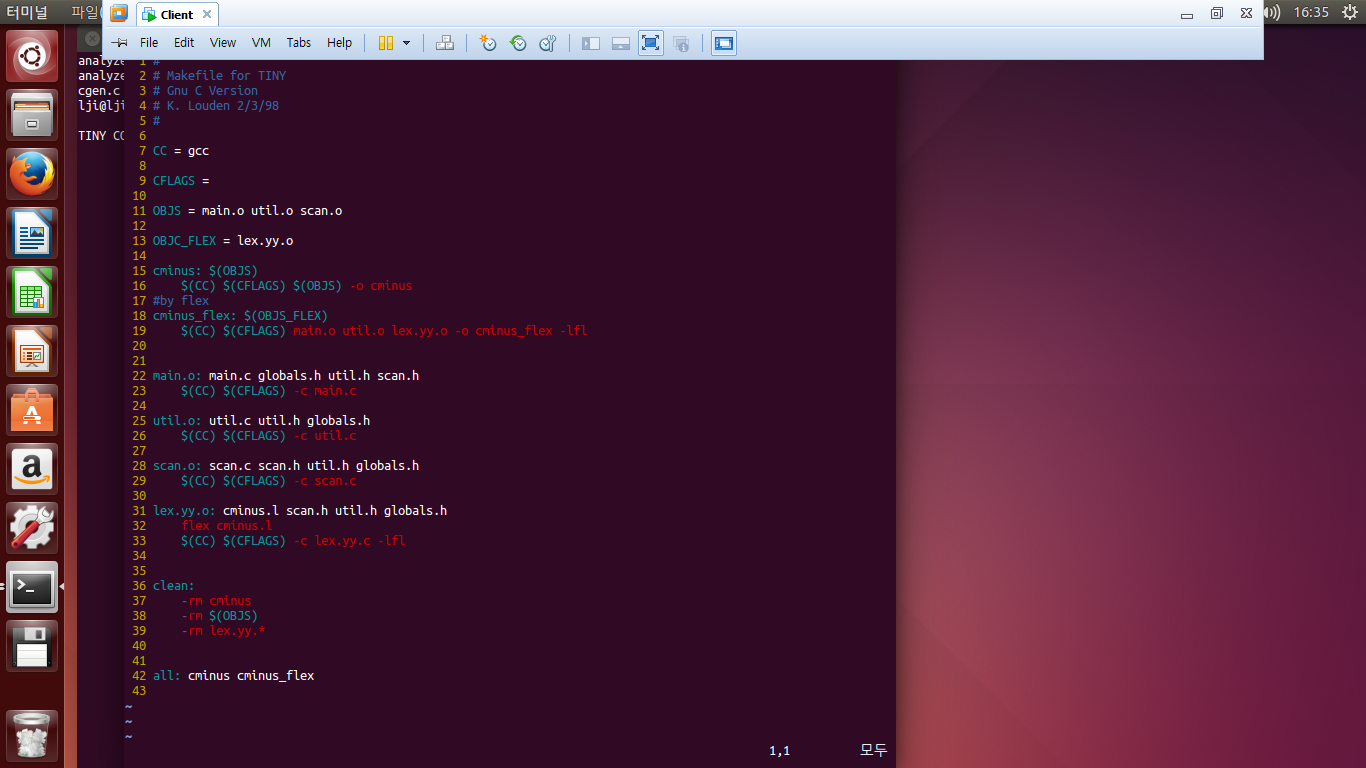
<cminus.l 구현>



<주석처리 구현>



<Makefile 변환>



<실행결과>

