

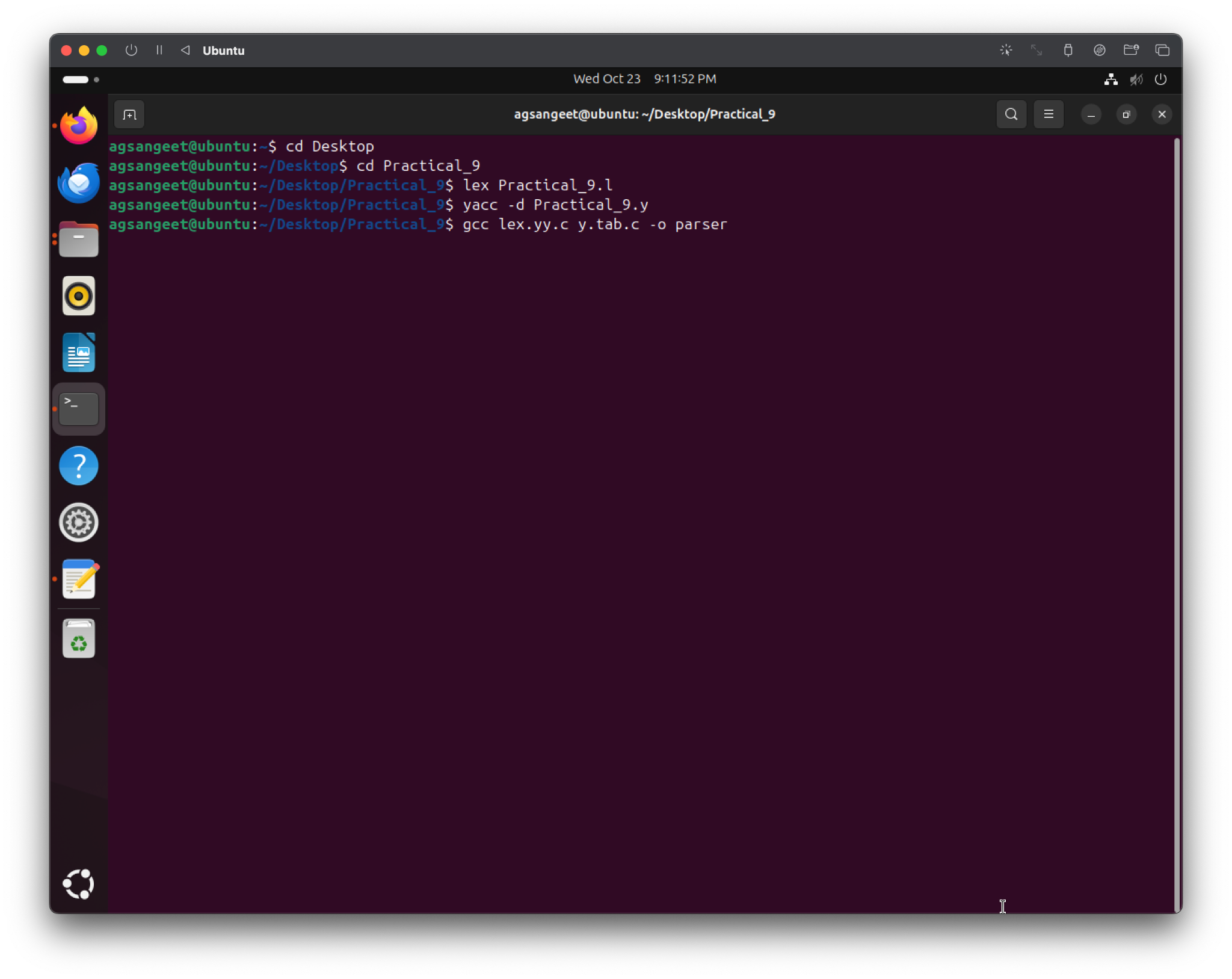
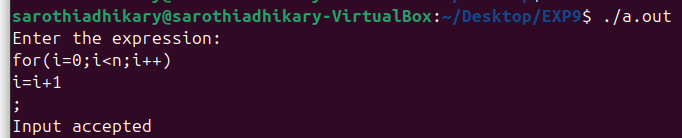
**Code for LEX:**

| alpha [A-Za-z] digit [0-9]  %% [\t \n] for return FOR; {digit}+ return NUM; {alpha}({alpha}|{digit})\* return ID; "<=" return LE; ">=" return GE; "==" return EQ; "!=" return NE; "||" return OR; "&&" return AND; . return yytext[0]; %% |
| --- |

**Code for YACC:**

| %{ #include <stdio.h> #include <stdlib.h>  void yyerror(char\*); int yylex(void);  %}  %token ID NUM FOR LE GE EQ NE OR AND %right "=" %left OR AND %left '>' '<' LE GE EQ NE %left '+' '-' %left '\*' '/' %right UMINUS %left '!'   %%   S : ST {printf("Input accepted\n"); exit(0);} ST : FOR '(' E ';' E2 ';' E ')' DEF  ; DEF : '{' BODY '}'  | E';'  | ST  |  ; BODY : BODY BODY  | E ';'   | ST  |   ;   E : ID '=' E  | E '+' E  | E '-' E  | E '\*' E  | E '/' E  | E '<' E  | E '>' E  | E LE E  | E GE E  | E EQ E  | E NE E  | E OR E  | E AND E  | E '+' '+'  | E '-' '-'  | ID   | NUM  ;    E2 : E'<'E  | E'>'E  | E LE E  | E GE E  | E EQ E  | E NE E  | E OR E  | E AND E  ;  %%  #include "lex.yy.c" int main() {  printf("Enter the expression:\n");  yyparse();  return 0; }  void yyerror(char\* errorText){  printf("%s",errorText); } |
| --- |

**Output:**

****