UE20CS353-Compiler Design

```
Ayush Singh
PES2UG20CS081
SEC-6B
```

```
%{

#define YYSTYPE char*

#include "y.tab.h"

#include <stdio.h>

extern void yyerror(const char*);

int yylineno;

%}



/* Regular definitions */

digit [0-9]

letter [a-zA-Z]

id ({letter}|_)({letter}|{digit}|_)*

unary "++"|"--"

%%

\/\/(.*) ; // ignore comments

[\f\r\t ] ; // ignore whitespaces
```

```
\n { ++yylineno;}

"int" { return T_INT;}

"char" { return T_CHAR;}

"double" { return T_DOUBLE;}

"float" { return T_FLOAT;}

"printf" { return T_PRINTF;}

"scanf" { return T_SCANF;}

"void" { return T_VOID;}

"return" { return T_RETURN;}

"if" { return T_IF;}

"while" { return T_WHILE;}

"for" { return T_FOR;}

"else" { return T_ELSE;}

^"#include"[ ]*<.+\.h> { return T_INCLUDE; }

"true" { return T_TRUE; }

"false" { return T_FALSE; }

[-]?{digit}+ { return T_NUMBER; }

[-]?{digit}+\.{digit}{1,6} { return T_FLOAT_NUM; }

{id} { return T_ID; }

{unary} { return T_UNARY; }

"<=" { return T_LE; }

">=" { return T_GE; }

"==" { return T_EQ; }
```

```
"!=" { return T_NE; }

">" { return T_GT; }

"<" { return T_LT; }

"&&" { return T_AND; }

"||" { return T_OR; }

"+" { return T_ADD; }

"-" { return T_SUBTRACT; }

"/" { return T_DIVIDE; }

"*" { return T_MULTIPLY; }

"," {return *yytext;}

";" {return *yytext;}

\/\/(.*) ;

[\f\r\t ] ;

\n {++yylineno;}

. { return *yytext; }

["].*["] { return T_STR; }

['].['] { return T_CHARACTER; }

"," {return *yytext;}

";" {return *yytext;}

. {yyerror("Unrecognized token");}


%%
```

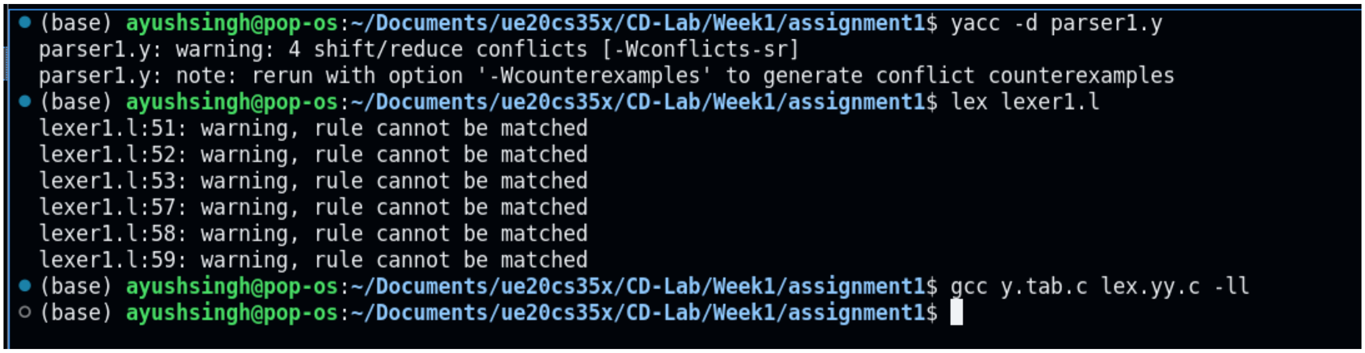
```
int yywrap()

{

return(1);

}
```

```
%{

#include <stdio.h>

#include <stdlib.h>

#include <string.h>



void yyerror(const char*); // error handling function

int yylex(); // declare the function performing lexical analysis

extern int yylineno; // track the line number

extern char* yytext;

int err = 0;

%}



%token T_INT T_CHAR T_DOUBLE T_ID T_FLOAT T_PRINTF T_SCANF T_VOID T_RETURN
T_IF T_WHILE T_FOR T_ELSE T_INCLUDE T_TRUE T_FALSE T_NUMBER T_FLOAT_NUM
T_UNARY T_LE T_GE T_EQ T_NE T_GT T_LT T_AND T_OR T_ADD T_SUBTRACT T_DIVIDE
T_MULTIPLY T_STR T_CHARACTER




%start START



%%
```

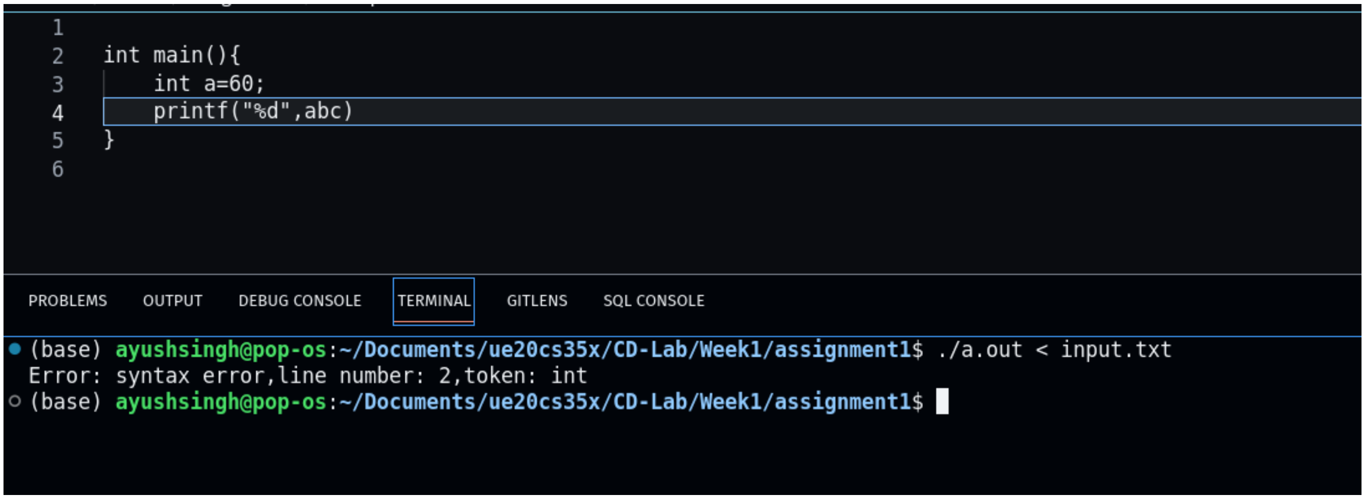
```
START : PROG { if(err==0) {printf("Valid syntax\n"); YYACCEPT;} }

;



PROG: headers main '(' ')' '{' body return '}'

| error ';' PROG

;



headers: headers T_STR

| T_INCLUDE

;



main: datatype T_ID

;



datatype: T_INT

| T_FLOAT

| T_CHAR

| T_VOID

;



body: T_FOR '(' statement ';' condition ';' statement ')' '{' body '}'

| T_IF '(' condition ')' '{' body '}'
| T_IF '(' condition ')' '{' body '}' else
| statement ';'
| body ';'
| T_PRINTF '(' T_STR ')' ';'
| T_SCANF '(' T_STR ',' '&' T_ID ')' ';'
;

else: T_ELSE '{' body '}'
;

condition: value relop value
| T_TRUE
| T_FALSE
;

statement: datatype T_ID init
| T_ID '=' expression
| T_ID relop expression
| T_ID T_UNARY
| T_UNARY T_ID
|
;
```

```
init: '=' value
|
;

expression: expression arithmetic expression
| value
|
;

arithmetic: T_ADD
| T_SUBTRACT
| T_MULTIPLY
| T_DIVIDE
;

relop: T_LT
| T_GT
| T_LE
| T_GE
| T_EQ
| T_NE
;
```

```
value: T_NUMBER

| T_FLOAT_NUM

| T_CHARACTER

| T_ID

;


return: T_RETURN value ';'

;


%%



void yyerror(const char* s)

{

printf("Error: %s,line number: %d,token: %s\n",s,yylineno,yytext);

err = 1;

}



int main(int argc, char* argv[])

{

yyparse();
return 0;



}
```

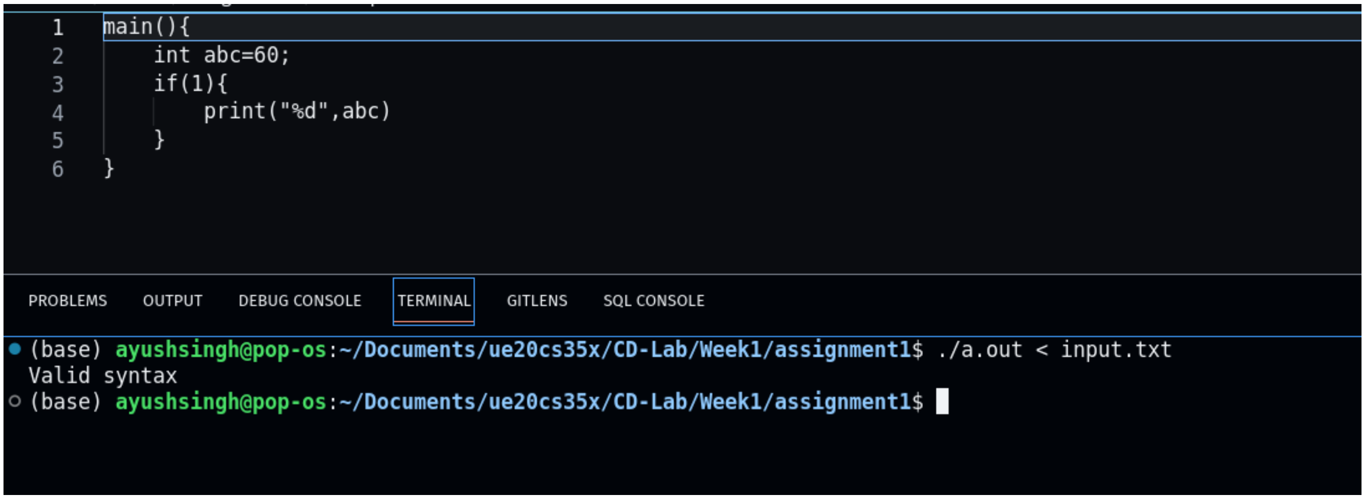


```
(base) ayushsingh@pop-os:~/Documents/ue20cs35x/CD-Lab/Week1/assignment1$ yacc -d parser1.y
parser1.y: warning: 4 shift/reduce conflicts [-Wconflicts-sr]
parser1.y: note: rerun with option '-Wcounterexamples' to generate conflict counterexamples
(base) ayushsingh@pop-os:~/Documents/ue20cs35x/CD-Lab/Week1/assignment1$ lex lexer1.l
lexer1.l:51: warning, rule cannot be matched
lexer1.l:52: warning, rule cannot be matched
lexer1.l:53: warning, rule cannot be matched
lexer1.l:57: warning, rule cannot be matched
lexer1.l:58: warning, rule cannot be matched
lexer1.l:59: warning, rule cannot be matched
(base) ayushsingh@pop-os:~/Documents/ue20cs35x/CD-Lab/Week1/assignment1$ gcc y.tab.c lex.yy.c -ll
(base) ayushsingh@pop-os:~/Documents/ue20cs35x/CD-Lab/Week1/assignment1$
```



```
1
2  int main(){
3      int a=60;
4      printf("%d",abc)
5  }
6
```

```
(base) ayushsingh@pop-os:~/Documents/ue20cs35x/CD-Lab/Week1/assignment1$ ./a.out < input.txt
Error: syntax error,line number: 2,token: int
(base) ayushsingh@pop-os:~/Documents/ue20cs35x/CD-Lab/Week1/assignment1$
```



```
main(){
    int abc=60;
    if(1){
        print("%d",abc)
    }
}
```

```
(base) ayushsingh@pop-os:~/Documents/ue20cs35x/CD-Lab/Week1/assignment1$ ./a.out < input.txt
Valid syntax
(base) ayushsingh@pop-os:~/Documents/ue20cs35x/CD-Lab/Week1/assignment1$
```