# CS6612 – Compiler Lab

**Ex no : 6** **Name : Sreedhar V**

**Date : 02.03.2021** **Reg no: 185001161**

Programming Assignment-6 - Implementation of Syntax Checker using Yacc Tool

---

Develop a Syntax checker to recognize the tokens necessary for the following statements by writing suitable grammars

Assignment statement

Conditional statement

Looping statement

## Code

```
(%{
#include<stdio.h>
#include<string.h>
#include"syn_ch.tab.h"
%}

id ([a-zA-Z_][a-zA-Z0-9_]*|[0-9]+)
rl ("<"|"<="|">"|">="|"=="|"!=")
op ("+"|"-"|"*"|"/"|"%")
un ("++"|"--")
nl "\n"
ts "\t"|" "

%%
"if" {return IF;}
"else" {return ELSE;}
"while" {return WHILE;}
"do" {return DO;}
"for" {return FOR;}
{id} {return ID;}
{rl} {return RL;}
{op} {return OP;}
{un} {return UN;}
{nl} {return NL;}
{ts} ;
. return yytext[0];
```

```
%%

int yywrap(){return 1;}
```

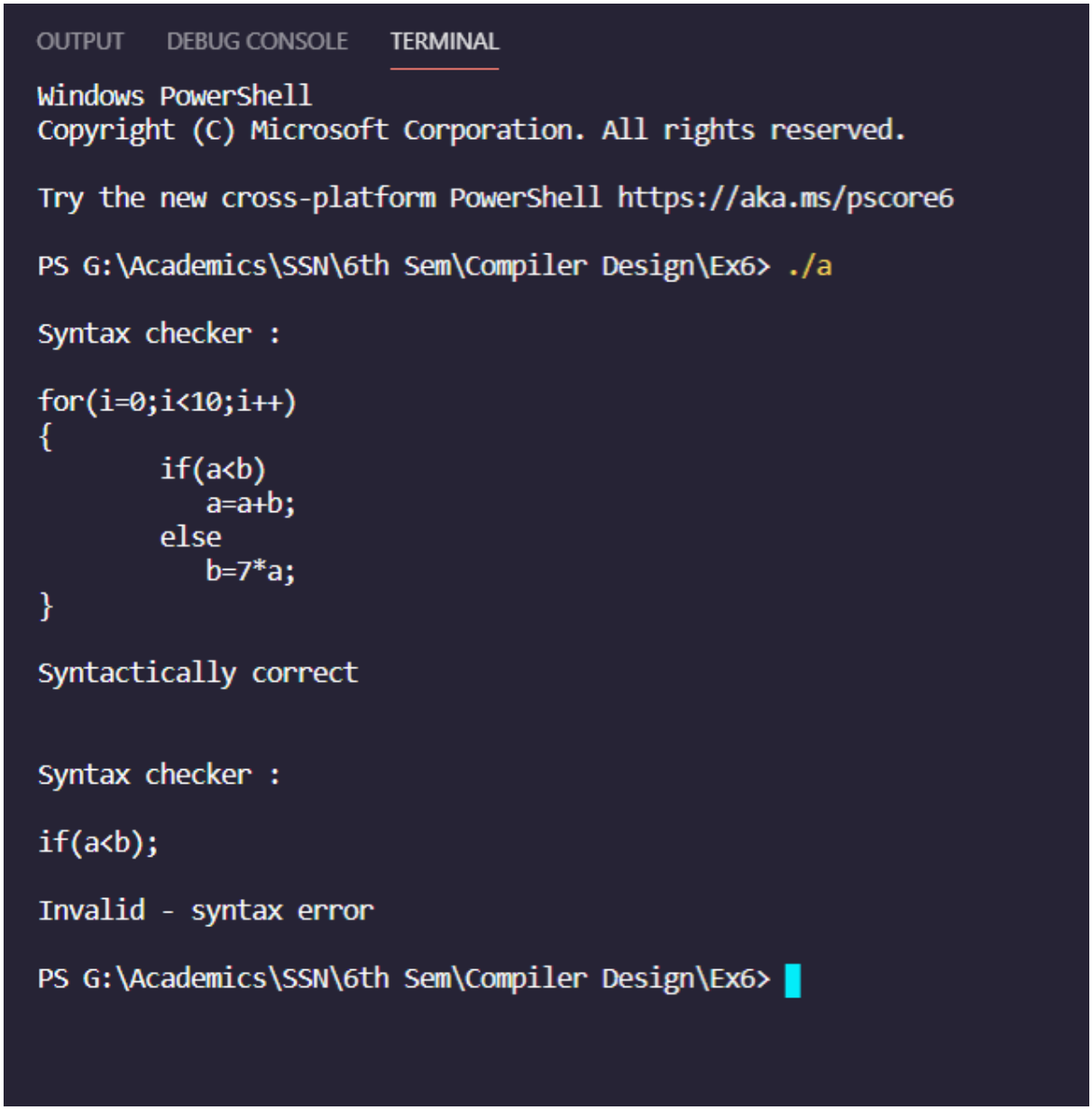
*file)*

*(yacc file)*

```
%{
#include <stdio.h>
int yyerror(char *er);
int yylex(void);
#include <math.h>
#include<stdlib.h>
%}
%token INT STR ID RELOP ARITHOP UNOP DATATYPE IF ELSE

%%
S   : DATATYPE VAR';'|VAR';'|CONDT
VAR : VAR ',' INIT | INIT
INIT : ID | EXPR
EXPR : ID'='ST | ID UNOP | ID'='ID ARITHOP ST | ID'='ID | ID ARITHOP '=
' INT
ST : INT|STR|ID
CONDT :IF '(' CONDT_EXP ')' | ELSE
CONDT_EXP : ID RELOP CONDT_EXP | INT RELOP CONDT_EXP | ID | INT
%%
int main()
{
    while(1)
    {
        yyparse();
    }
    return 0;
}
```

*(Output)*




## Learning Outcome:

- I've learnt how to implement the syntax checker considering all its grammar rules and syntax for C language while execution.

- I’ve learnt the basic syntax of the yacc program and how to implement the grammar rules in c code.
- I’ve leant how the lex program sends the token based on its syntax and yacc program evaluates the stream of tokens based on the given grammar rules and produces the result.