

ASSIGNMENT 6: IMPLEMENTATION OF SYNTAX CHECKER USING YACC TOOL

SRINITHYEE S K
185001166

Code:

SyntaxCheck.y

```
%{
    #include <stdio.h>
    #define YYSTYPE double
    int flag = 0;
}%

%token NUM ASSIGN ID
%token RELOP LOGIC ARITH INCDEC
%token IF ELIF ELSE
%token FOR WHILE

%%

Lines : Block Lines
      | Block
      ;

Block : Loop '{' Block
      | ConStmt '{' Block
      | Expr ';'
      | '}'
      ;

Loop : FOR '(' Expr ';' Condns ';' Expr ')'
      | FOR '(' ';' Condns ';' ')'
      | WHILE '(' Condns ')'
      ;

ConStmt : IF '(' Condns ')'
         | ELIF '(' Condns ')'
         | ELSE
         ;

Condns : Condn LOGIC Condns
        | Condn
        ;
```

```

Condn : ID RELOP ID
      | ID RELOP NUM
      | ID
      ;

```

```

Expr  : Init
      | ID ASSIGN ID ARITH ID
      | ID ASSIGN ID ARITH NUM
      | ID ASSIGN NUM ARITH NUM
      | ID INCDEC
      | INCDEC ID
      ;

```

```

Init  : ID ASSIGN Init
      | ID ASSIGN ID
      | ID ASSIGN NUM
      ;

```

```

%%

```

```

int yyerror(char *s){
    flag = 1;
    //fprintf(stderr, "%s\n", s);
    return 1;
}

```

```

int main(void){

    printf("\nCode Entered:\n\n");
    system("cat program.txt");
    yyparse();

    if(flag){
        printf("\nSyntactically Incorrect.\n");
    }

    else{
        printf("\nSyntactically Correct.\n");
    }

    return 0;
}

```

SyntaxCheck.l

```

%{
    #include <stdio.h>

```

```

#include "y.tab.c"
extern YYSTYPE yylval;
%}

assign    ("=")
relop     ("=="|"!="|>="|<="|<|">")
arithop   ("+"|"-"|"/"|"%"|"*")
incdec    ("++"|"--")
logical   ("||"|"&&")
identifier [a-zA-Z_][a-zA-Z0-9_]*

```

%%

```

[0-9]+    {return NUM;}
{assign}  {return ASSIGN;}
{relop}   {return RELOP;}
{logical} {return LOGIC;}
{arithop} {return ARITH;}
{incdec}  {return INCDEC;}
"if"      {return IF;}
"else if" {return ELIF;}
"else"    {return ELSE;}
"for"     {return FOR;}
"while"   {return WHILE;}
{identifier} {return ID;}

```

```

[ \t]    {};
[\n]     {};
.        {return *yytext;}

```

%%

```

int yywrap(){
    return 1;
}

```

Output:

```
seccon@ubuntu: ~/Desktop
seccon@ubuntu:~/Desktop$ yacc -d check.y
seccon@ubuntu:~/Desktop$ lex check.l
seccon@ubuntu:~/Desktop$ gcc lex.yy.c -lm -w
seccon@ubuntu:~/Desktop$ ./a.out <program.txt

Code Entered:

for(i=0;i<10;i++)
{
    if(i==2){
        x=x+8;
    }
    else{
        y=3*8;
    }
}

Syntactically Correct.
seccon@ubuntu:~/Desktop$ ./a.out <program.txt

Code Entered:

for(i=0;i<10;i++)
{
    if(i==2)
        x=x+8
    else
        y=3*8
}

Syntactically Incorrect.
```