CS6109 - COMPILER DESIGN – LAB

Week 11 – 15.11.2022

(Observations)

1. Write a yacc program to validate the postfix expression.

Code:

```
%{
    #include<stdio.h>
    #include<stdlib.h>
    #include<ctype.h>
%}
%token NUM
%left '+'
%left '-'
%left '*'
%left '/'
%%
start: expr '\n' {printf("%d\n",
$1);return 1; }
;
expr : expr' 'term'+' { $$=$1 + $3;}
    | expr' 'term'-' { $$=$1-$3; }
    | term { $$=$1;}
    ;
term : NUM factor'*' {$$=$1*$3;}
    | NUM factor'/' {$$=$1/$3;}
    | factor
    ;
factor : '('expr')' {$$=$2;}
    | NUM
    ;
%%
yyerror(char const *s)
{
    printf("yyerror %s",s);
```

```
} int yylex() {
    int c;
    c=getchar();
    if (isdigit(c)) {
        yylval=c-'0';
        return NUM;
        }
    return c;
    }
int main(){
    while(1){
        yyparse();
    } return 1;
}
```

2. Write a yacc program to validate the prefix expression.

3. Write a yacc program to validate the nested for construct.

Input:

```
int main()
    {
        for (int i=0; i<2; i++)
        {
            for (int j=0; j<4; j++)
            {
            printf("%d, %d\n",i ,j);
            }
        }
    return 0;
    }
```