**19CSE401 – Compiler Design**

**RDP construction**

**Name:** Ravella Abhinav

**Roll No.:** CB.EN.U4CSE19453

**RDP:**

**package** assignment;

**import** org.antlr.v4.runtime.Token;

**publicclass** RDP {

Lexspeclexer;

Token token;

RDP(Lexspeclexer)

{

**this**.lexer = lexer;

}

**publicboolean**Goal() {

System.***out***.println("VISIT Goal()");

token = lexer.nextToken();

**if**(Expr()) {

**if**(token.getType() == -1) {

**returntrue**;

} **else** {

**return**FAIL();

}

} **else** {

**return**FAIL();

}

}

**publicboolean**FAIL() {

System.***out***.println("FAILED");

**returnfalse**;

}

**publicboolean**Expr() {

System.***out***.println("VISIT Expr()");

**if**(Term()) {

**return**EPrime();

} **else** {

**return**FAIL();

}

}

**publicboolean**Term() {

System.***out***.println("VISIT Term()");

**if**(Factor()) {

**return**TPrime();

} **else** {

**return**FAIL();

}

}

**publicboolean**EPrime() {

System.***out***.println("VISIT EPrime()");

**if**(token.getType() == 8) {

token = lexer.nextToken();

**if**(Term()) {

**return**EPrime();

} **else** {

**return**FAIL();

}

} **elseif**(token.getType() == 11 || token.getType() == -1) {

**returntrue**;

} **else** {

**return**FAIL();

}

}

**publicboolean**TPrime() {

System.***out***.println("VISIT TPrime()");

**if**(token.getType() == 9) {

token = lexer.nextToken();

**if**(Factor()) {

**return**TPrime();

} **else** {

**return**FAIL();

}

} **elseif**(token.getType() == 8 || token.getType() == 11 || token.getType() == -1) {

**returntrue**;

} **else** {

**return**FAIL();

}

}

**publicboolean**Factor() {

System.***out***.println("VISIT Factor()");

**if**(token.getType() == 10) {

token = lexer.nextToken();

**if**(!Expr()) {

**return**FAIL();

}

**if**(token.getType() != 11) {

**return**FAIL();

}

token = lexer.nextToken();

**returntrue**;

} **elseif**(token.getType() == 6 || token.getType() == 7){

token = lexer.nextToken();

**returntrue**;

}**else** {

**return**FAIL();

}

}

}

**Lexspec.g4:**

**lexergrammar** Lexspec;

*Expr* :*TermEPrime*;

*EPrime* :*PLUSORMINUSTermEPrime*

| *EOF*;

*Term* :*FactorTPrime*;

*TPrime* :*MULTORDIVFactorTPrime*

| *EOF*;

*Factor* :*OBExprCB*

| *NUM*

| *NAME*;

*NUM* : [0-9]+;

*NAME*: [a-zA-Z]+;

*PLUSORMINUS* :'+'

| '-';

*MULTORDIV*: '\*'

| '/';

*OB*: '(';

*CB*: ')';

*WS* : [ \n\t\r]+ **->**skip;

**Mymain.java**

package assignment;

import java.io.IOException;

import org.antlr.v4.runtime.\*;

public class mymain {

@SuppressWarnings("deprecation")

public static void main(String[] args) throws IOException

{

// TODO Auto-generated method stub

try

{

CharStream input = new ANTLRFileStream("C:\\Users\\Sujit\\eclipse-workspace2\\assignment\\src\\input");

Lexspeclexer = new Lexspec(input);

RDP rdp = new RDP(lexer);

System.out.println(rdp.Goal());

}

catch(Throwable t)

{

System.out.println("Exception: "+t);

t.printStackTrace();

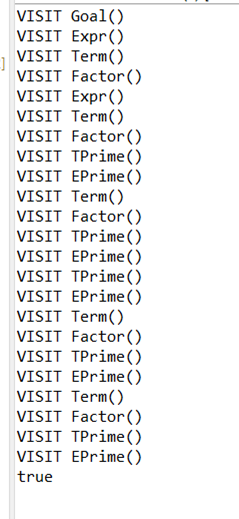
}

}

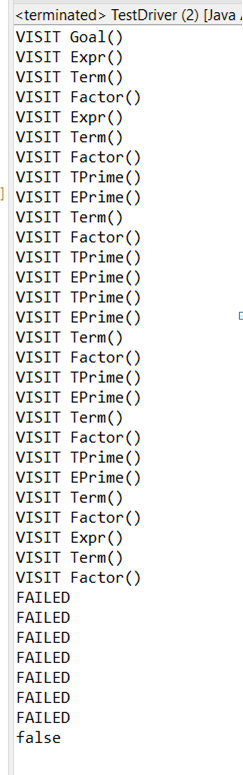
}

**Input/Output:**

**(5\*3)-9+4**

****

1. **Input: (5\*3)-9+4-(**

****