

UNIVERSIDADE FEDERAL DO ESPÍRITO SANTO - UFES
CENTRO UNIVERSITÁRIO DO NORTE DO ESPÍRITO SANTO - CEUNES
DEPARTAMENTO DE COMPUTAÇÃO E ELETRÔNICA - DCEL

HERICK LIMA NUNES
PEDRO HENRIQUE RIZZI FOLLADOR

RELATÓRIO PARA O TRABALHO 1

CONSTRUÇÃO DE UM ANALISADOR (PARSER) ELEMENTAR

SÃO MATEUS
2021

Introdução

O Parser, ou analisador sintático, é uma importante parte de um compilador. Seu trabalho é ler e interpretar o texto puro de um arquivo alvo, tentando reconhecer seus caracteres como símbolos pertencentes à linguagem de programação a qual se refere.

São resumidas como “*Parsing*”, em conjunto, as etapas de: análise léxica e análise sintática.

A análise léxica (ou *scan*) é o processo de leitura simples dos caracteres no arquivo e a verificação de pertinência daqueles caracteres ao alfabeto da linguagem alvo.

Ex.: numa linguagem $L = \{a, b\}$, a análise indicaria erro léxico ao identificar um ‘c’ na sequência que se está a analisar, pois ‘c’ não pertence ao alfabeto L.

A análise sintática (ou *parse*) é a etapa seguinte ao *scan* e que, sabendo que os caracteres são pertencentes à linguagem, se preocupa com a estrutura de organização destes dentro das sequências lidas, de forma que façam sentido. Ex.: se, determinado foi, que um comando $f(x)$ deve ter a estrutura de: *letra* \rightarrow *abre parênteses* \rightarrow *letra* \rightarrow *fecha parênteses*, quaisquer outras formas de estruturação deste comando estarão fora do domínio da linguagem e serão reconhecidas como erro sintático por parte do programa analisador.

Com base nessa breve introdução, na disciplina de Compiladores ministrada pelo Professor Dr. Francisco Santos, foi solicitado que, utilizando o Java Compiler-To-Compiler (JavaCC), construíssemos um analisador léxico e sintático que reconheça uma gramática a qual gera expressões aritméticas e algébricas com parênteses balanceados, tendo como operadores “+”, “-”, “*”, “/” e “^”, e como operandos quaisquer números reais e duas variáveis algébricas (representadas por X e Y).

Metodologia

Com base nos materiais apresentados, utilizou-se neste trabalho uma implementação modular das regras de produção da gramática e utilização de símbolos “tokens” não literais, seja em expressões regulares ou na forma de gramática livre de contexto. Como trata-se de um parser para identificação somente de expressões, até o momento, preocupou-se apenas com os problemas de ambiguidade (a qual não é permitida pelo JavaCC) e recursão à esquerda. Desta forma, a gramática deste parser é dividida entre a regra principal *start()* que leva para a regra que trata o comportamento de expressões: *exp()*. Daí, deriva-se para várias outras sub-regras (funções do tipo *void*) no intuito de eliminar os problemas acima citados.

Cada regra possui tratamento interno para caso de ambiguidade em sua árvore sintática de derivação e cada uma delas também possui no mínimo uma chamada recursiva, o que permite o *parsing* de expressões complexas, como as sugeridas no enunciado do trabalho.

Desenvolvimento

Para a atividade, foi disponibilizado o material onde nele pudemos nos basear para entender quais as expressões que seriam válidas ou não.

Das válidas, seguem:

X , $X^*(X+X)$, $(((((X))))))$, Y , $Y^*(Y+Y)$, $(((((Y))))))$, $X+Y$, $X^*(X+Y)$, $(((((X+Y))))))$,
 $X+Y^AX$, $X^A(X+Y)$, $(((((X^AY))))))$, $(X+Y)^AX-Y^*(X^AY)/Y+(X^AY+X)^AY-X^*(X^AY)/Y$, -256.74 ,
 $-256.74*(5+7)$, $(((((-256.74)))))$, $12.78+14.12^{-3}$, $12.78+14.12^{-3}+5*(32.89^4)/-158$
 $(12.78+18)^{12-3+5*(32.89^4)/87}$,
 $(12.78+18)^{12-3+5*(32.89^4)/87}+(19+17.77)^{15-9*(32.89^4)/87}$,
 $(12.782598+X)^{12-3+5*(32.89^AY)/87}+(19+17.77)^{15-X*(32.8925^4)/87}$.

Das inválidas, seguem:

$X^*(X+Z)$, $(((((X)))))$, $Y(Y+Y)$, $(((((XY))))))$, $X+Y^A$,
 $(X+Y)^AX-Y^*(X^AY)/Y+(XY+X)^AY-X^*(X^AY)/Y$, $(X+Y)^AX-Y^*(X^AY)/Y+(X+X)^AY-X^*(X^AY)/Y$,
 $-+256.74$, $-256.74*(5+/7)$, $12.78+14.12^{-3}+5*(32.89^4/-158$,
 $(12.78+18)^{12-3+5*(32.89^4)/87}+(19+17.77)^{15-9*(32.89^4)/87}$,
 $(12.78+18)^{12-3+5*(32.89^4X)/87}+(19+17.77)^{15-9*(32.89^4)/87}$.
 $(12.78+18)^{XY-3+5*(32.89^4)/87}+(19+17.77)^{AY-9*(32.89^4)/87}$,
 $(12.78+18)^{X90.4-3+5*(32.89^4)/87}+(19+17.77)^{AY-9*(32.89^4)/87}$,
 $Y(12.78+18)^{X90.4-3+5*(32.89^4)/87}+(19+17.77)^{AY-9*(32.89^4)/87}$,
 $XY(12.78+18)^{X90.4-3+5*(32.89^4)/87}+(19+17.77)^{AY-9*(32.89^4)/87}$,
 $Y(12.78+18)^{X90.4-3+5*(32.89^4)/87}+(19+17.77)^{AY-9*(32.89^4)/87}Y$,
 $(12,782598+X)^{12-3+5*(32.89^AY)/87}+(19+17.77)^{15-X*(32.8925^4)/87}$.

Os resultados obtidos pelo Parser foram então capturados e colocados nas figuras abaixo de acordo com a ordem dos testes tanto para as expressões válidas, quanto para as expressões inválidas.

Capturas do Parser para Expressões Válidas

1) X

```

C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Variavel: X
variavel corresponde: X
Compilando.....

C:\Compiladores\Trabalho001>
```

2) $X*(X+X)$

```

C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Variavel: X
variavel corresponde: X
Operador/Sinal: *
Parentese: (
Variavel: X
variavel corresponde: X
Operador/Sinal: +
Variavel: X
variavel corresponde: X
Parentese: )
Compilando.....

C:\Compiladores\Trabalho001>
```

3) ((((((X))))))

```

C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Variavel: X
variavel corresponde: X
Parentese: )
Parentese: )
Parentese: )
Parentese: )
Parentese: )
Parentese: )
Compilando.....

C:\Compiladores\Trabalho001>
```

4) Y

```

C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Variavel: Y
variavel corresponde: Y
Compilando.....

C:\Compiladores\Trabalho001>
```

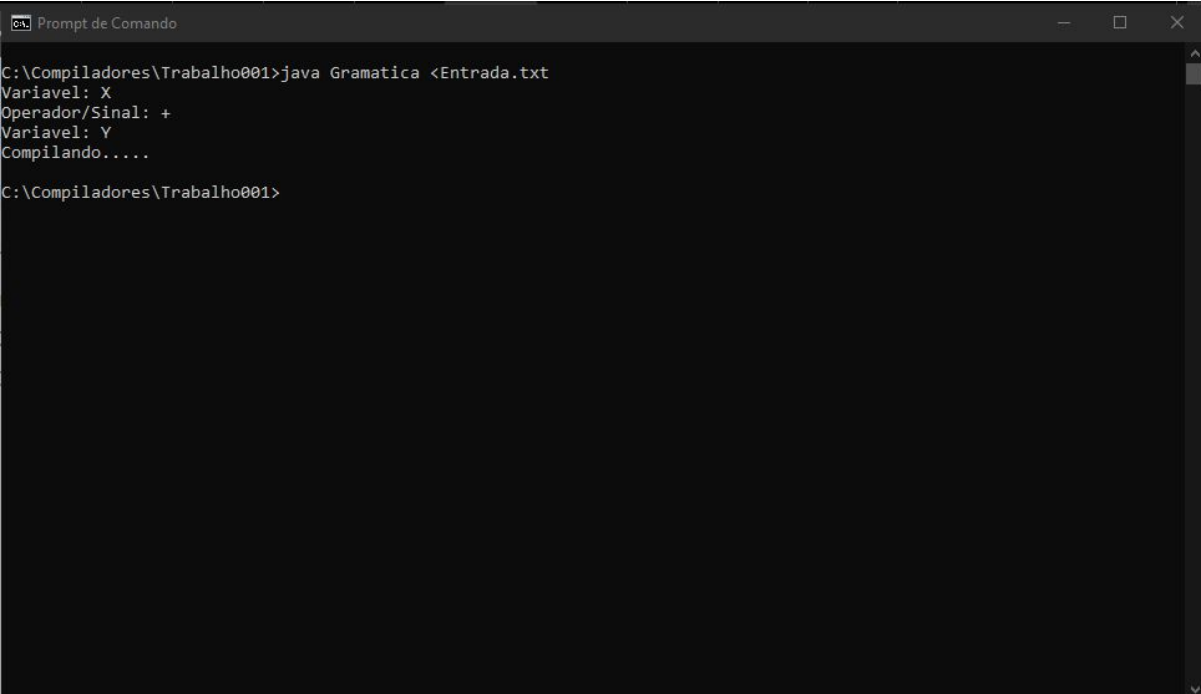
5) $Y*(Y+Y)$

```
Prompt de Comando
C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Variavel: Y
Operador/Sinal: *
Parentese: (
Variavel: Y
Operador/Sinal: +
Variavel: Y
Parentese: )
Compilando.....
C:\Compiladores\Trabalho001>
```

6) ((((((Y))))))

```
Prompt de Comando
C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Variavel: Y
Parentese: )
Parentese: )
Parentese: )
Parentese: )
Parentese: )
Parentese: )
Compilando.....
C:\Compiladores\Trabalho001>
```

7) $X+Y$

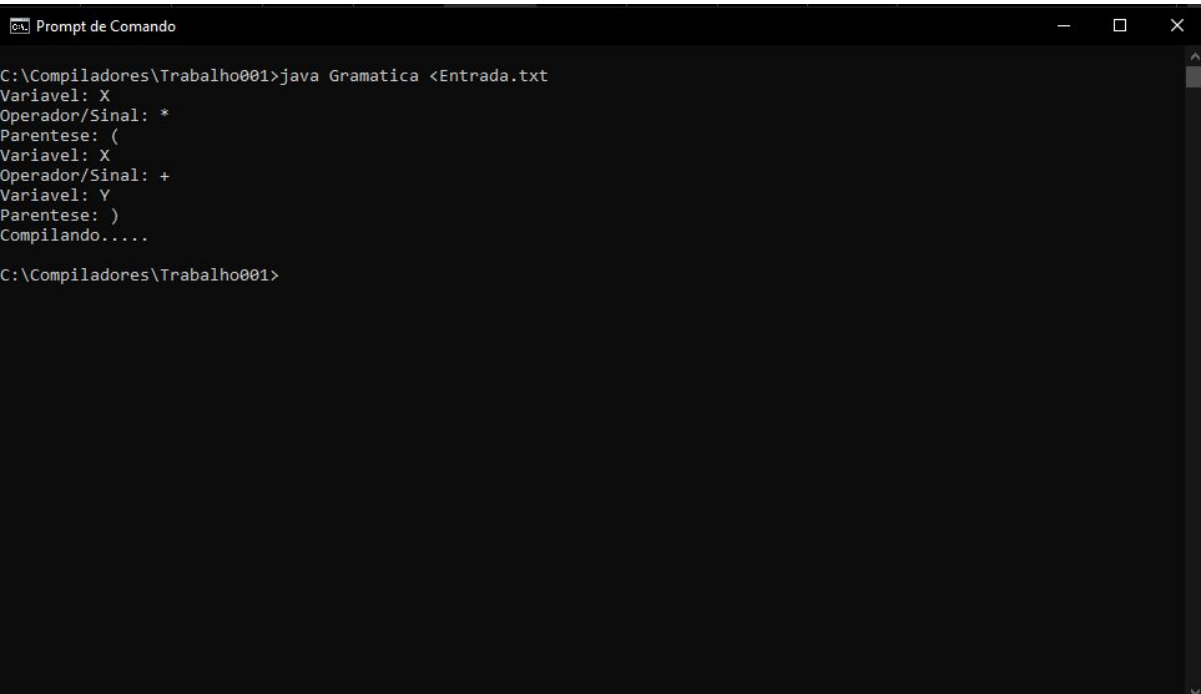


```

C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Variavel: X
Operador/Sinal: +
Variavel: Y
Compilando.....

C:\Compiladores\Trabalho001>
```

8) $X*(X+Y)$



```

C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Variavel: X
Operador/Sinal: *
Parentese: (
Variavel: X
Operador/Sinal: +
Variavel: Y
Parentese: )
Compilando.....

C:\Compiladores\Trabalho001>
```


9) ((((((X+Y))))))

```

C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Variavel: X
Operador/Sinal: +
Variavel: Y
Parentese: )
Parentese: )
Parentese: )
Parentese: )
Parentese: )
Parentese: )
Compilando.....

C:\Compiladores\Trabalho001>
```

10) $X+Y^X$

```

C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Variavel: X
Operador/Sinal: +
Variavel: Y
Operador/Sinal: ^
Variavel: X
Compilando.....

C:\Compiladores\Trabalho001>
```

11) $X^{(X+Y)}$

```
Prompt de Comando
C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Variavel: X
Operador/Sinal: ^
Parentese: (
Variavel: X
Operador/Sinal: +
Variavel: Y
Parentese: )
Compilando.....

C:\Compiladores\Trabalho001>
```

12) ((((((X*Y))))))

```
Prompt de Comando
C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Variavel: X
Operador/Sinal: *
Variavel: Y
Parentese: )
Parentese: )
Parentese: )
Parentese: )
Parentese: )
Parentese: )
Parentese: )
Compilando.....

C:\Compiladores\Trabalho001>
```

13) $(X+Y)^X - Y * (X^Y) / Y + (X * Y + X) * Y - X * (X^Y) / Y$

```
Prompt de Comando
C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Parentese: (
Variavel: X
Operador/Sinal: +
Variavel: Y
Parentese: )
Operador/Sinal: ^
Variavel: X
Operador/Sinal: -
Variavel: Y
Operador/Sinal: *
Parentese: (
Variavel: X
Operador/Sinal: ^
Variavel: Y
Parentese: )
Operador/Sinal: /
Variavel: Y
Operador/Sinal: +
Parentese: (
Variavel: X
Operador/Sinal: *
Variavel: Y
Operador/Sinal: +
Variavel: X
Parentese: )
Operador/Sinal: *
Variavel: Y
Operador/Sinal: -
Variavel: X
Operador/Sinal: *
Parentese: (
Variavel: X
Operador/Sinal: ^
Variavel: Y
Parentese: )
Operador/Sinal: /
Variavel: Y
Compilando.....
C:\Compiladores\Trabalho001>
```

14) -256.74

```
Prompt de Comando
C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Operador/Sinal: -
Numero: 256.74
Compilando.....
C:\Compiladores\Trabalho001>
```

15) $-256.74 \cdot (5+7)$

```
Prompt de Comando
C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Operador/Sinal: -
Numero: 256.74
Operador/Sinal: *
Parentese: (
Numero: 5
Operador/Sinal: +
Numero: 7
Parentese: )
Compilando.....

C:\Compiladores\Trabalho001>
```

16) $((((((-256.74))))))$

```
Prompt de Comando
C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Operador/Sinal: -
Numero: 256.74
Parentese: )
Parentese: )
Parentese: )
Parentese: )
Parentese: )
Parentese: )
Compilando.....

C:\Compiladores\Trabalho001>
```

17) $12.78 + 14.12^{-3}$

```
Prompt de Comando

C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Numero: 12.78
Operador/Sinal: +
Numero: 14.12
Operador/Sinal: ^
Operador/Sinal: -
Numero: 3
Compilando.....

C:\Compiladores\Trabalho001>
```

18) $12.78 + 14.12^{-3} + 5 * (32.89^4) / -158$

```
Prompt de Comando

C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Numero: 12.78
Operador/Sinal: +
Numero: 14.12
Operador/Sinal: ^
Operador/Sinal: -
Numero: 3
Operador/Sinal: +
Numero: 5
Operador/Sinal: *
Parentese: (
Numero: 32.89
Operador/Sinal: ^
Numero: 4
Parentese: )
Operador/Sinal: /
Operador/Sinal: -
Numero: 158
Compilando.....

C:\Compiladores\Trabalho001>
```

19) $(12.78+18)^{12-3+5*(32.89^4)}/87$

```

C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Parentese: (
Numero: 12.78
Operador/Sinal: +
Numero: 18
Parentese: )
Operador/Sinal: ^
Numero: 12
Operador/Sinal: -
Numero: 3
Operador/Sinal: +
Numero: 5
Operador/Sinal: *
Parentese: (
Numero: 32.89
Operador/Sinal: ^
Numero: 4
Parentese: )
Operador/Sinal: /
Numero: 87
Compilando.....

C:\Compiladores\Trabalho001>
```

20) $(12.78+18)^{12-3+5*(32.89^4)}/87+(19+17.77)*15-9*(32.89^4)/87$

```

C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Parentese: (
Numero: 12.78
Operador/Sinal: +
Numero: 18
Parentese: )
Operador/Sinal: ^
Numero: 12
Operador/Sinal: -
Numero: 3
Operador/Sinal: +
Numero: 5
Operador/Sinal: *
Parentese: (
Numero: 32.89
Operador/Sinal: ^
Numero: 4
Parentese: )
Operador/Sinal: /
Numero: 87
Operador/Sinal: +
Parentese: (
Numero: 19
Operador/Sinal: +
Numero: 17.77
Parentese: )
Operador/Sinal: *
Numero: 15
Operador/Sinal: -
Numero: 9
Operador/Sinal: *
Parentese: (
Numero: 32.89
Operador/Sinal: ^
Numero: 4
Parentese: )
Operador/Sinal: /
Numero: 87
Compilando.....

C:\Compiladores\Trabalho001>
```

21) $(12.782598+X)^{12-3+5*(32.89^Y)/87+(19+17.77)*15-X*(32.8925^4)/87}$

```
Prompt de Comando
C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Parentese: (
Numero: 12.782598
Operador/Sinal: +
Variavel: X
Parentese: )
Operador/Sinal: ^
Numero: 12
Operador/Sinal: -
Numero: 3
Operador/Sinal: +
Numero: 5
Operador/Sinal: *
Parentese: (
Numero: 32.89
Operador/Sinal: ^
Variavel: Y
Parentese: )
Operador/Sinal: /
Numero: 87
Operador/Sinal: +
Parentese: (
Numero: 19
Operador/Sinal: +
Numero: 17.77
Parentese: )
Operador/Sinal: *
Numero: 15
Operador/Sinal: -
Variavel: X
Operador/Sinal: *
Parentese: (
Numero: 32.8925
Operador/Sinal: ^
Numero: 4
Parentese: )
Operador/Sinal: /
Numero: 87
Compilando.....
C:\Compiladores\Trabalho001>
```

Capturas do Parser para Expressões Inválidas

1) $X*(X+Z)$

```

C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Variavel: X
Operador/Sinal: *
Parentese: (
Variavel: X
Operador/Sinal: +
Exception in thread "main" TokenMgrError: Lexical error at line 1, column 6.  Encountered: "Z" (90), after : ""
    at GramaticaTokenManager.getNextToken(GramaticaTokenManager.java:290)
    at Gramatica.jj_ntk_f(Gramatica.java:259)
    at Gramatica.termo(Gramatica.java:44)
    at Gramatica.expressaoFormada(Gramatica.java:39)
    at Gramatica.termo(Gramatica.java:52)
    at Gramatica.expressaoFormada(Gramatica.java:39)
    at Gramatica.main(Gramatica.java:7)

C:\Compiladores\Trabalho001>
```

2) ((((((X))))))

```

C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Variavel: X
Parentese: )
Parentese: )
Parentese: )
Parentese: )
Parentese: )
Encountered "<EOF>" at line 1, column 10.
Was expecting one of:
    "+" ...
    "-" ...
    "*" ...
    "/" ...
    "^" ...
    ")" ...

C:\Compiladores\Trabalho001>
```


3) Y(Y+Y)

```
Prompt de Comando

C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Variavel: Y
Parentese: (
Erro.
ParseException: Encountered " "(" "(" at line 1, column 2.
Was expecting:
    <EOF>

    at Gramatica.generateParseException(Gramatica.java:568)
    at Gramatica.jj_consume_token(Gramatica.java:452)
    at Gramatica.start(Gramatica.java:17)
    at Gramatica.main(Gramatica.java:7)

C:\Compiladores\Trabalho001>
```

4) ((((((XY))))))

```
Prompt de Comando

C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Parentese: (
Variavel: X
Variavel: Y
Erro.
ParseException: Encountered " <VAR> "Y "" at line 1, column 8.
Was expecting:
    ")" ...

    at Gramatica.generateParseException(Gramatica.java:568)
    at Gramatica.jj_consume_token(Gramatica.java:452)
    at Gramatica.element(Gramatica.java:87)
    at Gramatica.unary(Gramatica.java:72)
    at Gramatica.term(Gramatica.java:43)
    at Gramatica.exp(Gramatica.java:21)
    at Gramatica.element(Gramatica.java:86)
    at Gramatica.unary(Gramatica.java:72)
    at Gramatica.term(Gramatica.java:43)
    at Gramatica.exp(Gramatica.java:21)
    at Gramatica.element(Gramatica.java:86)
    at Gramatica.unary(Gramatica.java:72)
    at Gramatica.term(Gramatica.java:43)
    at Gramatica.exp(Gramatica.java:21)
    at Gramatica.element(Gramatica.java:86)
    at Gramatica.unary(Gramatica.java:72)
    at Gramatica.term(Gramatica.java:43)
    at Gramatica.exp(Gramatica.java:21)
    at Gramatica.element(Gramatica.java:86)
    at Gramatica.unary(Gramatica.java:72)
    at Gramatica.term(Gramatica.java:43)
    at Gramatica.exp(Gramatica.java:21)
    at Gramatica.start(Gramatica.java:16)
    at Gramatica.main(Gramatica.java:7)

C:\Compiladores\Trabalho001>
```

5) $X+Y^{\wedge}$

```
Prompt de Comando

C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Variavel: X
Operador/Sinal: +
Variavel: Y
Operador/Sinal: ^
Erro.
ParseException: Encountered " ^" "" at line 1, column 4.
Was expecting:
    <EOF>

    at Gramatica.generateParseException(Gramatica.java:568)
    at Gramatica.jj_consume_token(Gramatica.java:452)
    at Gramatica.start(Gramatica.java:17)
    at Gramatica.main(Gramatica.java:7)

C:\Compiladores\Trabalho001>
```

6) $(X+Y)^{\wedge}X-Y*(X^{\wedge}Y)/Y+(XY+X)*Y-X*(X^{\wedge}Y)/Y$

```
Selecionar Prompt de Comando

C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Parentese: (
Variavel: X
Operador/Sinal: +
Variavel: Y
Parentese: )
Operador/Sinal: ^
Variavel: X
Operador/Sinal: -
Variavel: Y
Operador/Sinal: *
Parentese: (
Variavel: X
Operador/Sinal: ^
Variavel: Y
Parentese: )
Operador/Sinal: /
Variavel: Y
Operador/Sinal: +
Parentese: (
Variavel: X
Variavel: Y
Erro.
ParseException: Encountered " <VAR> "Y" "" at line 1, column 21.
Was expecting:
    ")" ...

    at Gramatica.generateParseException(Gramatica.java:568)
    at Gramatica.jj_consume_token(Gramatica.java:452)
    at Gramatica.element(Gramatica.java:87)
    at Gramatica.unary(Gramatica.java:72)
    at Gramatica.term(Gramatica.java:43)
    at Gramatica.exp(Gramatica.java:21)
    at Gramatica.exp(Gramatica.java:31)
    at Gramatica.exp(Gramatica.java:34)
    at Gramatica.start(Gramatica.java:16)
    at Gramatica.main(Gramatica.java:7)

C:\Compiladores\Trabalho001>
```

7) $(X+Y)^X - Y * (X^Y) / Y + (X+X) * Y - X^*$

```
Prompt de Comando
C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Parentese: (
Variavel: X
Operador/Sinal: +
Variavel: Y
Parentese: )
Operador/Sinal: ^
Variavel: X
Operador/Sinal: -
Variavel: Y
Operador/Sinal: *
Parentese: (
Variavel: X
Operador/Sinal: ^
Variavel: Y
Parentese: )
Operador/Sinal: /
Variavel: Y
Operador/Sinal: +
Parentese: (
Variavel: X
Operador/Sinal: +
Variavel: X
Parentese: )
Operador/Sinal: *
Variavel: Y
Operador/Sinal: -
Variavel: X
Operador/Sinal: *
Operador/Sinal: /
Erro.
ParseException: Encountered " "*" "*" "" at line 1, column 28.
Was expecting:
    <EOF>

    at Gramatica.generateParseException(Gramatica.java:568)
    at Gramatica.jj_consume_token(Gramatica.java:452)
    at Gramatica.start(Gramatica.java:17)
    at Gramatica.main(Gramatica.java:7)

C:\Compiladores\Trabalho001>
```

8) $-+256.74$

```
Prompt de Comando
C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Operador/Sinal: -
Operador/Sinal: +
Erro.
ParseException: Encountered "" at line 1, column 1.
Was expecting one of:

    at Gramatica.generateParseException(Gramatica.java:568)
    at Gramatica.jj_consume_token(Gramatica.java:452)
    at Gramatica.unary(Gramatica.java:74)
    at Gramatica.term(Gramatica.java:43)
    at Gramatica.exp(Gramatica.java:21)
    at Gramatica.start(Gramatica.java:16)
    at Gramatica.main(Gramatica.java:7)

C:\Compiladores\Trabalho001>
```

9) $-256.74 * (5 + /7)$

```
Prompt de Comando

C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Operador/Sinal: -
Numero: 256.74
Operador/Sinal: *
Parentese: (
Numero: 5
Operador/Sinal: +
Operador/Sinal: /
Erro.
ParseException: Encountered " "+" "+" at line 1, column 11.
Was expecting:
    ")" ...

    at Gramatica.generateParseException(Gramatica.java:568)
    at Gramatica.jj_consume_token(Gramatica.java:452)
    at Gramatica.element(Gramatica.java:87)
    at Gramatica.unary(Gramatica.java:72)
    at Gramatica.term(Gramatica.java:43)
    at Gramatica.term(Gramatica.java:53)
    at Gramatica.exp(Gramatica.java:21)
    at Gramatica.start(Gramatica.java:16)
    at Gramatica.main(Gramatica.java:7)

C:\Compiladores\Trabalho001>
```

10) $12.78 + 14.12^{-3} + 5 * (32.89^4 / -158$

```
Prompt de Comando

C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Numero: 12.78
Operador/Sinal: +
Numero: 14.12
Operador/Sinal: ^
Operador/Sinal: -
Numero: 3
Operador/Sinal: +
Numero: 5
Operador/Sinal: *
Parentese: (
Numero: 32.89
Operador/Sinal: ^
Numero: 4
Operador/Sinal: /
Operador/Sinal: -
Numero: 158
Erro.
ParseException: Encountered "<EOF>" at line 1, column 30.
Was expecting:
    ")" ...

    at Gramatica.generateParseException(Gramatica.java:568)
    at Gramatica.jj_consume_token(Gramatica.java:452)
    at Gramatica.element(Gramatica.java:87)
    at Gramatica.unary(Gramatica.java:72)
    at Gramatica.term(Gramatica.java:43)
    at Gramatica.term(Gramatica.java:53)
    at Gramatica.exp(Gramatica.java:21)
    at Gramatica.exp(Gramatica.java:31)
    at Gramatica.exp(Gramatica.java:31)
    at Gramatica.start(Gramatica.java:16)
    at Gramatica.main(Gramatica.java:7)

C:\Compiladores\Trabalho001>
```

11) $(12.78 + 18^{12} - 3 + 5 * (32.89^4) / 87 + (19 + 17.77) * 15 - 9 * (32.89^4) / 87$

```
Prompt de Comando
C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Parentese: (
Numero: 12.78
Operador/Sinal: +
Numero: 18
Operador/Sinal: ^
Numero: 12
Operador/Sinal: -
Numero: 3
Operador/Sinal: +
Numero: 5
Operador/Sinal: *
Parentese: (
Numero: 32.89
Operador/Sinal: ^
Numero: 4
Parentese: )
Operador/Sinal: /
Numero: 87
Operador/Sinal: +
Parentese: (
Numero: 19
Operador/Sinal: +
Numero: 17.77
Parentese: )
Operador/Sinal: *
Numero: 15
Operador/Sinal: -
Numero: 9
Operador/Sinal: *
Parentese: (
Numero: 32.89
Operador/Sinal: ^
Numero: 4
Parentese: )
Operador/Sinal: /
Numero: 87
Erro.
ParseException: Encountered "<EOF>" at line 1, column 58.
Was expecting:
    ")" ...

    at Gramatica.generateParseException(Gramatica.java:568)
    at Gramatica.jj_consume_token(Gramatica.java:452)
    at Gramatica.element(Gramatica.java:87)
    at Gramatica.unary(Gramatica.java:72)
    at Gramatica.term(Gramatica.java:43)
    at Gramatica.exp(Gramatica.java:21)
    at Gramatica.start(Gramatica.java:16)
```

12) $(12.78+18)^{12-3+5*(32.89^4X)/87+(19+17.77)*15-9*(32.89^4)/87}$

```
Prompt de Comando
C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Parentese: (
Numero: 12.78
Operador/Sinal: +
Numero: 18
Parentese: )
Operador/Sinal: ^
Numero: 12
Operador/Sinal: -
Numero: 3
Operador/Sinal: +
Numero: 5
Operador/Sinal: *
Parentese: (
Numero: 32.89
Operador/Sinal: ^
Numero: 4
Variavel: X
Erro.
ParseException: Encountered " <VAR> "X "" at line 1, column 27.
Was expecting:
    ")" ...

    at Gramatica.generateParseException(Gramatica.java:568)
    at Gramatica.jj_consume_token(Gramatica.java:452)
    at Gramatica.element(Gramatica.java:87)
    at Gramatica.unary(Gramatica.java:72)
    at Gramatica.term(Gramatica.java:43)
    at Gramatica.term(Gramatica.java:53)
    at Gramatica.exp(Gramatica.java:21)
    at Gramatica.exp(Gramatica.java:31)
    at Gramatica.exp(Gramatica.java:34)
    at Gramatica.start(Gramatica.java:16)
    at Gramatica.main(Gramatica.java:7)

C:\Compiladores\Trabalho001>
```

13) $(12.78+18)^{XY-3+5*(32.89^4)/87+(19+17.77^Y-9*(32.89^4)/87}$

```
Prompt de Comando
C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Parentese: (
Numero: 12.78
Operador/Sinal: +
Numero: 18
Parentese: )
Operador/Sinal: ^
Variavel: X
Variavel: Y
Erro.
ParseException: Encountered " <VAR> "Y "" at line 1, column 13.
Was expecting:
    <EOF>

    at Gramatica.generateParseException(Gramatica.java:568)
    at Gramatica.jj_consume_token(Gramatica.java:452)
    at Gramatica.start(Gramatica.java:17)
    at Gramatica.main(Gramatica.java:7)

C:\Compiladores\Trabalho001>
```


14) $(12.78+18)^X 90.4-3+5*(32.89^4)/87+(19+17.77)^Y-9*(32.89^4)/87$

```
Prompt de Comando
C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Parentese: (
Numero: 12.78
Operador/Sinal: +
Numero: 18
Parentese: )
Operador/Sinal: ^
Variavel: X
Numero: 90.4
Erro.
ParseException: Encountered " <NUMBER> "90.4 "" at line 1, column 13.
Was expecting:
    <EOF>

    at Gramatica.generateParseException(Gramatica.java:568)
    at Gramatica.jj_consume_token(Gramatica.java:452)
    at Gramatica.start(Gramatica.java:17)
    at Gramatica.main(Gramatica.java:7)

C:\Compiladores\Trabalho001>
```

15) $Y(12.78+18)^X 90.4-3+5*(32.89^4)/87+(19+17.77)^Y-9*(32.89^4)/87$

```
Prompt de Comando
C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Variavel: Y
Parentese: (
Erro.
ParseException: Encountered " "(" "(" "" at line 1, column 2.
Was expecting:
    <EOF>

    at Gramatica.generateParseException(Gramatica.java:568)
    at Gramatica.jj_consume_token(Gramatica.java:452)
    at Gramatica.start(Gramatica.java:17)
    at Gramatica.main(Gramatica.java:7)

C:\Compiladores\Trabalho001>
```

16) $XY(12.78+18)^X 90.4-3+5*(32.89^4)/87+(19+17.77)^Y-9*(32.89^4)/87$

```
Prompt de Comando
C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Variavel: X
Variavel: Y
Erro.
ParseException: Encountered " <VAR> "Y "" at line 1, column 2.
Was expecting:
    <EOF>

    at Gramatica.generateParseException(Gramatica.java:568)
    at Gramatica.jj_consume_token(Gramatica.java:452)
    at Gramatica.start(Gramatica.java:17)
    at Gramatica.main(Gramatica.java:7)

C:\Compiladores\Trabalho001>
```

17) $Y(12.78+18)^X 90.4-3+5*(32.89^4)/87+(19+17.77)^Y-9*(32.89^4)/87Y$

```
Prompt de Comando
C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Variavel: Y
Parentese: (
Erro.
ParseException: Encountered " "(" "(" at line 1, column 2.
Was expecting:
    <EOF>

    at Gramatica.generateParseException(Gramatica.java:568)
    at Gramatica.jj_consume_token(Gramatica.java:452)
    at Gramatica.start(Gramatica.java:17)
    at Gramatica.main(Gramatica.java:7)

C:\Compiladores\Trabalho001>
```

18) $(12,782598+X)^{12-3+5*(32.89^Y)/87+(19+17.77)*15-X*(32.8925^4)/87}$

```
Prompt de Comando
C:\Compiladores\Trabalho001>java Gramatica <Entrada.txt
Parentese: (
Numero: 12
Exception in thread "main" TokenMgrError: Lexical error at line 1, column 4.  Encountered: "," (44), after : ""
    at GramaticaTokenManager.getNextToken(GramaticaTokenManager.java:235)
    at Gramatica.jj_scan_token(Gramatica.java:462)
    at Gramatica.jj_3_5(Gramatica.java:307)
    at Gramatica.jj_3_4(Gramatica.java:295)
    at Gramatica.jj_2_4(Gramatica.java:121)
    at Gramatica.term(Gramatica.java:46)
    at Gramatica.exp(Gramatica.java:21)
    at Gramatica.element(Gramatica.java:86)
    at Gramatica.unary(Gramatica.java:72)
    at Gramatica.term(Gramatica.java:43)
    at Gramatica.exp(Gramatica.java:21)
    at Gramatica.start(Gramatica.java:16)
    at Gramatica.main(Gramatica.java:7)

C:\Compiladores\Trabalho001>
```


Código

```
TOKEN: {
    <ADD:("+")> { System.out.println("Operador/Sinal: "+image); }
    | <SUB:("-")> { System.out.println("Operador/Sinal: "+image); }
    | <MUL:("*")> { System.out.println("Operador/Sinal: "+image); }
    | <DIV:("/")> { System.out.println("Operador/Sinal: "+image); }
    | <EXP:("^")> { System.out.println("Operador/Sinal: "+image); }
    | <LBR:("(")> { System.out.println("Parentese: "+image); }
    | <RBR:(")")> { System.out.println("Parentese: "+image); }
    | <DOT:(".")> { System.out.println("Ponto: "+image); }
}

TOKEN: {
    <NUMBER: (<DIGIT>)+ ( <DOT> (<DIGIT>)+ )?> { System.out.println("Numero: "+image); }
    | <DIGIT: ["0"-"9"]>
}

TOKEN : {
    <VAR: ["X", "Y"]> { System.out.println("Variavel: "+image); }
}

void start() : {}{
    exp() <EOF>
}

void exp() : {}{
    term() ( <ADD> exp() | <SUB> exp() ) *
}

void term() : {}{
    unary() ( <MUL> term() | <DIV> term() | <EXP> term() ) *
}

void unary(): {}{
    "-" element() | element()
}

void element(): {}{
    <NUMBER> | <VAR> | <LBR> exp() <RBR>
}
```

Conclusão

No trabalho prático desenvolvido acima, abordamos um assunto muito importante para a disciplina de compiladores, o Parser para expressões algébricas e aritméticas. De acordo com o que foi solicitado no enunciado proposto na descrição da atividade, conseguimos implementar todos os tópicos, apesar de encontrarmos algumas dificuldades referentes ao uso dos parênteses balanceados, números reais e da potenciação.

O desenvolvimento deste trouxe uma maior afinidade com o ambiente de desenvolvimento do JavaCC, alguns conhecimentos que serão necessários no decorrer da disciplina e também a aplicação dos conceitos aprendidos até o momento.