

# Trabalho Prático 2 - Compiladores 1

João Antonio Oliveira Pedrosa  
2019006752

Dezembro de 2021

## 1 Questão 1

| non terminal | (          | )          | ]          | \$ |
|--------------|------------|------------|------------|----|
| S            | S -> "(" X | S -> F ")" | S -> E "]" |    |
| X            |            | X -> E ")" | X -> F "]" |    |
| E            |            | E -> A     | E -> A     |    |
| F            |            | F -> A     | F -> A     |    |
| A            |            | A -> empty | A -> empty |    |

Essa gramática faz parte de LL(1) pois não existe recursividade à esquerda.

## 2 Questão 2

### 2.1

- E0

```
{ [goal' -> goal, $]; Go To E1
[goal -> expr, $]; Go To E2
[expr -> term + expr, $]; Go To E3
[expr -> term, $]; Go To E3
[term -> factor * term, +/$]; Go To E4
[term -> factor, +/$]; Go To E4
[factor -> id, */+/$] Go To E5}
```

- E1

```
{ [goal' -> goal , $] }
```

- E2

```
{ [goal -> expr , $] }
```

- E3

```
{ [expr -> term + expr, $]; Go To E6  
  [expr -> term , $] }
```

- E4

```
{ [term -> factor * term, +/$]; Go To E7  
  [term -> factor , +/$] }
```

- E5

```
{ [factor -> id , */+/$] }
```

- E6

```
{ [expr -> term + expr, $]; Go To E8  
  [expr -> term + expr, $];  
  [expr -> term, $];  
  [term -> factor * term, +/$];  
  [term -> factor, +/$];  
  [factor -> id, */+/$] }
```

- E7

```
{ [term -> factor * term, +/$]; Go To E9  
  [term -> factor * term, +/$];  
  [term -> factor, +/$];  
  [factor -> id, */+/$] }
```

- E8

```
{ [expr -> term + expr , $] }
```

- E9

{ [term -> factor \* term , +/\$] }

## 2.2

| State | Action |    |    |     | Go To |      |      |      |        |
|-------|--------|----|----|-----|-------|------|------|------|--------|
|       | +      | *  | id | \$  | goal' | goal | expr | term | factor |
| 0     |        |    | s5 |     |       | 1    | 2    | 3    | 4      |
| 1     |        |    |    | acc |       |      |      |      |        |
| 2     |        |    |    | r1  |       |      |      |      |        |
| 3     | s6     |    |    | r3  |       |      |      |      |        |
| 4     | r5     | s7 |    | r5  |       |      |      |      |        |
| 5     | r6     | s6 |    | r6  |       |      |      |      |        |
| 6     |        |    | s5 |     |       |      |      |      |        |
| 7     |        |    | s5 |     |       |      | 8    | 3    | 4      |
| 8     |        |    |    | r2  |       |      |      | 9    | 4      |
| 9     | r4     |    |    | r4  |       |      |      |      |        |

## 2.3

Sim, toda gramática LALR(1) é LR(1)

## 2.4

Sim, não há conflitos durante a ação de análise e podemos perceber que nenhuma das regras necessárias para ser LALR(1) foi quebrada.

## 3 Questão 3

### 3.1

FIRST(S') : {a ,b}  
 FIRST(S) : {a, b}  
 FIRST(B) : {d}  
 FIRST(D) : {d}  
 FOLLOW(S') : {\$}  
 FOLLOW(S) : {\$}  
 FOLLOW(B) : {d, e}  
 FOLLOW(D) : {d, e}

Conjunto canônico LR(1):

- E0

```
{ [S' -> S, $];  
  [S -> a B d, $];  
  [S -> a D e, $];  
  [S -> b B e, $];  
  [S -> b D d, $] }
```

- E1

```
{ [S' -> S , $] }
```

- E2

```
{ [S -> a B d, $];  
  [S -> a D e, $];  
  [B -> d, d];  
  [D -> d, e] }
```

- E3

```
{ [S -> b B e, $];  
  [S -> b D d, $];  
  [B -> d, e];  
  [D -> d, d] }
```

- E4

```
{ [S -> a B d, $] }
```

- E5

```
{ [S -> a D e, $] }
```

- E6

```
{ [B -> d , d];  
  [D -> d , e] }
```

- E7

{ [S -> b B e, \$] }

- E8

{ [S -> b D d, \$] }

- E9

{ [B -> d , e];  
[D -> d , d] }

- E10

{ [S -> a B d , \$] }

- E11

{ [S -> a D e , \$] }

- E12

{ [S -> a D e , \$] }

- E13

{ [S -> b D d , \$] }

### 3.2

Conjunto canônico LALR(1):

- E0

{ [S' -> S, \$];  
[S -> a B d, \$];  
[S -> a D e, \$];  
[S -> b B e, \$];  
[S -> b D d, \$] }

- E1

{ [S' -> S , \$] }

- E2

{ [S -> a B d, \$];  
[S -> a D e, \$];  
[B -> d, d];  
[D -> d, e] }

- E3

{ [S -> b B e, \$];  
[S -> b D d, \$];  
[B -> d, e];  
[D -> d, d] }

- E4

{ [S -> a B d, \$] }

- E5

{ [S -> a D e, \$] }

- E6

{ [B -> d , d/e];  
[D -> d , e/e] }

- E7

{ [S -> b B e, \$] }

- E8

{ [S -> b D d, \$] }

- E9

{ [S -> a B d , \$] }

- E10

{ [S -> a D e , \$] }

- E11

{ [S -> b B e , \$] }

- E12

{ [S -> b D d , \$] }

### 3.3

| Tabela LALR(1) |        |         |         |    |     |       |   |   |   |
|----------------|--------|---------|---------|----|-----|-------|---|---|---|
| State          | Action |         |         |    |     | Go To |   |   |   |
|                | a      | d       | e       | b  | \$  | S'    | S | B | D |
| 0              | s2     |         |         | s3 |     |       | 1 |   |   |
| 1              |        |         |         |    | acc |       |   |   |   |
| 2              |        | s6      |         |    |     |       |   | 4 | 5 |
| 3              |        | s6      |         |    |     |       |   | 7 | 8 |
| 4              |        | s9      |         |    |     |       |   |   |   |
| 5              |        |         | s10     |    |     |       |   |   |   |
| 6              |        | r5 / r6 | r5 / r6 |    |     |       |   |   |   |
| 7              |        |         | s11     |    |     |       |   |   |   |
| 8              |        | s12     |         |    |     |       |   |   |   |
| 9              |        |         |         |    | r1  |       |   |   |   |
| 10             |        |         |         |    | r2  |       |   |   |   |
| 11             |        |         |         |    | r3  |       |   |   |   |
| 12             |        |         |         |    | r4  |       |   |   |   |

Com essa tabela é percebemos que a gramática não é LALR(1), pois existe uma ambiguidade reduce-reduce não resolvida no estado 6.

Pelo mesmo motivo, a gramática também não é SLR, afinal, SLR é um subconjunto de LALR(1).

## 4 Questão 4

### 4.1

FIRST(S') : {b, d}

FIRST(S) : {b,d}

FIRST(A) : {d}

FOLLOW(S') : {\$}

FOLLOW(S) : {\$}

FOLLOW(A) : {a,c}

Coleção LR(0):

- E0

```
{ S' -> S;
  S -> A a;
  S -> b A c;
  S -> d c;
  S -> b d a;
  A -> d }
```



- E1

{ S' -> S }

- E2

{ S -> A a }

- E3

{ S -> b A c;  
S -> b d a;  
A -> d }

- E4

{ S -> d c;  
A -> d }

- E5

{ S -> A a }

- E6

{ S -> b A c }

- E7

{ S -> b d a;  
A -> d }

- E8

{ S -> d c }

- E9

{ S -> b A c }

- E10

{ S -> b d a }

## 4.2

| Tabela SLR(1) |          |    |         |    |     |       |   |   |
|---------------|----------|----|---------|----|-----|-------|---|---|
| State         | Action   |    |         |    |     | Go To |   |   |
|               | a        | b  | c       | d  | \$  | S'    | S | A |
| 0             |          | s3 |         | s4 |     |       | 1 | 2 |
| 1             |          |    |         |    | acc |       |   |   |
| 2             | s5       |    |         |    |     |       |   |   |
| 3             |          |    |         | s7 |     |       |   | 6 |
| 4             | r5       |    | s8 / r5 |    |     |       |   |   |
| 5             |          |    |         | r1 |     |       |   |   |
| 6             |          |    | s9      |    |     |       |   |   |
| 7             | s10 / r5 |    | r5      |    |     |       |   |   |
| 8             |          |    |         | r3 |     |       |   |   |
| 9             |          |    |         | r2 |     |       |   |   |
| 10            |          |    |         | r4 |     |       |   |   |

A adição da produção  $S' \rightarrow S$  faz com que a gramática não seja mais SLR(1) devido às ambiguidades shift reduce dos estados 7 e 4.