Ugochi Madubata

10/17/2019

Compilers – HW2 – Context Free Grammers and LR Parsing

Problem 1

1.

|  |  |
| --- | --- |
| 1 | Sentential Form |
|  | S |
|  | cSdS |
|  | c∈dS |
|  | cdcSdS |
|  | cdc∈dS |
|  | cdcd∈ |
|  | cdcd |

|  |  |
| --- | --- |
| 1 | Sentential Form |
|  | S |
|  | cSdS |
|  | cdScSdS |
|  | cd∈cSdS |
|  | cdc∈dS |
|  | cdcd∈ |
|  | cdcd |

2.

|  |  |
| --- | --- |
| 1 | Sentential Form |
|  | S |
|  | cSdS |
|  | cSdcSdS |
|  | cSdcSd∈ |
|  | cSdc∈d |
|  | c∈dcd |
|  | cdcd |

|  |  |
| --- | --- |
| 1 | Sentential Form |
|  | S |
|  | cSdS |
|  | cSd∈ |
|  | cdScSd |
|  | cdSc∈d |
|  | cd∈cd |
|  | cdcd |

3. See attached

Problem 2

1.

|  |  |
| --- | --- |
| 1 | Sentential Form |
|  | S |
|  | (L) |
|  | (L, S) |
|  | (S, S) |
|  | (x, S) |
|  | (x, (L)) |
|  | (x, (L, S)) |
|  | (x, (S, S)) |
|  | (x, (x, S)) |
|  | (x, (x, x)) |

2.

|  |  |
| --- | --- |
| 1 | Sentential Form |
|  | S |
|  | (L) |
|  | (L, S) |
|  | (L, (L)) |
|  | (L, (L, S)) |
|  | (L, (L, x)) |
|  | (L, (S, x)) |
|  | (L, (x, x)) |
|  | (S, (x, x)) |
|  | (x, (x, x)) |

3.

Goal ::=S eof

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step | Parse Stack | Look Ahead | Unscanned | Parser Action |
| 0 | Empty | ( | x, x, x) | Shift |
| 1 | ( | x | , x, x) | Shift |
| 2 | (x | , | x, x) | Reduce by x ← S |
| 3 | (S | , | x, x) | Shift |
| 4 | (S, | x | , x) | Shift |
| 5 | (S, x | , | x) | Reduce by S ← L |
| 6 | (L, x | , | x) | Reduce by x ← S |
| 7 | (L, S | , | x) | Reduce by L,S ← L |
| 8 | (L | , | x) | Shift |
| 9 | (L, | x | ) | Shift |
| 10 | (L,x | ) | eof | Reduce by x ← S |
| 11 | (L,S | ) | eof | Reduce by L,S ← L |
| 12 | (L | ) | eof | Shift |
| 13 | (L) | eof |  | Reduce by (L) ← S |
| 14 | S | eof |  | Shift |
| 15 | S eof |  |  | Accept |

4.

Goal ::= S eof

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Step | Parse Stack | Look Ahead | Unscanned | Parser Action |
| 0 | Empty | ( | x, x, x) | Shift |
| 1 | ( | x | , x, x) | Shift |
| 2 | (x | , | x, x) | Reduce by x ← S |
| 3 | (S | , | x, x) | Shift |
| 4 | (S, | x | , x) | Shift |
| 5 | (S, x | , | x) | Reduce by x ← S |
| 6 | (S, S | , | x) | Reduce by L ← S |
| 7 | (S, L | , | x) | Reduce to S, L ← L |
| 8 | (L | , | x) | Shift |
| 9 | (L, | x | ) | Shift |
| 10 | (L, x | ) | eof | Reduce by x← S |
| 11 | (L, S | ) | eof | Shift |
| 12 | (L,S) | eof |  | Error |

It doesn’t have an effect on the depth of the stack if it waits until the last id variable to start using the right-recursive production. If it doesn’t and starts as soon as there is a second S, it leads to a stack that is smaller by one but an error occurs rather than an accept.

Problem 3

See attached