BU CS320 Assignment 5: Context Free Grammars

October 30, 2023

1. Given the following grammar where $\langle expr \rangle$ is the starting symbol

$$
\begin{array}{lll}
\langle digit \rangle & ::= & 0 \mid 1 \mid 2 \mid 3 \mid 4 \mid 5 \mid 6 \mid 7 \mid 8 \mid 9 \\
\langle nat \rangle & ::= & \langle digit \rangle \mid \langle digit \rangle \langle nat \rangle \\
\langle int \rangle & ::= & \langle nat \rangle \mid \text{-}\langle nat \rangle \\
\langle expr \rangle & ::= & \langle int \rangle \\
 & \mid & (\ \langle expr \rangle\ ) \\
 & \mid & \langle expr \rangle + \langle expr \rangle \\
 & \mid & \langle expr \rangle * \langle expr \rangle
\end{array}
$$

Derive the sentence using *rightmost derivation*.

12 + 2 * -07

$\langle expr \rangle = \langle expr \rangle * \langle expr \rangle$

$= \langle expr \rangle * \langle int \rangle$

$= \langle expr \rangle * - \langle nat \rangle$

$= \langle expr \rangle * - \langle digit \rangle \langle nat \rangle$

$= \langle expr \rangle * - \langle digit \rangle \langle digit \rangle$

$= \langle expr \rangle * - \langle digit \rangle 7$

$= \langle expr \rangle * - 07$

$= \langle expr \rangle + \langle expr \rangle * - 07$

$= \langle expr \rangle + \langle int \rangle * - 07$

$= \langle expr \rangle + \langle nat \rangle * - 07$

$= \langle expr \rangle + \langle digit \rangle * - 07$

$= \langle expr \rangle + 2 * - 07$

$= \langle int \rangle + 2 * - 07$

$= \langle nat \rangle + 2 * - 07$

$= \langle digit \rangle \langle nat \rangle + 2 * - 07$

$= \langle digit \rangle \langle digit \rangle + 2 * - 07$

$= \langle digit \rangle 2 + 2 * - 07$

$= 12 + 2 * - 07$

2. Given the following grammar where ⟨*stmt*⟩ is the starting symbol.

⟨*digit*⟩ ::= 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9

⟨*letter*⟩ ::= a | b | c | . . . | z

⟨*nat*⟩ ::= ⟨*digit*⟩ | ⟨*digit*⟩⟨*nat*⟩

⟨*int*⟩ ::= ⟨*nat*⟩ | -⟨*nat*⟩

⟨*expr*⟩ ::= ⟨*int*⟩
| (⟨*expr*⟩)
| ⟨*expr*⟩ + ⟨*expr*⟩
| ⟨*expr*⟩ * ⟨*expr*⟩

⟨*id*⟩ ::= ⟨*letter*⟩ | ⟨*letter*⟩⟨*id*⟩

⟨*stmt*⟩ ::= ⟨*id*⟩ = ⟨*expr*⟩
| for ⟨*id*⟩ = ⟨*expr*⟩ to ⟨*expr*⟩ do ⟨*stmt*⟩
| { ⟨*stmts*⟩ }
| pass

⟨*stmts*⟩ ::= ⟨*stmt*⟩ | ⟨*stmt*⟩ ; ⟨*stmts*⟩

Derive the sentence using *leftmost derivation*.

```
for x = -12 to 10 do { y = 0; pass }
```

=⟨stmt⟩
= for ⟨id⟩ = ⟨expr⟩ to ⟨expr⟩ do ⟨stmt⟩
= for ⟨letter⟩ = ⟨expr⟩ to ⟨expr⟩ do ⟨stmt⟩
= for x = ⟨expr⟩ to ⟨expr⟩ do ⟨stmt⟩
= for x = ⟨int⟩ to ⟨expr⟩ do ⟨stmt⟩
= for x = -⟨nat⟩ to ⟨expr⟩ do ⟨stmt⟩
= for x = -⟨digit⟩⟨nat⟩ to ⟨expr⟩ do ⟨stmt⟩
= for x = -1 ⟨nat⟩ to ⟨expr⟩ do ⟨stmt⟩
= for x = -1⟨digit⟩ to ⟨expr⟩ do ⟨stmt⟩
= for x = -12 to ⟨expr⟩ do ⟨stmt⟩
= for x = -12 to ⟨int⟩ do ⟨stmt⟩
= for x = -12 to ⟨nat⟩ do ⟨stmt⟩
= for x = -12 to ⟨digit⟩⟨nat⟩ do ⟨stmt⟩
= for x = -12 to 1⟨nat⟩ do ⟨stmt⟩
= for x = -12 to 1⟨digit⟩ do ⟨stmt⟩
= for x = -12 to 10 do ⟨stmt⟩

= for x = -12 to 10 do {⟨stmts⟩}
= for x = -12 to 10 do {⟨stmt⟩; ⟨stmts⟩}
= for x = -12 to 10 do {⟨id⟩ = ⟨expr⟩; ⟨stmts⟩}
= for x = -12 to 10 do {⟨letter⟩ = ⟨expr⟩; ⟨stmts⟩}
= for x = -12 to 10 do {y = ⟨expr⟩; ⟨stmts⟩}
= for x = -12 to 10 do {y = ⟨int⟩; ⟨stmts⟩}
= for x = -12 to 10 do {y = ⟨nat⟩; ⟨stmts⟩}
= for x = -12 to 10 do {y = ⟨digit⟩; ⟨stmts⟩}
= for x = -12 to 10 do {y = 0; ⟨stmts⟩}
= for x = -12 to 10 do {y = 0; ⟨stmt⟩}
= for x = -12 to 10 do {y = 0; pass}