\[ \text{trans}(s, x) = t \] (state)

\[ t \to \varnothing \]

\text{repeat}

\text{if } A \to a, X, A \text{ is in } S, \text{ where } A \text{ is a non-terminal, } a, b \text{ are symbols of grammar,}\n
\text{X is a proper symbol}\n
\text{add } A \to aX, \varnothing \text{ to } t

\text{// } A \to aX, \varnothing

\text{until all such items are added to } t

\text{return } \text{clause(t)}

\text{Start state:}

\[ S_0 = \text{Closure}( (S \to S) ) \]

where \( S' \) is the start symbol at the augmented grammar.

\[ S_0 \text{ is start state; } S \to E \]

\[ E \to E + T | T \]

\[ T \to T \times F | F \]

\[ F \to c \]

\[ F \to (E) \]

\[ S_0 \text{ =trans}(S_0, E): \]

\[ S \to E, \]

\[ E \to E + T \]

\[ S_0 \text{ =trans}(S_0, T): \]

\[ E \to T, \]

\[ T \to T \times F \]

\[ S_0 \text{ =trans}(S_0, F): \]

\[ T \to F, \]

\[ F \to c, \]

\[ F \to (E) \]

\[ S_0 \text{ =trans}(S_0, '('): \]

\[ F \to (E), \]

\[ E \to E + T \]

\[ E \to T, \]

\[ T \to T \times F \]

\[ F \to c, \]

\[ F \to (E) \]

\[ \text{etc.} \]

---

Construct the goto and action tables for the SLR(1) parser.

For every non-terminal \( A \) and state \( S \), define

\[ \text{goto}(S, A) = \text{trans}(S, A) \]

For every state containing the item \( S \to s \), (found on top of stack)

set action[\( S, \theta \)] = "accept"

For every state containing an item of the form \( A \to a \cdot b \)

where \( A \) is a non-terminal

\( b \) is a token

set action[\( S, b \)] = "shift t"

where \( t = \text{trans}(S, b) \)

\( t \) contains \( A \to a \cdot b, \varnothing \)

For every state \( S \) containing \( A \to a \cdot \),

set action[\( S, a \)] = "reduce \( A \to a \)"

for each token \( a \in \text{Follow}(A) \).