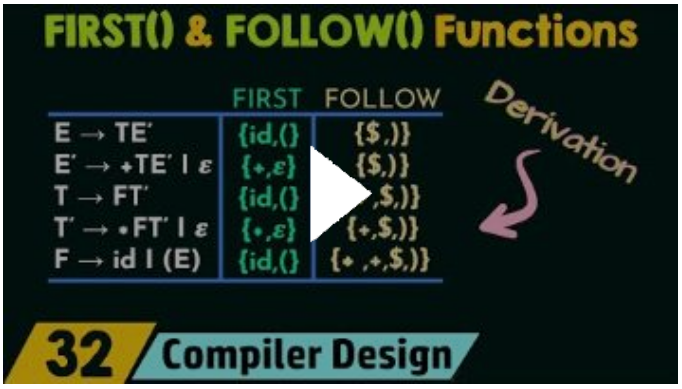
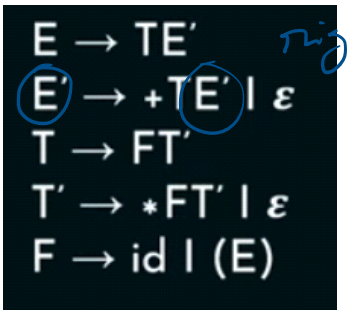
FIRST() and FOLLOW() functions

2 de junho de 2024 11:56

FIRST() and FOLLOW() Functions





Right most element

→ cannot have left recursion

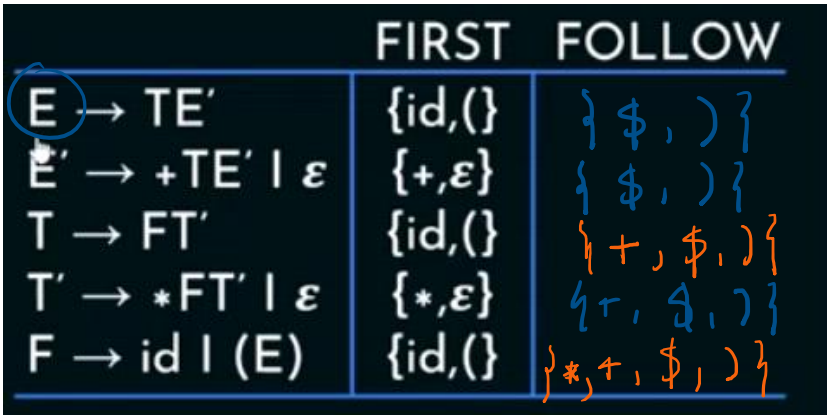
FIRST(F) = { id, (}

FIRST(T') = { *, ε }

FIRST(T) = FIRST(F) = { id, (}

FIRST(E') = { +, ε }

FIRST(E) = FIRST(T) = { id, (}



FOLLOW(E') = FOLLOW(E)

in *E -> TE'*, E' has nothing to follow
and since it is part and at the end of the production E,
it has the same follow as E

FOLLOW(E') = FOLLOW(E')
↓ right ↓ left

FOLLOW(T) = FIRST(E') ←

T is followed by E' in those 2 cases, therefore the symbols which can be generated from E' will have to be considered now
Ou seja, quando o follow é uma outra produção, usa-se o first dessa produção

= {+, ε}

3 casos:

1. The following terminal symbol will be selected as FOLLOW
   (in those two cases, T is not followed by terminal symbols)

2. The FIRST of the following non-terminal will be selected as FOLLOW
   (if the following non-terminal derives in epsilon, then there is no following non-terminal)

3. If it is the right most in the right-hand-side, the follow of the left-hand-side will be selected
   (in those two cases, if E' derives in epsilon, T becomes the right-most symbol in the right-hand-side. In that case, it will be selected the follow of the left-hand-side, E and E')

(Note: follow of any non-terminal will never include an epsilon)

FOLLOW(T') = FOLLOW(T') ∪ FOLLOW(T)
= FOLLOW(T)

— " —

FOLLOW(F) = FIRST(T')

= {*, ε}

= {*} ∪ FOLLOW(T) ∪ FOLLOW(T')