```
procedure firstSet( Xβ, set T )
X,β is valid sequence of grammar elements
(part of a rewrite's RHS), and T is an empty set on the
first call of procedure.
returns the set of terminals {t ∈ Σ$ | Xβ ⇒ tπ},
  and an updated set T

Recall notationally: Σ$ = Σ + {$}; X ∈ N ⋃ Σ$; β,π ∈ (N ⋃ Σ$)*;
P is the set of production rules of the grammar.

if ( X ∈ Σ$ ) then (
  return {X}, T
)

# F for the First Set
let F be a set
if ( X not in T ) then (
  add X to set T
  foreach ( p ∈ P with X on LHS of p ) do (
    let R ← RHS of p
    # I for ignorable
    G, I ← firstSet(R, T)
    F ← F ⋃ G
  )
)
if( derivesToLambda( X ) is true AND |β| > 0 ) then (
  G, I ← firstSet(β, T)
  F ← F ⋃ G
)

return F, T
```