**First and Follow Sets**

When I learned about first and follow sets at university I found them difficult to follow, so I have tried to rewrite the rules I was taught for creating them so that they would be easier to understand. I hope it helps :)

**Rules for First Sets**

1. If X is a terminal **then** First(X) is just X!
2. If there is a Production X → ε **then** add ε to first(X)
3. If there is a Production X → Y1Y2..Yk**then** add first(Y1Y2..Yk) to first(X)
4. First(Y1Y2..Yk) is **either**
   1. First(Y1) (if First(Y1) doesn't contain ε)
   2. **OR** (if First(Y1) does contain ε) then First (Y1Y2..Yk) is everything in First(Y1) <except for ε > as well as everything in First(Y2..Yk)
   3. If First(Y1) First(Y2)..First(Yk) all contain ε **then** add ε to First(Y1Y2..Yk) as well.

**Rules for Follow Sets**

1. First put $ (the end of input marker) in Follow(S) (S is the start symbol)
2. If there is a production A → aBb, (where a can be a whole string) **then** everything in FIRST(b) except for ε is placed in FOLLOW(B).
3. If there is a production A → aB, **then** everything in FOLLOW(A) is in FOLLOW(B)
4. If there is a production A → aBb, where FIRST(b) contains ε, **then** everything in FOLLOW(A) is in FOLLOW(B)

**Here an example for you to follow through.**

The Grammar

E → TE'

E' → +TE'

E' → ε

T → FT'

T' → \*FT'

T' → ε

F → (E)

F → id

|  |  |
| --- | --- |
| First Sets | Follow Sets |
| We Want to make First sets so first we list the sets we need  FIRST(E) = {}  FIRST(E') = {}  FIRST(T) = {}  FIRST(T') = {}  FIRST(F) = {}  First We apply rule 2 to T' → ε and E' → ε  FIRST(E) = {}  FIRST(E') = {ε}  FIRST(T) = {}  FIRST(T') = {ε}  FIRST(F) = {}  First We apply rule 3 to T' → \*FT' this rule tells us that we can add everything in First(\*FT') into First(T')  Since First(\*) useing rule 1 is \* we can add \* to First(T')  FIRST(E) = {}  FIRST(E') = {+,ε}  FIRST(T) = {}  FIRST(T') = {\*,ε}  FIRST(F) = {}  First We apply rule 3 to T' → \*FT' this rule tells us that we can add everything in First(\*FT') into First(T')  Since First(\*) useing rule 1 is \* we can add \* to First(T')  FIRST(E) = {}  FIRST(E') = {+,ε}  FIRST(T) = {}  FIRST(T') = {\*,ε}  FIRST(F) = {}  Two more productions begin with terminals F → (E) and F → id If we apply rule 3 to these we get...  FIRST(E) = {}  FIRST(E') = {+,ε}  FIRST(T) = {}  FIRST(T') = {\*,ε}  FIRST(F) = {'(',id}  Next we apply rule 3 to T → FT' once again this tells us that we can add First(FT') to First(T)  Since First(F) doesn't contain ε that means that First(FT') is just First(F)  FIRST(E) = {}  FIRST(E') = {+,ε}  FIRST(T) = {'(',id}  FIRST(T') = {\*,ε}  FIRST(F) = {'(',id}  Lastly we apply rule 3 to E → TE' once again this tells us that we can add First(TE') to First(E)  Since First(T) doesn't contain ε that means that First(TE') is just First(T)  FIRST(E) = {'(',id}  FIRST(E') = {+,ε}  FIRST(T) = {'(',id}  FIRST(T') = {\*,ε}  FIRST(F) = {'(',id}  Doing anything else doesn't change the sets so we are done! | We want to make Follow sets so first we list the sets we need  FOLLOW(E) = {}  FOLLOW(E') = {}  FOLLOW(T) ={}  FOLLOW(T') = {}  FOLLOW(F) = {}  The First thing we do is Add $ to the start Symbol 'E'  FOLLOW(E) = {$}  FOLLOW(E') = {}  FOLLOW(T) ={}  FOLLOW(T') = {}  FOLLOW(F) = {}  Next we apply rule 2 to E' →+TE' This says that everything in First(E') except forε should be in Follow(T)  FOLLOW(E) = {$}  FOLLOW(E') = {}  FOLLOW(T) ={+}  FOLLOW(T') = {}  FOLLOW(F) = {}  Next we apply rule 3 to E →TE' This says that we should add everything in Follow(E) into Follow(E')  FOLLOW(E) = {$}  FOLLOW(E') = {$}  FOLLOW(T) ={+}  FOLLOW(T') = {}  FOLLOW(F) = {}  Next we apply rule 3 to T → FT' This says that we should add everything in Follow(T) into Follow(T')  FOLLOW(E) = {$}  FOLLOW(E') = {$}  FOLLOW(T) ={+}  FOLLOW(T') = {+}  FOLLOW(F) = {}  Now we apply rule 2 to T' →\*FT' This says that everything in First(T') except for ε should be in Follow(F)  FOLLOW(E) = {$}  FOLLOW(E') = {$}  FOLLOW(T) ={+}  FOLLOW(T') = {+}  FOLLOW(F) = {\*}  Now we apply rule 2 to F → (E) This says that everything in First(')') should be in Follow(E)  FOLLOW(E) = {$,)}  FOLLOW(E') = {$}  FOLLOW(T) ={+}  FOLLOW(T') = {+}  FOLLOW(F) = {\*}  Next we apply rule 3 to E → TE' This says that we should add everything in Follow(E) into Follow(E')  FOLLOW(E) = {$,)}  FOLLOW(E') = {$,)}  FOLLOW(T) = {+}  FOLLOW(T') = {+}  FOLLOW(F) = {\*}  Next we apply rule 4 to E' → +TE' This says that we should add everything in Follow(E') into Follow(T) (because First(E') contains ε)  FOLLOW(E) = {$,)}  FOLLOW(E') = {$,)}  FOLLOW(T) = {+,$,)}  FOLLOW(T') = {+}  FOLLOW(F) = {\*}  Next we apply rule 3 to T → FT' This says that we should add everything in Follow(T) into Follow(T')  FOLLOW(E) = {$,)}  FOLLOW(E') = {$,)}  FOLLOW(T) = {+,$,)}  FOLLOW(T') = {+,$,)}  FOLLOW(F) = {\*}  Finaly we apply rule 4 to T' → \*FT' This says that we should add everything in Follow(T') into Follow(F)  FOLLOW(E) = {$,)}  FOLLOW(E') = {$,)}  FOLLOW(T) = {+,$,)}  FOLLOW(T') = {+,$,)}  FOLLOW(F) = {\*,+,$,)} |

Author:[James Brunskill](mailto:james@jambe.co.nz) ([jmb.nz](http://jmb.nz/))

i