**Grammar G’** (Numbered list created for referencing)

1. P → L
2. L →I M
3. M → ; L | ε
4. I →A | C | W
5. A → id := E
6. C → if E then L O endif
7. O → else L | ε
8. W → while E do L end
9. E →  E2 R
10. R → Op1 E2 R | ε
11. E2 → T K
12. K → Op2 E2 | ε
13. T → c | id
14. Op1  → < | = | !=
15. Op2  → + | -

|  |  |  |
| --- | --- | --- |
|  | FIRST | FOLLOW |
| P | { id, if, while } | { $ } |
| L | { id, if, while } | { $, else, endif, end } |
| M | { ε, ; } | { $, else, endif, end } |
| I | { id, if, while } | { $, ;, else, endif, end } |
| A | { id } | { $, ;, else, endif, end } |
| C | { if } | { $, ;, else, endif, end } |
| O | { ε, else } | { endif } |
| W | { while } | { $, ;, else, endif, end } |
| E | { c, id } | {$, ;, else, endif, do, then, end} |
| R | { ε, <, =, != } | {$, ;, else, endif, do, then <, =, !=, c, id, end} |
| E2 | { c, id } | {$, ;, else, endif, do, then, -, +,<, =, != , c, id, end} |
| K | { ε, +, - } | { $, ;, else, endif, do, then, -, +,<, =, != , c, id, end} |
| T | { c, id } | {$, ;, else, endif, do, then, -, +,<, =, != , c, id, end} |
| Op1 | { <, =, != } | { c, id } |
| Op2 | { +, - } | { c, id } |

P →L

L →I M

M → **;** L | 

I →A | C | W

A →**id :=** E

C →**if** E **then** L O **endif**

O →**else** L | 

W →**while** E **do** L **end**

E → E2 R

R →Op1 E2 R | 

E2 →T K

K →Op2 E2 |  

T →**c** | **id**

Op1  →**<** |**=** | **!=**

Op2  →**+** | **-**

**Building Follow Sets (notes)**

**Variables**

Follow(P) = Follow(L) #1 jumps back

Follow(M) = Follow(L) #3 jumps back

Follow(L) = Follow(O) #7 jumps back

Follow(O) = { endif } #6

Follow(P) = Follow(L) = Follow(M) = Follow(O) = { endif, end }

Follow(I) = { ; endif } #2: First(M) = {e, ;}; Also b/c of e->Follow(I)=Follow(L) ->looping

Follow(A) = Follow(C)= Follow(W) = Follow(I) = { ; endif } #4 ->looping

Follow(E) = { ; endif, do } #5 jumps back to Follow(A); Line 8 (do)

Follow(R) = Follow(E) = { ; endif, do } #9 jumps back

Follow(E2)={ <, =, !=, ; endif, do } #9, First(R) and First(R)=e->Follow(E2)=Follow(E)

Follow(K) = { <, =, !=, ; endif, do } Follow(K) = Follow(E2) -> looping due to e

Follow(T) = { +, -, <, =, !=, ; endif, do } #7, First(K), First(K), also e-> Follow(T) = Follow(E2)

Follow(Op1) = { c, id } #10, First(E2)={c, id}

Follow(Op2) = { c, id } #12, First(E2)={c, id}

**Terminals**

Follow(;) ={id, if, while} #3 First(L)

Follow(id)={:=, +, -, <, =, !=, ; endif} #5 and #13 looping to T

Follow(:=)={c, id} #5 First(E)

Follow(if) = {c, id} #6 First(E)

Follow(then)={id, if, while} #6 First(L)

Follow(endif)= {; endif} #6 jumps to Follow(C)

Follow(else)={id, if, while} #7 -> First(L)

Follow(while)={c,id} #8 -> First(E)

Follow(do) ={id, if, while} #8 -> First(L)

Follow(end)= {; endif} #8 loops to Follow(W)

Follow(c)= {+, -, <, =, !=, ; endif} #13 jumps to Follow(T)

Follow(<)=Follow(=)=Follow(!=) = {c,id} #14 jumps to Follow(Op1)

Follow(+)=Follow(-) = {c,id} #15 jumps to Follow(Op2)