Tulio Paulucci Olivieri - 101629230

**Extended Backus-Naur-Form**

start ::= begin, cmd\_var, cmd, end

begin ::= "begin"

end ::= "end"

cmd\_var ::= (tipo\_var, var, ponto\_virgula) [cmd\_var]

tipo\_var ::= "int" | "flut"

var ::= nome\_var [atribuicao] [virgula, var]

ponto\_virgula ::= ";"

cmd ::= cmd\_se | cmd\_enquanto| (atribuicao, ponto\_virgula) [cmd]

nome\_var ::= {letra}

atribuicao ::= "=" valor\_var

cmd\_se ::= se ,abre\_parenteses, condicao, fecha\_parenteses, abre\_chaves, cmd, fecha\_chaves

se ::= "se"

cmd\_enquanto ::= enquanto, abre\_parenteses ,condicao, fecha\_parenteses, abre\_chaves, cmd, fecha\_chaves

enquanto ::= "enquanto"

abre\_parenteses ::= "("

fecha\_parenteses ::= ")"

abre\_chaves ::= "{"

fecha\_chaves ::= "}"

valor\_var ::= [-] numero | nome\_var | (numero, operador\_aritmetico , numero) | (nome\_var , operador\_aritmetico , numero)

condicao ::= (nome\_var, operador\_relacional, nome\_var) | (nome\_var,

operador\_relacional, numero)

numero ::= {digitos}+ | ({digitos}+,".",{digitos}+)

digitos ::= 0-9

operador\_relacional ::= "<" | ">" | "==" | "!=" | ">=" | "<="

operador\_aritmetico ::= "+", "-", "\*", "/"

Letra ::= letra\_minuscula | letra\_maiscula

letra\_minuscula ::= a-z

letra\_maiuscula ::= A-Z

|  |  |  |
| --- | --- | --- |
| Expressão | First | Follow |
| start | first(begin) | {‘$’} |
| begin | {‘begin’} | firs(cmd\_var) |
| end | {‘end’} | {‘$’} |
| cmd\_var | first(tipo\_var) | first(cmd) |
| var | first(nome\_var) | first(ponto\_virgula) |
| ponto\_virgula | {‘;’} | first(cmd\_var) U first(cmd) |
| cmd | {‘se’, ‘enquanto’} U first(atribuicao) | first(end) U first(fecha\_chave) |
| tipo\_var | {int, flut, frase} | first(nome\_var) |
| nome\_var | first(letra) | first(atribuicao) U ‘,’ U first(ponto\_virgula) U first(operador\_relacional) U first(operador\_aritmetico) U first(fecha\_parenteses) |
| atribuicao | ‘=’ | first(ponto\_virgula) U ‘,’ |
| cmd\_se | {‘se’} | follow(cmd) |
| se | {‘se’} | {‘(‘} |
| cmd\_enquanto | {‘enquanto’} | follow(cmd) |
| enquanto | {‘enquanto’} | {‘(‘} |
| abre\_parenteses | {‘(‘} | first(condicao) |
| fecha\_parenteses | {‘)’} | first(abre\_chaves) |
| abre\_chaves | {‘{‘} | first(cmd) |
| fecha\_chaves | {‘}’} | first(cmd) U follow(cmd) |
| valor\_var | first(numero) | ‘,’ U first(ponto\_virgula) |
| condicao | first(nome\_var) | first(fecha\_parenteses) |
| numero | first(digitos) | first(operador\_aritmetico) U ‘,’ U ponto\_virgula |
| digitos | {0, 1, 2, 3, 4, 5, 6, 7, 8, 9} | first(digitos) |
| operador\_relacional | {< , > , == , != , >= , <=} | first(nome\_var) U first(numero) |
| operador\_aritmetico | {+, -, \*, /} |  |
| letra | first(letra\_minuscula) U first(letra\_maiuscula) | first(letra) U ‘,’ U ponto\_virgula |
| letra\_minuscula | {a, b, c, d, e, f, g, h, i, j, k, l,  m, n, o, p, q, r, s, t, u, v, w, x, y, z} | follow(letra) |
| letra\_maiuscula | {A, B, C, D, E, F, G, H, I, J,  K, L, M, N, O, P, Q, R, S, T, U, V, W, X, Y, Z} | follow(letra) |