Análisis First

first(Program) = {process}
first(Program') = {function}
first(Class) = {process}

first(Cont) = {definir id write read if for repeat while switch}
first(Cont') = {definir id write read if for repeat while switch ε}

first(Accon) = {definir id write read if for repeat while switch}

first(Condif) = {if}
first(Condif') = {else}

first(Func) = {function}

first(CycleWhile) = {while}
first(CycleRep) = {repeat}
first(CycleFor) = {for}

first(Multselec) = {switch}
first(Multselec') = {data_int ε}
first(Multselec'') = {data_int}

first(Contblo) = {id write read if for repeat while switch}
first(Contblo') = {id write read if for repeat while switch ε}

first(Acblo) = {id write read if for repeat while switch}

first(Contswi) = {id write read if for repeat while}
first(Contswi') = {id write read if for repeat while ε}

first(Acswi) = {id write read if for repeat while}

first(Exprelog) = {id data_int data_double}
first(Exprelog') = {and or not ε}

first(Log) = {id data_int data_double}

first(Expremath) = {data_int data_double fun_sqrt fun_abs fun_ln fun_exp fun_sen fun_cos fun_atan fun_trunc fun_round fun_rand}

first(Expremath’) = {plus minus mult div exp modulo}
first(Mathfunc) = {fun_sqrt fun_abs fun_ln fun_exp fun_sen fun_cos fun_atan fun_trunc fun_round fun_rand}
first(Mathfunc') = {fun_sqrt fun_abs fun_ln fun_exp fun_sen fun_cos fun_atan fun_trunc fun_round fun_rand}

first(Exprestring) = {data_string}
first(Exprestring') = {plus}

first(Expression) = {id}
first(Expression’) = {delim_lparen plus minus mult div exp modulo ε}

first(Typed) = {int real bool char string}

first(Simb) = {plus minus mult div exp modulo}

first(Valor) = {data_int data_double data_string true false}
first(Valornum) = {data_int data_double}
first(Valorlog) = {data_int data_double id}
first(Valorexp) = {data_int data_double id data_string}
first(Valorstring) = {id data_string}
first(Valorbool) = {true false}

first(Opeasig) = {less more same less_same more_same diff}
first(Opelog) = {and or not}

first(Asig) = {id}
first(Asig') = {data_int data_double data_string true false fun_sqrt fun_abs fun_ln fun_exp fun_sen fun_cos fun_atan fun_trunc fun_round fun_rand id}

first(Impr) = {write}
first(Impr') = {data_int data_double data_string true false id}

first(Printmul) = {delim_comma ε}
first(Printmul') = {id data_int data_double data_string true false}

first(Lect) = {read}
first(Defi) = {definir}
first(Usfun) = {delim_lparen}

first(Varmul) = {id}
first(Varmul') = {delim_comma ε}

Análisis Follow

follow(Program) = {eof}
follow(Program') = {eof}
follow(Class) = {function}

follow(Cont) = {delim_rkey}
follow(Cont') = {delim_rkey}

follow(Accon) = {definir id write read if for repeat while switch}

follow(Condif) = {definir id write read if for repeat while switch}
follow(Condif') = {else end_if}

follow(Func) = {function}

follow(CycleWhile) = {definir id write read if for repeat while switch}
follow(CycleRep) = {definir id write read if for repeat while switch}
follow(CycleFor) = {definir id write read if for repeat while switch}

follow(Multselec) = {definir id write read if for repeat while switch}
follow(Multselec') = {default}
follow(Multselec'') = {data_int}

follow(Contblo) = {delim_rkey}
follow(Contblo') = {delim_rkey}
follow(Acblo) = {id write read if for repeat while switch}

follow(Contswi) = {delim_rkey}
follow(Contswi') = {delim_rkey}
follow(Acswi) = {id write read if for repeat while}

follow(Exprelog) = {delim_rparen}
follow(Exprelog') = {delim_rparen}

follow(Log) = {and or not}

follow(Expremath) = {delim_line}
follow(Expremath') = {delim_line}
follow(Mathfunc) = {delim_line}
follow(Mathfunc') = {delim_lparen}

follow(Exprestring) = {delim_line}
follow(Exprestring') = {delim_line}

follow(Expression) = {delim_line}
follow(Expression') = {delim_line}

follow(Typed) = {delim_line}

follow(Simb) = {data_int data_double id}

follow(Valor) = {delim_line delim_comma}
follow(Valornum) = {less more same less_same more_same diff and or no plus minus mult div exp modulo delim_line}
follow(Valorlog) = {and or not delim_line}
follow(Valorexp) = {delim_line}
follow(Valorstring) = {delim_line}
follow(Valorbool) = {delim_line}

follow(Opeasig) = {id data_int data_double}
follow(Opelog) = {id data_int data_double}

follow(Asig) = {definir id write read if for repeat while switch}
follow(Asig') = {delim_line}

follow(Impr) = {definir id write read if for repeat while switch}
follow(Impr') = {delim_line}
follow(Printmul) = {delim_line}
follow(Printmul') = {delim_line}

follow(Lect) = {definir id write read if for repeat while switch}
follow(Defi) = {definir id write read if for repeat while switch}
follow(Usfun) = {delim_line}

follow(Varmul) = {delim_rparen}
follow(Varmul') = {delim_rparen}

Análisis Intersecciones

first(Program'->Func Program') = first(Func) = function
first(Program'->ε) = follow(Program') = eof
first(Cont'->Accon Cont') = first(Accon) = definir id write read if for repeat while switch
first(Cont'->Accon) = first(Accon) = delim_rkey
first(Accon->Defi) = first(Defi) = definir
first(Accon->Asig) = first(Asig) = id
first(Accon->Impr) = first(Impr) = write
first(Accon->Lect) = first(Lect) = read
first(Accon->Condif) = first(Condif) = if
first(Accon->Cyclefor) = first(Cyclefor) = for
first(Accon->Cyclerep) = first(Cyclerep) = repeat
first(Accon->Cyclewhile) = first(Cyclewhile) = while
first(Accon->Multselec) = first(Multselec) = switch
first(Condif'->else delim_lkey Cont delim_rkey end_if) = else
first(Condif'->end_if) = end_if
first(Multselec'->Multselec'' Multselec') = first(Multselec'') = data_int
first(Multselec'->ε) = follow(Multselec') = default
first(Contblo'->Acblo Contblo') = first(Acblo) = id write read if for repeat while switch
first(Contblo'->ε) = follow(Contblo') = delim_rkey
first(Acblo->Asig) = first(Asig) = id
first(Acblo->Impr) = first(Impr) = write
first(Acblo->Lect) = first(Lect) = read
first(Acblo->Condif) = first(Condif) = if
first(Acblo->Cyclefor) = first(Cyclefor) = for
first(Acblo->Cyclerep) = first(Cyclerep) = repeat
first(Acblo->Cyclewhile) = first(Cyclewhile) = while
first(Acblo->Multselec) = first(Multselec) = switch
first(Contswi'->Acswi Contswi') = first(Acswi) = id write read if for repeat while
first(Contswi'->ε) = follow(Contswi') = delim_rkey
first(Exprelog'->Opelog Log Exprelog') = first(Opelog) = and or not
first(Exprelog'->ε) = follow(Exprelog') = delim_rparen
first(Log->id Opeasig Valorlog) = id
first(Log->Valornum Opeasig Valorlog) = first(Valornum) = data_int data_double
first(Expremath->Valornum Expremath') = first(Valornum) = data_int data_double
first(Expremath->Mathfunc) = first(Mathfunc) = fun_sqrt fun_abs fun_ln fun_exp fun_sen fun_cos fun_atan fun_trunc fun_round fun_rand
first(Mathfunc'->fun_sqrt) = fun_sqrt
first(Mathfunc'->fun_abs) = fun_abs
first(Mathfunc'->fun_ln) = fun_ln

first(Mathfunc'->fun_exp) = fun_exp
first(Mathfunc'->fun_sen) = fun_sen
first(Mathfunc'->fun_cos) = fun_sqrt
first(Mathfunc'->fun_atan) = fun_atan
first(Mathfunc'->fun_trunc) = fun_trunc
first(Mathfunc'->fun_round) = fun_round
first(Mathfunc'->fun_rand) = fun_rand
first(Exprestring'->plus Valorstring) = plus
first(Exprestring'->ε) = follow(Exprestring') = delim_lparen
first(Expression'->Usfun) = first(Usfun) = delim_lparen
first(Expression'->Simb Valorexp) = first(Simb) = plus minus mult div exp modulo
first(Typed->int) = int
first(Typed->real) = real
first(Typed->bool) = bool
first(Typed->char) = char
first(Typed->string) = string
first(Simb->plus) = plus
first(Simb->minus) = minus
first(Simb->mult) = mult
first(Simb->div) = div
first(Simb->exp) = exp
first(Simb->modulo) = modulo
first(Valor->data_int) = data_int
first(Valor->data_double) = data_double
first(Valor->data_string) = data_string
first(Valor->true) = true
first(Valor->false) = false
first(Valornum->data_int) = data_int
first(Valornum->data_double) = data_double
first(Valorlog->id) = id
first(Valorlog->Valornum) = first(Valornum) = data_int data_double
first(Valorexp->id) = id
first(Valorexp->Valornum) = first(Valornum) = data_int data_double
first(Valorexp->data_string) = data_string
first(Valorstring->id) = id
first(Valorstring->data_string) = data_string
first(Valorbool->true) = true
first(Valorbool->false) = false
first(Opeasig->less) = less
first(Opeasig->more) = more

first(Opeasig->same) = same
first(Opeasig->less_same) = less_same
first(Opeasig->more_same) = more_same
first(Opeasig->diff) = diff
first(Opelog->and) = and
first(Opelog->or) = or
first(Opelog->not) = not
first(Asig'->Expremath) = first(Expremath) = data_int data_double fun_sqrt fun_abs fun_ln fun_exp fun_sen fun_cos fun_atan fun_trunc fun_round fun_rand
first(Asig'->Exprestring) = first(Exprestring) = data_string
first(Asig'->Valorbool) = first(Valorbool) = true false
first(Asig'->Expression) = first(Expression) = id
first(Impr'->Valor Printmul) = first(Valor) = data_int data_double data_string true false
first(Impr'->id Printmul) = id
first(Printmul->delim_comma Printmul' Printmul) = delim_comma
first(Printmul->ε) = follow(Printmul) = delim_line
first(Printmul'->id) = id
first(Printmul'->Valor) = first(Valor) = data_int data_double data_string true false
first(Usfun->delim_lparen Varmul delim_rparen) = delim_lparen
first(Usfun->ε) = follow(Usfun) = delim_line
first(Varmul'->delim_comma id Varmul') = delim_comma
first(Varmul'->ε) = follow(Varmul') = delim_rparen