

### Gramática da linguagem *PasC*

prog	→ “program” “id” body
body	→ decl-list “{” stmt-list “}”
decl-list	→ decl “;” decl-list   $\epsilon$
decl	→ type id-list
type	→ “num”   “char”
id-list	→ “id”   “id” “,” id-list
stmt-list	→ stmt “;” stmt-list   $\epsilon$
stmt	→ assign-stmt   if-stmt   while-stmt   read-stmt   write-stmt
assign-stmt	→ “id” “=” simple_expr
if-stmt	→ “if” “(“ condition “)” “{” stmt-list “}”   “if” “(“ condition “)” “{” stmt-list “}” “else” “{” stmt-list “}”
condition	→ expression
while-stmt	→ stmt-prefix “{” stmt-list “}”
stmt-prefix	→ “while” “(“ condition “)”
read-stmt	→ “read” “id”
write-stmt	→ “write” writable
writable	→ simple_expr   “literal”
expression	→ simple_expr   simple_expr relop simple_expr
simple_expr	→ term   simple_expr addop term
term	→ factor-a   term mulop factor-a
factor-a	→ factor   “not” factor
factor	→ “id”   constant   “(“ expression “)”
relop	→ “==”   “>”   “>=”   “<”   “<=”   “!=”
addop	→ “+”   “-”   “or”
mulop	→ “*”   “/”   “and”
constant	→ “num_const”   “char_const”

## Gramática Corrigida da linguagem PasC

prog	→ “program” “id” body 1
body	→ decl-list “{“ stmt-list “}” 2
decl-list	→ decl “;” decl-list 3   $\epsilon$ 4
decl	→ type id-list 5
type	→ “num” 6   “char” 7
id-list	→ “id” id-list’ 8
id-list’	→ “,” id-list 9   $\epsilon$ 10
stmt-list	→ stmt “;” stmt-list 11   $\epsilon$ 12
stmt	→ assign-stmt 13   if-stmt 14   while-stmt 15   read-stmt 16   write-stmt 17
assign-stmt	→ “id” “=” simple_expr 18
if-stmt	→ “if” “(“ condition “)” “{“ stmt-list “}” if-stmt’ 19
if-stmt’	→ “else” “{“ stmt-list “}” 20   $\epsilon$ 21
condition	→ expression 22
while-stmt	→ stmt-prefix “{“ stmt-list “}” 23
stmt-prefix	→ “while” “(“ condition “)” 24
read-stmt	→ “read” “id” 25
write-stmt	→ “write” writable 26
writable	→ simple_expr 27   “literal” 28
expression	→ simple_expr expression’ 29
expression’	→ relop simple_expr 30   $\epsilon$ 31
simple_expr	→ term simple_expr’ 32
simple_expr’	→ addop term simple_expr’ 33   $\epsilon$ 34
term	→ factor-a term’ 35
term’	→ mulop factor-a term’ 36   $\epsilon$ 37
factor-a	→ factor 38   “not” factor 39
factor	→ “id” 40   constant 41   “(“ expression “)” 42
relop	→ “==” 43   “>” 44   “>=” 45   “<” 46   “<=” 47   “!=” 48
addop	→ “+” 49   “-” 50   “or” 51
mulop	→ “*” 52   “/” 53   “and” 54
constant	→ “num_const” 55   “char_const” 56

### Cálculo FIRST e FOLLOW

	FRIST	FOLLOW
<b>prog</b>	“program”	\$
<b>body</b>	“num”, “char”, “{“	\$
<b>decl-list</b>	“num”, “char”, “ε”	“{“
<b>decl</b>	“num”, “char”	“;”
<b>type</b>	“num”, “char”	“id”
<b>id-list</b>	“id”	“;”
<b>id-list’</b>	“,”, “ε”	“;”
<b>stmt-list</b>	“id”, “if”, “while”, “read”, “write”, “ε”	“}”
<b>stmt</b>	“id”, “if”, “while”, “read”, “write”	“;”
<b>assign-stmt</b>	“id”	“;”
<b>if-stmt</b>	“if”	“;”
<b>if-stmt’</b>	“else”, “ε”	“;”
<b>condition</b>	“id”, “num_const”, “char_const”, “(“, “not”	)”
<b>while-stmt</b>	“while”	“;”
<b>stmt-prefix</b>	“while”	“{“
<b>read-stmt</b>	“read”	“;”
<b>write-stmt</b>	“write”	“;”
<b>writable</b>	“id”, “num_const”, “char_const”, “(“, “not”, “literal”	“;”
<b>expression</b>	“id”, “num_const”, “char_const”, “(“, “not”	)”
<b>expression’</b>	“==”, “>”, “>=”, “<”, “<=”, “!=”, “ε”	)”
<b>simple-expr</b>	“id”, “num_const”, “char_const”, “(“, “not”	“;”, “==”, “>”, “>=”, “<”, “<=”, “!=”, “)”
<b>simple-expr’</b>	“+”, “-”, “or”, “ε”	“;”, “==”, “>”, “>=”, “<”, “<=”, “!=”, “)”
<b>term</b>	“id”, “num_const”, “char_const”, “(“, “not”	“+”, “-”, “or”, “;”, “==”, “>”, “>=”, “<”, “<=”, “!=”, “)”
<b>term’</b>	“*”, “/”, “and”, “ε”	“+”, “-”, “or”, “;”, “==”, “>”, “>=”, “<”, “<=”, “!=”, “)”
<b>factor-a</b>	“id”, “num_const”, “char_const”, “(“, “not”	“*”, “/”, “and”, “+”, “-”, “or”, “;”, “==”, “>”, “>=”, “<”, “<=”, “!=”, “)”
<b>factor</b>	“id”, “num_const”, “char_const”, “(“	“*”, “/”, “and”, “+”, “-”, “or”, “;”, “==”, “>”, “>=”, “<”, “<=”, “!=”, “)”

		=", ")"
<b>relop</b>	"==", ">", ">=", "<", "<=", "!="	"id", "num_const", "char_const", "(", "not"
<b>addop</b>	"+", "-", "or"	"id", "num_const", "char_const", "(", "not"
<b>mulop</b>	"*", "/", "and"	"id", "num_const", "char_const", "(", "not"
<b>constant</b>	"num_const", "char_const"	"*", "/", "and", "+", "-", "or", ";", "==", ">", ">=", "<", "<=", "!=", =", ")"