

Grupo: Bruna Keila Oliveira Santos  
Eduardo Santos Santana Bispo  
Janaina Ferreira Santos  
Thalyta Costa Dos Santos

## Linguagem Golang

programa -> declaration |  
          programa declaration

declaration -> declarationvar “,” |  
              declarationfunc

declarationvar -> VAR declara |  
                  CONST declara |  
                  ID “=” expbinaria

declara -> ID tipo |  
          ID tipo “=” expbinaria

declarationfunc -> FUNC ID assinatura corpo

assinatura -> “(“ listaparam “)” tipo

listaparam -> ID tipo |  
              listaparam “,” ID tipo

corpo -> “{“ stms “}”

stms -> stm |  
      stm stms

stm -> listaexp “,” |  
      declarationvar “,” |  
      declarationvarshort “,” |  
      ifstm |  
      forstm |  
      returnstm

returnstm -> RETURN “,” |  
            RETURN expbinaria “,”

declarationvarshort -> ID “:=” expbinaria

“ifstm -> IF listaexp corpo |  
IF listaexp corpo ELSE corpo”

ifstm -> IF listaexp corpo ifstm2

ifstm2 -> ELSE corpo | ε

forstm -> FOR corpo |  
FOR listaexp corpo |  
FOR listaexp “;” listaexp “;” listaexp corpo

tipo -> array |  
INT |  
FLOAT |  
BOOL |  
STRING

array -> “[ listaexp ”] tipo |  
“[ listaexp ”] tipo “{ params ”}” |  
“[ listaexp ”]

listaexp -> expbinaria |  
expatribui

expbinaria -> terminal operadorbinario expbinaria |  
call

expatribui -> ID operadoratribuicao expbinaria

operadoratribuicao -> "=" | "+=" | "-=" | "\*=" | "/=" | "%=" | ":="

operadorbinario -> "+" | "\*" | "%" | "/" | "-" | ">" | "<" | ">=" | "<=" | "==" | "!=" | "&&" |  
"||"

terminal → INTEIRO |  
FLUTUANTE |  
ID |  
PALAVRA |  
TRUE |  
FALSE

call -> ID “(” params “)” |

ID “(“ “)”

params -> listaexp “,” params |  
listaexp