

**UNIVERSIDADE FEDERAL DE ALAGOAS
INSTITUTO DE COMPUTAÇÃO
CIÊNCIA DA COMPUTAÇÃO**



**UNIVERSIDADE FEDERAL
DE ALAGOAS**

Gramática - BFS

**Bruna Leal Torres Silva
Eirene de Oliveira Fireman
Samuel Lucas Vieira Lins Barbosa**

**Maceió - AL
20 de Dezembro de 2021**

Sumário

Sumário	1
Gramática livre de Contexto	2

1. Gramática livre de Contexto

S = Function S | 'function' 'main' '(' ')' '{' FunctionBody '}'

Function = 'function' FTipo '_id' '(' ParF ')' '{' FunctionBody '}'

FTipo = Tipo | 'void'

Tipo = 'int' | 'float' | 'char' | 'string' | 'bool'

ParF = LParF | ϵ

LParF = DecPar | LParF ',' Tipo PId

DecPar = Tipo '_id' | 'array' Tipo '_id' '[' Ea ']'

PId = 'ID' | 'ID' '[' ']' | 'ID' '[' Ea ']'

FunctionBody = DecVar FunctionBody |

DecVar = Tipo '_id' InId | 'array' Tipo '_id' '[' Ea ']' InArr

InId = '=' Ec | ϵ

InArr = '=' '[' LEc ']' | ϵ

LEc = Ec | LEc ',' Ec

===== (ANTIGO) =====

S = Function S | DeclD S | ϵ

Function = 'function' FunctionType FunctionName '(' FunctionParam ')' '{' IFunc '}' Function | ϵ

FunctionType = IDType | 'void'

FunctionName = 'ID' | PR_MAIN

FunctionParam = IDType ID | IDType ID ',' MPFParam IDType ID
FunctionParam | ϵ

MPFParam = IDType ID ',' | ϵ

IDType = 'int' | 'float' | 'char' | 'string' | 'bool'

ArrayDeclaration = 'array' DcID '[' Number ']' | 'array' DcID '[' Number ']'
ArrayAtribuition

ArrayAtribuition = '[' Ea | Ea ',' MPArray Ea ']'

MPArray = Ea ',' | ϵ

AritOperation = (ID | FunCall | Number) AritSymbols AritOperation | (ID | FunCall |
Number)

IFunc = DcID IFunc ';' | FunCall IFunc ';' | Loop IFunc | While IFunc | VarAtribuition
IFunc ';' | ArrayDeclaration IFunc ';' | ArrayAtribuition IFunc ';' | Coments IFunc | SysIn
IFunc ';' | SysOut IFunc ';' | StringConcat IFunc ';' | Return IFunc ';' | Condicionais
IFunc IFunc ';' | ϵ

StringConcat = ID '&' CT_SRING | ID '&' CT_CHAR

Condicionais = 'if' '(' Eb ')' '{ Instrucao '}' | 'elif' '(' Eb ')' '{ Instrucao '}' | 'else' '{
Instrucao '}'

FuncCall = FunctionName '(' FunctionParam ')'

DcID = IDType ID Atribuition | IDType ID ',' MPDcID ID | ϵ

MPDcID = ID ',' | ϵ

Atribuition = '=' Ec | ϵ

AritSymbols = '+' | '-' | '*' | '/' | '%'

VarAtribuition = ID Atribuition;

SysParam = ID SysParam | ID ',' ID SysParam | ϵ

SysIn = 'SysIn' '(' SysParam ')'

SysOut = 'SysOut' '(' SysParam ')'

Return = 'return' ID

Loop = 'for' '(' ID '=' Ea ',' Ea ')' '{ Instrucao }'

While = 'while' '(' Eb ')' '{ Instrucao }'

Ec = Ec 'OPR_CONC' Eb Eb

Eb = Eb 'PR_OR' Tb | Tb

Tb = Tb 'PR_AND' Fb | Fb

Fb = Fb 'OPR_REL' Ra | 'OPR_NOT' Fb | Ra

Ra = Ra 'OPR_REL' Ea | Ea

Ea = Ea 'OPR_ADD' Ta | Ea 'OPR_SUB' Ta | Ta

Ta = Ta 'OPR_MULT' Fa | Ta 'OPR_DIV' Fa | Fa

Fa = '(' Ec ')' | 'OPR_SUB' Fa | IdOuFunCham | 'CT_INT' | 'CT_FLOAT' | 'CT_BOOL'
| 'CT_STRING' | 'CT_CHAR'

OPR_REL = 'OPR_DIGUAL' | 'OPR_DIF' | 'OPR_MAIOR' | 'OPR_MENOR' |
'OP_MAIORIG' | 'OP_MENORIG'

2. Gramática LL(1)

S = 'function' DcFunction S

DcFunction = 'main' FunctionHeader | Type ID FunctionHeader

Type = void | float | char | string | bool

FunctionHeader = '(' Param ')' '{' Instructions '}'

Param = DcParam | DcParamArray | ϵ

DcParam = Type ID MultDcParam

MultDcParam = ',' DcParam | ',' DcParamArray | ϵ

DcParamArray = 'array' Type ID '[' ']' MultDcParam

Instructions = {DcVar|DcArr|Command|InOut|Return|ID AtrId} Instructions | ϵ

DcVar = Type DcVarAtr ';'

DcVarAtr = ID Atr DcVarAtrFat

Atr = '=' Ec MultAtr | ϵ

MultAtr = ',' DcVarAtr | ϵ

DcArr = 'array' Type ID '[' 'int' ']' DcArrAtr ';'

DcArrAtr = '=' '{' ArrAtr '}' | epsilon

ArrAtr = CT | ID MultArrAtr

MultArrAtr = ',' ArrAtr | epsilon

Command = if | while | for

Condicional = '(' Eb ')' '{' Instructions '}' Elif Else

If = 'if' Condicional

Elif = 'elif' Condicional | epsilon

Else = 'else' '{' Instructions '}' | epsilon

While = 'while' '(' Eb ')' '{' Instructions '}'

For = 'for' '('('Start Stop Increment')' '{' Instructions '}'

Start = ID '=' Ec

Stop = ',' Ec

Increment = ',' CT_INT | epsilon

InOut = SysIn | SysOut

SysIn = 'SysIn' '(' ParamIn ')' ';'

ParamIn = ID MultiParamIn

MultiParamIn = ',' ParamIn | epsilon

SysOut = 'SysOut' '(' ParamOut ')' ';'

ParamOut = '\" CT_STRING '\" MultiParamOut | ID MultiParamOut

MultiParamOut = '+' ParamOut | epsilon

Return = 'return' ParamReturn

ParamReturn = Ec ';' | ';'

AtrId = {Atr | FunctionCall} ID

FunctionCall = '(' ParamFunctionCall

ParamFunctionCall = ')' ';' | ParamFunction ')' ';'

ParamFunction = Ec MultiParamFunction

MultiParamFunction = ',' ParamFunction | epsilon

Ec = Eb EcLL

EcLL = 'OPR_CONC' Eb EcLL | ϵ

Eb = Tb EbLL

EbLL = 'OPR_OR' Tb EbLL | 'OPR_AND' Tb EbLL | ϵ

Tb = Ra TbLL

TbLL = 'OPR_NOT' Ra TbLL | ϵ

Ra = Rb RaLL

RaLL = Rel Rb RaLL | ϵ

Rb = Ea RbLL

RbLL = Comparison Ea RbLL | ϵ

Ea = Ta EaLL

EaLL = 'OPR_ADD' Ta EaLL | 'OPR_SUB' Ta EaLL | ϵ

Ta = Fa TaLL

TaLL = 'OPR_MULT' Fa TaLL | 'OPR_DIV' Fa TaLL | 'OPR_MOD' Fa TaLL | ϵ

Fa = '(' Ec ')' | 'id' IdFunCall | 'CT_INT' | 'CT_FLOAT' | 'PR_TRUE' | 'PR_FALSE' |
'PR_BOOL' | 'CT_CHAR' | 'CT_STR'

Equality = 'OPR_DIGUAL' | 'OPR_DIF'

Comparison = 'OPR_MAIOR' | 'OPR_MAIORIG' | 'OPR_MENOR' |
'OPR_MENORIG'