



Universidade Federal de Alagoas - UFAL

Instituto de computação - IC

Disciplina: Compiladores

Prof: Drº Alcino Dall Igna Junior

Especificação dos tokens da linguagem ENL

Alunos: Thiago Emmanuel G. Rodrigues
Erivaldo Lourenço Mariano

Maceió, 19 julho de 2019

Especificação dos tokens

A linguagem de Programação que será utilizada para a implementação dos analisadores léxicos e sintáticos será o **Python**.

Enumeração com as categorias de Tokens

```
class Category(Enum):
    """Operadores aritmeticos"""
    OP_ADD
    OP_SUB
    OP_MULT
    OP_DIV
    OP_RET
    OP_UNA

    """Operadores de comparacao"""
    OP_IGU
    OP_DIF
    OP_MAIGU
    OP_MEIGU
    OP_ME
    OP_MA

    """Operador de concatenação"""
    OP_CONC

    """Operadores Logicos"""
    OP_AND
    OP_OR
    OP_NOT

    """Atribuição"""
    ATRIBUICAO

    """Palavras Reservadas"""
    IF
    ELSE
    WHILE
```

FOR
BEGIN
FUNCTION
RETURN
VOID
READ
PUT
BREAK

""Tipos""

INT
FLOAT
CHAR
CCHAR
VECTOR
BOOL

""Leitura e impressão""

RP_INT
RP_FLOAT
RP_CHAR
RP_CCHAR
RP_VECTOR
RP_BOOL

""Simbolos especiais""

ABR_CH
FEC_CH
ABR_PAR
FEC_PAR
ABR_COC
FEC_COC
SEP_VIRG
SEP_P_VIRG
SIMPLE_ASP

ID
CONST
UNKNOWN

Expressões Regulares dos Lexemas

Operadores		Palavras reservadas		Delimitadores	
'+'	OP_ADD	'if'	IF	'{'	ABR_CH
'-'	OP_SUB	'else'	ELSE	'}'	FEC_CH
'*'	OP_MULT	'while'	WHILE	'('	ABR_PAR
'/'	OP_DIV	'for'	FOR	')'	FEC_PAR
'%'	OP_RET	'begin'	BEGIN	'['	ABR_COC
'~'	OP_UNA	'function'	PR_FUNCTION	']'	FEC_COC
'=='	OP_IGU	'return'	PR_RETURN	'.'	SEP_VIRG
'!='	OP_DIF	'void'	PR_VOID	','	SEP_P_VIRG
'<='	OP_MEIGU	'int'	TYPE_INT	'	SIMPLE_ASP
'>='	OP_MAIGU	'float'	TYPE_FLOAT		
'<'	OP_ME	'char'	TYPE_CHAR		
'>'	OP_MA	'cchar'	TYPE_CCHAR		
'&&'	OP_AND	'vector'	TYPE_VECTOR		
' '	OP_OR	'bool'	TYPE_BOOL		
'!'	OP_NOT	'true'	PR_TRUE		
'='	ATRIBUICAO	'false'	PR_FALSE		
'++'	OP_CONC	'break'	PR_BREAK		

Expressões Regulares auxiliares

Letras minúsculas	[‘a’ - ‘z’]
Letras maiúsculas	[‘A’ - ‘Z’]
Números	[‘0’ - ‘9’]
símbolos	“ ” “ . ” “ , “ “ : “ “ ; “ “ ! “ “ + “ “ - “ “ * “ “ \ “ “ / “ “ _ “ “ % “ “ & “ “ # “ “ = ” “ (“ “) “ “ [“ “] ” “ { “ “ } “ “ ~ “ “ ’ “ “ “ ” “ > ” “ < ”