

Sumário

Sumário	1
Linguagem utilizada	2
Enumeração das categorias dos tokens	2
Expressões regulares auxiliares	3
Categorias	3

Linguagem utilizada

Os analisadores léxico e sintático serão implementados em Python.

Enumeração das categorias dos tokens

```
from enum import Enum, auto

class TokenCategory(Enum):
    ID = auto()
    TYPEBOOL = auto()
    TYPECHAR = auto()
    TYPESTRING = auto()
    TYPEINT = auto()
    TYPEFLOAT = auto()
    OPASUM = auto()
    OPAMINUS = auto()
    OPADIV = auto()
    OPAMULT = auto()
    OPRLESS = auto()
    OPRGREAT = auto()
    OPREQ = auto()
    OPRLESSEQ = auto()
    OPRGREATEQ = auto()
    OPRNOTEQ = auto()
    ARRAYBEGIN = auto()
    ARRAYEND = auto()
    PARAMSBEGIN = auto()
    PARAMSEND = auto()
    BLOCKBEGIN = auto()
    BLOCKEND = auto()
    FCDEF = auto()
    FCRETURN = auto()
    LINEEND = auto()
```

SEPARATOR = auto()
 ASSIGN = auto()
 TYPEASSIGN = auto()
 FCTYPEVOID = auto()
 CONSTLINT = auto()
 CONSTLSTRING = auto()
 CONSTLBOOL = auto()
 CONSTLFLOAT = auto()
 CONSTLCHAR = auto()
 PRINT = auto()
 INPUT = auto()
 CONDIF = auto()
 CONDELSEIF = auto()
 CONDELSE = auto()
 CASTINGINT = auto()
 CASTINGFLOAT = auto()
 CASTINGSTRING = auto()
 CASTINGBOOLEAN = auto()
 CASTINGCHAR = auto()
 LOOPFOR = auto()
 LOOPWHILE = auto()
 VAR = auto()
 OPLAND = auto()
 OPLOR = auto()
 OPLNOT = auto()
 EOF = auto()
 UNDEFINED = auto()

Expressões regulares auxiliares

letra	[a-zA-Z]
dígito	[0-9]

Categorias

Categoria	Lexema
-----------	--------

ID	('{letra}')('{letra}' '{dígito}' '_'*)
TYPEBOOL	'boolean'
TYPECHAR	'char'
TYPESTRING	'string'
TYPEINT	'int'
TYPEFLOAT	'float'
OPASUM	'+'
OPAMINUS	'_'
OPADIV	'/'
OPAMULT	'*'
OPRLESS	'<'
OPRGREAT	'>'
OPREQQL	'=='
OPRLESSEQQL	'<='
OPRGREATEQL	'>='
OPRNOTEQL	'!='
ARRAYBEGIN	'['
ARRAYEND	']'
PARAMSBEGIN	'('
PARAMSEND)'
BLOCKBEGIN	'begin'
BLOCKEND	'end'
FCDEF	'def'
FCRETURN	'return'
LINEEND	','
SEPARATOR	','
ASSIGN	'='

TYPEASSIGN	‘.’
FCTYPEVOID	‘void’
CONSTLINT	{‘dígitos’}+
CONSTLSTRING	(“”)(({‘letras’} {‘dígitos’})+)(“”)
CONSTLBOOL	(‘true’ ‘false’)
CONSTLFLOAT	({‘dígitos’})+\.({‘dígitos’})+
CONSTLCHAR	(“”)(‘letras’ {‘dígitos’})(“”)
PRINT	‘print’
INPUT	‘input’
CONDIF	‘if’
CONDELSEIF	‘elseif’
CONDELSE	‘else’
CASTINGINT	‘Int’
CASTINGFLOAT	‘Float’
CASTINGSTRING	‘String’
CASTINGBOOLEAN	‘Boolean’
CASTINGCHAR	‘Char’
LOOPFOR	‘for’
LOOPWHILE	‘while’
VAR	‘var’
OPLAND	‘and’
OPLOR	‘or’
OPLNOT	‘not’
EOF	
UNDEFINED	