

Trabalho 1 - Análise Léxica

Rodrigo da Silva Faria

07 de Dezembro de 2021

1 Definição da gramática

A gramática a ser implementada está baseada na linguagem Python que pode ser encontrada em ”<https://docs.python.org/3/reference/grammar.html>”.

DÍGITO	[0-9]
TEXTO	[_A-Za-z0-9-][_A-Za-z0-9-]
ID	[a-Za-z-][_A-Za-z0-9-]

2 Especificação Flex

{DIGITO}+\.{DIGITO}*	Valor Real
"TEXTO"	Cadeia de caracteres entre aspas duplas representam Strings
ID	Identificador para possíveis funções
{DIGITO}*	Valor Inteiro
"#"({TEXTO} / .)*	Comentário de Linha
True False Return Print def break	Palavras Reservadas
= += -= = /= %=	Sinais De Atribuição
()	Parênteses
:	Início de função
,	Separador
{ } []	Símbolos Utilizados para dicionários e Listas
+ - * / %	Operadores Lógicos
if elif else	Operadores Condicionais
== !=	Operadores de Igualdade
> < >= <=	Operadores Comparativos
in not in	Palavras reservadas para operadores de Sequência
for while	Palavras reservadas para laço de repetição

3 Comentários sobre a implementação (tabela de símbolos)

Exemplo:

```
def soma: a = 2 b = 3 return a+b//
```

ID	Tipo	Categoria	Valor	Acesso
soma	proc	...		global
a	int	...	2	proc parâmetro
b	int	...	3	proc parâmetro

4 Destaque para a compilação / ambiente / linguagem

O trabalho foi implementado utilizando a Oracle VM disponibilizada pelo professor e executada pelo Windows 10. A pasta contém o arquivo "**main.lex**", o PDF do trabalho e um ReadMe.txt para auxiliar na execução.

5 Observação

Ainda não consegui tratar totalmente a questão de dar início a uma função, mas pela maneira que estou pensando em fazer será verificar se temos um ":" (símbolo que antecede uma função), seguido de uma quebra de linha e uma tabulação "t".