# Como utilizar o analisador léxico e sintático

## Sobre o analisador

A pasta do analisador possui a seguinte estrutura:

* .git
* doc
  + diagramas-de-transicao-jflap
  + diagramas-de-transicao-png
  + mp.docx
  + mp.pdf
  + mu.docx
  + mu.pdf
* src
  + core.py
  + lexico.py
  + sintatico.py
  + util.py
* .gitignore
* LICENSE
* README.md

Na pasta doc, está toda a parte de documentação. Os arquivos mp são referentes ao manual do programador, e os arquivos mu são referentes ao manual do usuário. A pasta diagramas-de-transicao-jflap contém os diagramas com extensão jflap. Na pasta diagramas-de-transicao-png estão os diagramas exportados para png.

Na pasta src, está o código do analisador. O arquivo core.py é o arquivo principal, e é ele que será executado ao utilizar o analisador. O arquivo lexico.py implementa a análise léxica, e o arquivo sintático.py implementa a análise sintática. Já o arquivo util.py, possui funções úteis, que são utilizadas pelos outros arquivos.

Ao executar o analisador, outros arquivos serão criados, log-lexico.txt e log-sintatico.txt, que contém as informações do log da análise.

Os arquivos .gitignore, LICENSE e README.md e a pasta .git são referentes ao git, o qual foi utilizado para versionamento do analisador.

## Analisando um arquivo

Para fazer a análise léxica de um arquivo, deve-se utilizar o seguinte comando (na pasta src):

python core.py -l <arquivo>

Para fazer a análise sintática de um arquivo, deve-se utilizar o seguinte comando (na pasta src):

python core.py -s <arquivo>

o parâmetro <arquivo> é o arquivo que contém o código em d+.

## Interpretando a saída

### Léxico

Ao utilizar a análise léxica, o arquivo log-lexico.txt será gerado. Esse arquivo possuirá uma estrutura parecida com essa:

PR\_MAIN | main

SP\_ABREPARENTESES | (

SP\_FECHAPARENTESES | )

PR\_WHILE | while

ID | a

OP\_IGUAL | =

LB\_BOOL | true

PR\_DO | do

PR\_PRINT | print

SP\_ABREPARENTESES | (

ID | a

SP\_FECHAPARENTESES | )

PR\_LOOP | loop

SP\_PONTOEVIRGULA | ;

Do lado esquerdo, é o token retornado, e do lado direito, é o lexema referente aquele token.

### Sintático

Ao utilizar a análise sintática, o arquivo log-sintatico.txt será gerado. Esse arquivo possuirá uma estrutura parecida com essa:

PR\_MAIN decl\_main OK

SP\_ABREPARENTESES decl\_main OK

SP\_FECHAPARENTESES decl\_main OK

PR\_IF com\_selecao OK

ID var OK

Cada linha possui três informações, separadas por um \t. A primeira, é o token que foi analisado. A segunda, é o nome da função (regra da gramática) que gerou o log. E a terceira, é se a análise está certa. Caso ocorra algum erro, Ao invés de “OK”, terá uma mensagem, no seguinte formato “Esperado %s. Recebido %s”.