



**Universidade Federal de Mato Grosso**

Instituto de Computação - IC

**Compiladores I:**  
*Trabalho Final*

**Docente:** Prof. Raphael de Souza Rosa Gomes

**Discente:** Luís Antônio da Silva Dourado - 201621901003

Cuiabá, 2021

## Conteúdo

1	Enunciado	3
2	Autômato do analisador léxico	3
3	Arquivo de entrada	4
4	Código fonte	4

# 1 Enunciado

Implementação da parte frontend do compilador, ou seja até a geração de código intermediário. Pode ser em qualquer linguagem. Não pode usar bibliotecas que façam a tokenização, nem as análises sintáticas e semânticas. Pode ser qualquer análise sintática, top-down ou bottom-up. Deverá estar em algum versionamento de código (github, gitlab, etc) Deverá ser entregue:

- Autômato do analisador léxico
- A gramática com as regras semânticas
- Código fonte do programa

O repositório do projeto pode ser acessado através do **link do Github**

## 2 Autômato do analisador léxico

O autômato relacionado a análise léxica foi implementado no software de código livre, **JFlap**<sup>1</sup> e o arquivo fonte está disponível no repositório do **Github** (*analisador-lexico.jff*) e a versão imagem, *analisador-lexico.png*.

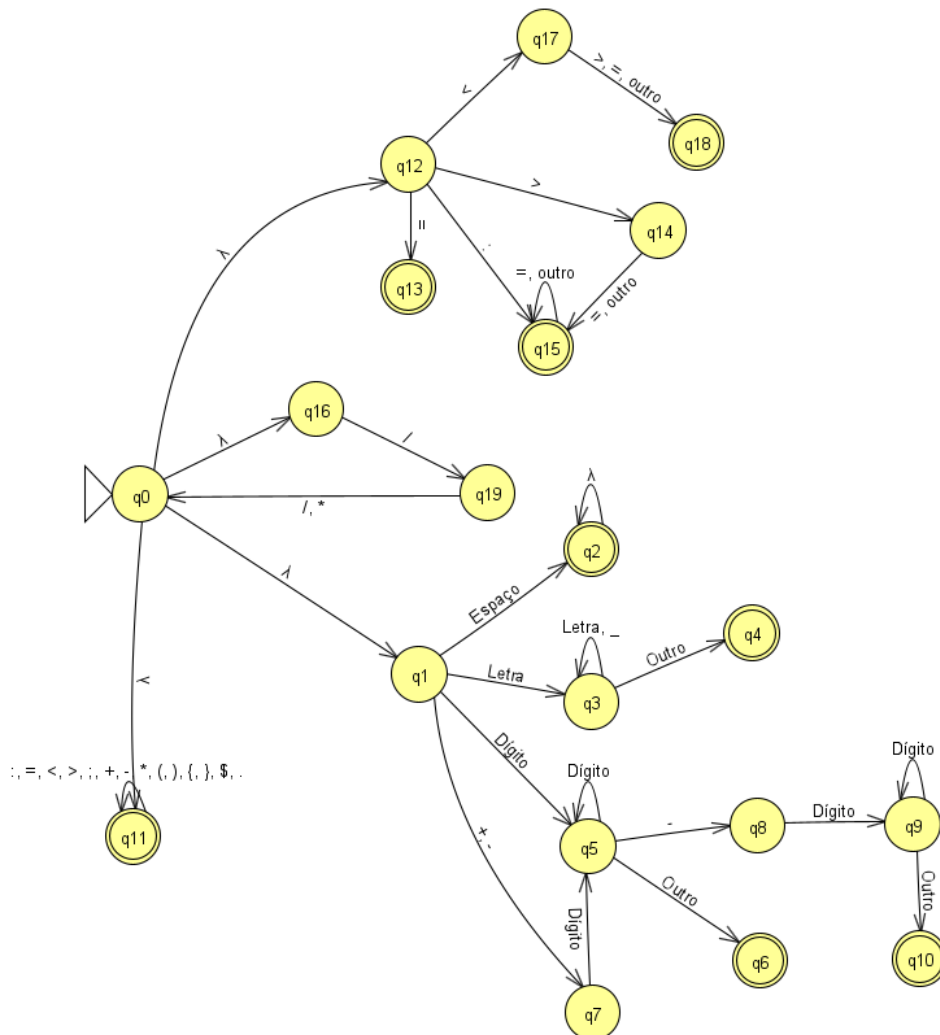


Figura 1: Autômato: Analisador Léxico

<sup>1</sup>JFlap: <https://www.jflap.org/>

### 3 Arquivo de entrada

O arquivo de entrada utilizado na implementação pode ser encontrado no arquivo *test.txt* no diretório do projeto.

```
program teste
    real: a,b;
    integer: c, d;
begin
    read(a);
    read(c);
    b := a * a + a;
    d := 2*c + c;
    if a > b then
        write(a)
    else
        write(b)
    $;
    if d < c then
        write(d)
    else
        write(c)
    $
end.
```

### 4 Código fonte

O código fonte do trabalho pode ser acessado através do **Github** (<https://github.com/luisdourado33/compiladores>)