

# Analizador Léxico

Lucas Garavaglia

UNIOESTE

3 de julho de 2021

# Conteúdo

- 1 Descrição da linguagem
- 2 Classe de Tokens
- 3 Implementação
- 4 Agradecimentos

# Descrição da Linguagem

Linguagem C:

- Compilada
- Procedural
- Estruturada

# Tokens

marcação: [{|}]

função: [(|)]

Loop: [while]

condição: [if]

final de função: [return]

tipo de dado: [int]

atribuição: [=]

Expressões: [>|=|<|=|!=|>|<]

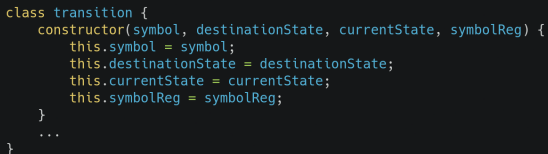
operadores: [=|-|\*|/|%]

final de linha: [;]

Variável: [\_|a-z|A-Z][\_|a-z|A-Z|0-9]\*

números: [0-9]+[[.][0-9]+]?

# Automatos



```
class transition {  
    constructor(symbol, destinationState, currentState, symbolReg) {  
        this.symbol = symbol;  
        this.destinationState = destinationState;  
        this.currentState = currentState;  
        this.symbolReg = symbolReg;  
    }  
    ...  
}
```

# Automatos

```
class state {
    constructor(name, isFinal) {
        this.isFinal = isFinal;
        this.name = name;
        this.transitions = [];
    }
    ...
    newTransition(symbol, destinationState, currentState, symbolReg = /[|/]) {
        this.transitions.push(
            new transition(symbol, destinationState, currentState, symbolReg)
        );
    }
    ...
    getNextState(symbol) {
        for (var i = 0; i < this.transitions.length; i++) {
            if (
                this.transitions[i].getSymbol() == symbol ||
                this.transitions[i].getSymbolReg().test(symbol)
            ) {
                return this.transitions[i].getState();
            }
        }
        return false;
    }
}
```

# Obrigado pela atenção!