

Equipe: Eric Klaus Brenner Melo e Santos, Matheus Rogério Pesarini, Ruan Rubino de Carvalho

Nome: Aurem

Variáveis

Tipos de dados suportados: int, float, string, bool

Regra de definição de variáveis: \$nome<tipo> = valor;

As variáveis podem conter quaisquer dígitos alfanuméricos em seu nome desde que não comece com números ou operadores lógicos, relacionais e aritméticos.

Exemplos possíveis:

```
$idade<int> = 20;
```

```
$nome_cliente <string> = “fulano”;
```

```
$_variavelPrivada<bool> = true;
```

Exemplos não permitidos:

```
$!teste<bool> = false;
```

```
$%=erro<float> = 0.5;
```

Operadores Suportados

Operadores Lógicos, Relacionais e Aritméticos: ==, !=, >=, >, <, <=, +=, -=, *=, /=, ^=, %, (), !

Entrada e Saída

print(): comando para a exibição de informações na linha de comando.

read(): comando para ler informações do teclado.

Exemplos:

```
print(“Olá, Mundo!”);
```

```
$idade<int> = read(“Qual é a sua idade? “);
```

Estruturas Suportadas

Seleção: if, else, else if

Laços: for, while

Exemplo de Código

```
$idade<int> = 25;
$altura<float> = 1.75;
$nome<string> = "Eric";
$ativo<bool> = true;
$lista<int[]> = {1, 2, 3, 4, 5};

$entrada<int> = read("Digite sua idade: ");

$idade += 1;
$altura *= 1.01;

if($entrada == $idade) {
    print("Mesma idade!");
} else if($entrada > $idade) {
    print("Você é mais velho.");
} else {
    print("Você é mais novo.");
}

if($idade > 30) {
    print("Idade não é maior que 30.");
}

if (idade % 2 == 0){
    print("Idade par");
}

for ($i<int> = 0; $i < 5; $i += 1) {
    print("for: " + $lista[$i]);
}

$i<int> = 0;
while ($i < 3) {
    print("while: " + $i);
    $i += 1;
}
```

Expressões regulares

A linguagem será capaz de interpretar campos:

naturais = [0-9]+

números = (+|-)? [0-9]+ (‘.’, naturais) ? (naturais*)

letras = [a-Z]

dígito=[0-9]

id = \$(letras | _) (letras | _ | dígito)*

tipos = { float | int | string | bool }

string = ["] [^\n] ["]

Comentário = >>>(~newline)*

[_] = (‘_’)

break = ‘break’

continue = ‘continue’

else = ‘else’

else if = ‘else if’

for = ‘for’

if = ‘if’

while = ‘while’

true = ‘true’

false = ‘false’

read = ‘read’

print = ‘print’

[\$] = ‘\$’

[‘ATRIBUICAO’] = ‘=’

[‘MENOR’] = [‘<’],

[‘MAIOR’] = [‘>’],

[‘INCREMENTO’] = [‘++’],

[‘DECREMENTO’] = [‘--’],

[IGUAL] = [‘==’],

[‘DIFERENTE’] = [‘!=’],

[‘MAIOR_IGUAL’] = [‘>=’],

[‘MENOR_IGUAL’] = [‘<=’],

[‘MAIS_IGUAL’] = [‘+=’],

[‘MENOS_IGUAL’] = [‘-=’],

[‘MULT_IGUAL’] = [‘*=’],

[‘DIV_IGUAL’] = [‘/=’],

[‘POT_IGUAL’] = [‘^=’],

[‘AND_LOGICO’] = [‘&&’],

[‘OR_LOGICO’] = [‘|||’],

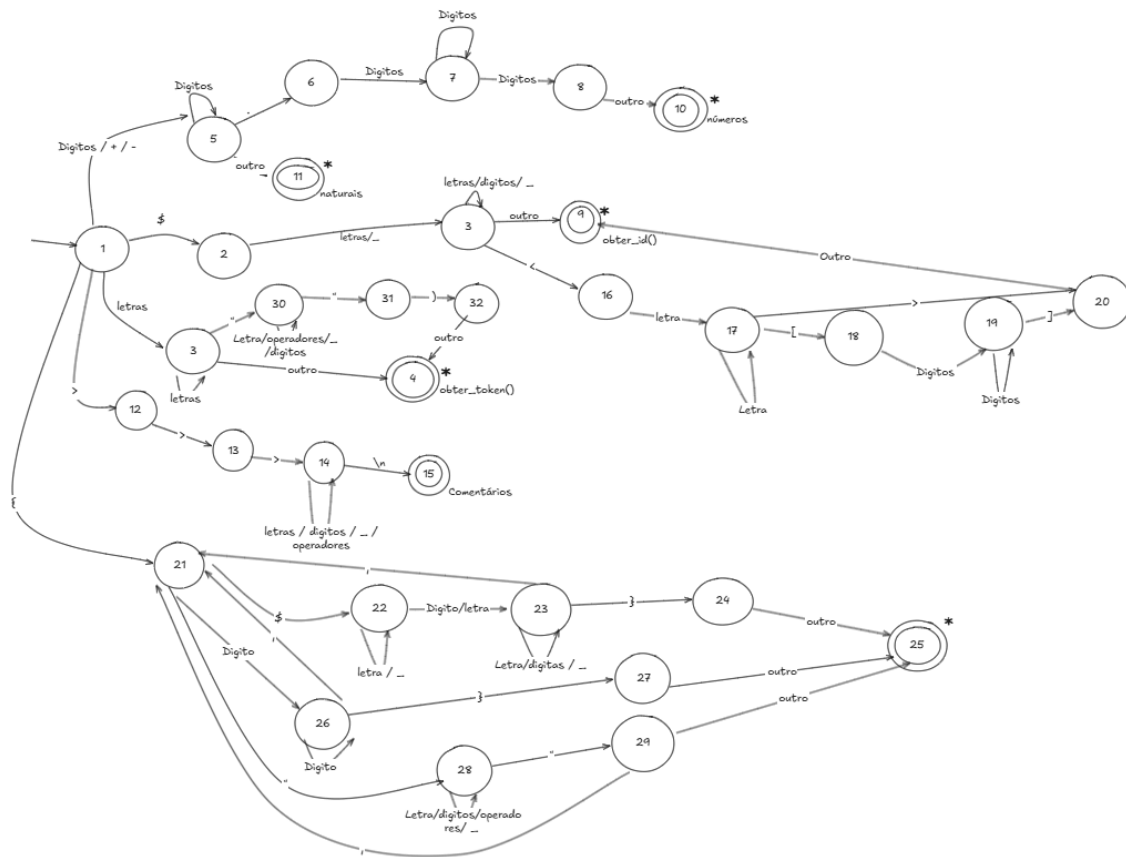
[‘MODULO’] = [‘%’],

[‘POTENCIA’] = [‘^’],

[‘NOT’] = [‘!’],

['MAIS'] = ['+'],
 ['MENOS'] = ['-'],
 ['MULT'] = ['*'],
 ['DIV'] = ['/'],
 ['PONTO_VIRGULA'] = ';',
 ['VIRGULA'] = ',',
 ['ABRE_PAREN'] = '(',
 ['FECHA_PAREN'] = ')',
 ['ABRE_COLCHETE'] = '[',
 ['FECHA_COLCHETE'] = ']',
 ['ABRE_CHAVE'] = '{',
 ['FECHA_CHAVE'] = '}'

Autômato da linguagem



Caso a imagem esteja pequena é possível acessar por [aqui](#).