






Alunos: Fernando Seiji Onoda Inomata, Lucas Batista Deinzer Duarte, Vitor Mayorca Camargo.

Nome da Linguagem: E-moji.

1. Caracterização da LP a ser Implementada

1.1 Tipos de Dados Suportados:


- **Inteiro:** Representado pelo token . Armazena valores numéricos inteiros;
- **String:** Representado pelo token . Armazena uma string de caracteres;
- **Booleano:** Representado pelo token . Pode assumir os valores  (Verdadeiro) ou  (Falso).

1.2 Operadores Suportados:



- **Operadores Aritméticos:**
 - **Adição:** Representado por ;
 - **Subtração:** Representado por ;
 - **Multiplicação:** Representado por ;
 - **Divisão:** Representado por ;
 - **Atribuição:** Representado por ;
- **Operadores Relacionais:**
 - **Maior que:** Representado por ;
 - **Menor que:** Representado por ;
 - **Igual a:** Representado por ;
- **Operadores Lógicos:**
 - **E Lógico/AND:** Representado por ;
 - **OU Lógico/OR:** Representado por ;
 - **Igual a:** Representado por ;

1.3 Formação dos Identificadores (Nomes de Variáveis e Funções):















Um identificador deve começar com uma letra (a-z, A-Z) e pode ser seguido por qualquer combinação de letras, números (0-9) ou o caractere de sublinhado (_).

- **Expressão Regular:** [a-zA-Z][a-zA-Z0-9_]*
- **Declaração de Variáveis:** <tipo> <nome_variavel>;
 - **Exemplo:**  int_a;

1.4 Comandos de Entrada e Saída:

- **Entrada (Scanf):** Lê um valor da entrada padrão e atribui ele à uma variável.
 -  (...)
- **Saída (Printf):** Exibe um valor ou texto na saída padrão.
 -  (...)

1.5 Palavras-reservadas:

Categoria	Lexema	Token	Descrição
Tipos de Dados		INT	Define uma variável do tipo inteiro.
		STRING	Define uma variável do tipo string.
		BOOL	Define uma variável do tipo booleano.
Estrutura de Decisão		IF	Inicia um bloco condicional.
		ELSEIF	Bloco condicional alternativo.
		ELSE	Bloco alternativo final para uma estrutura de decisão.
Estrutura de Repetição		WHILE	Inicia um laço de repetição condicional.
		FOR	Inicia um laço de repetição contado.
Valores Booleanos		TRUE	Representa o valor lógico verdadeiro.
		FALSE	Representa o valor lógico falso.
Delimitadores de Bloco		ABRIR_BLOCO	Inicia um escopo ou bloco de código ({}).
		FECHAR_BLOCO	Finaliza um escopo ou bloco de código ({}).
Comentários	 ... 	-	Delimitam uma seção de comentário a ser ignorada.

1.6 Estruturas Suportadas Pela Linguagem:

- **Estrutura de Decisão (if/else):**

🤔 (<condição>) 🗨️

[bloco de código]

🗨️ 🤔 (<condição>) 🗨️

[bloco de código]

🗨️ 🙅 🗨️

[bloco de código]

🗨️

- **Estrutura de Repetição Condicional (While):**

😞 (<condição>) 🗨️

[bloco de código]

🗨️

- **Estrutura de Repetição Contada (For):**

😮 (<inicialização>; <condição>; <incremento>) 🗨️

[bloco de código]

🗨️

1.7 Estrutura Geral do Programa (Sintaxe) com Exemplo:

O exemplo abaixo declara duas variáveis 'a' e 'b', atribui um valor a 'a', lê um valor para 'b', compara ambos, e printa o valor maior. Vale mencionar que toda linha deverá terminar com um ponto e vírgula (;).

😬 Exemplo de programa em E-moji! Eu sou um comentário 👁👁

 a;

 b;

a 🍳 10;

💋 ("Digite um valor para b:");

👂 (b);

😬 (a 🐔 b) 🗨

💋 ("O valor de a é maior!");



🗨 🙌 🗨




💋 ("O valor de b é maior ou igual!");

🗨

2. Definições Regulares e Autômato de Reconhecimento

2.1 Definições Regulares:

- **Letra:** letra = [a-zA-Z]
- **Dígito:** digito = [0-9]
- **Identificador (ID):** id = letra (letra | digito | _)*
- **Número Inteiro:** num_int = digito+
- **Strings:** string_literal = [""] [^\n] + ["]
- **Comentários:** comentario =  .* 
- **Espaço em branco:** ws = (' ' | '\t' | '\n' | '\r')+

As palavras-reservadas e operadores são representados por suas strings literais (ex: if = , soma = ) , conforme apresentado em 1.5. Exemplo: a expressão regular para o token de IF é .

2.2 Autômato Finito (AFD) de Reconhecimento:

