

CarLang – Linguagem de Alto Nível para MicrowaveVM

1. Sobre a linguagem CarLang

CarLang é uma linguagem de programação de domínio específico (DSL) criada para controlar uma máquina virtual baseada no modelo MicrowaveVM. A sintaxe é inspirada em C, com variáveis inteiros, condicionais e laços, além de comandos de direção, sensores e controle, voltados a um cenário simplificado de movimentação e leitura de estado do dispositivo alvo.

2. Visão geral da linguagem

CarLang é uma linguagem imperativa simples, contendo:

- Variáveis inteiros: `x = 10; tempo = 5;`
- Condicionais: `if (x > 5) { ... }`
- Laços: `while (x > 0) { ... }`
- Comandos específicos (`acelerar, frear, virar, parar, ler`)
- Expressões booleanas: `>, <, >=, <=, ==, !=` A linguagem foi projetada para ser minimalista, permitindo tradução direta para o assembly da MicrowaveVM.

3. Comandos de sensores e direção

A linguagem oferece comandos próprios, que são convertidos em instruções de baixo nível da MicrowaveVM.

Sensores:

- `ler(velocidade) -> var`
- `ler(combustivel) -> var`
- `ler(posicao) -> var`

Direção e controle:

- `acelerar(n)`
- `frear(n)`
- `virar(esquerda) / virar(direita)`
- `parar`

4. Flex, Bison e integração com MicrowaveVM

O compilador da CarLang utiliza ferramentas padrão:

- **Análise léxica (Flex):** Reconhece identificadores, números, operadores e palavras-chave da linguagem.
- **Análise sintática (Bison):** Implementa a gramática da linguagem e, através de ações nas regras, produz as instruções assembly da MicrowaveVM.
- **MicrowaveVM como máquina alvo:** Utiliza registradores TIME e POWER, instruções como SET,

DECJZ, INC, GOTO e HALT, e um modelo simples de execução baseado em contagem e saltos condicionais.

5. Exemplos de código CarLang

Exemplo 1 – Acelerar e parar

```
tempo = 5; forca = 2; acelerar(4); parar;
```

Exemplo 2 – Loop com freio

```
contador = 3; while (contador > 0) { frear(1); } parar;
```

Exemplo 3 – If com sensor e direção

```
vel = 10; if (vel > 5) { frear(2); } ler(velocidade) -> velAtual; virar(esquerda); parar;
```