



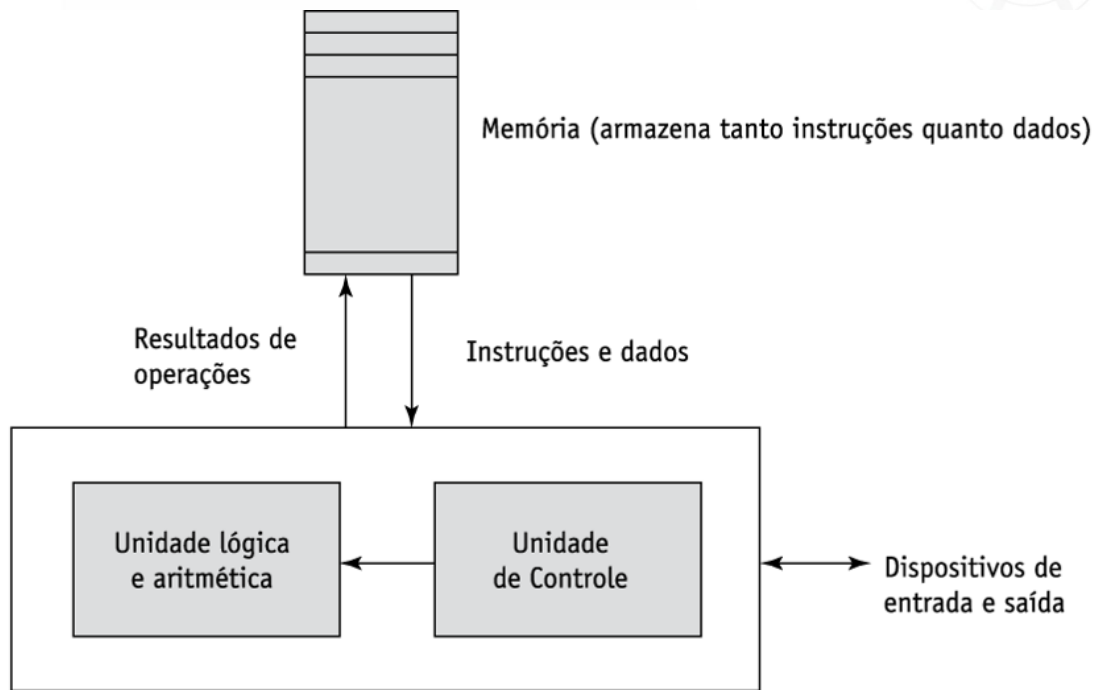
# Programação Imperativa

Kleber Jacques F. de Souza

# Programação Imperativa

- É o paradigma de programação mais antigo
- Descreve a computação como **ações**, enunciados ou comandos que mudam o **estado** (variáveis) de um programa.
- É baseado no modelo computacional clássico de **Von Neumann**

# Arquitetura de von Neumann



Unidade de processamento central

# Programação Imperativa

- Programas imperativos são uma **sequência de comandos** para o computador executar.
  - faça isso, depois isso, depois aquilo...
- No coração dessa arquitetura está a ideia de **atribuição**
  - Alterar o valor de um local da memória.

# Características

- Declaração de Variáveis
- Expressões
- Comandos (Sequenciais, Condicionais e de Repetição)
- Abstração Procedural

# Abstração Procedural

- O processo de **Abstração Procedural** permite ao programador se preocupar principalmente com a **interface entre a função e o que ela calcula**, ignorando os detalhes de como o cálculo é executado.
- Refinamento em Passos (Decomposição Funcional)

# Vantagens

- Eficiência (modelo Von Neumann)
- Paradigma dominante e bem estabelecido
- Modelagem natural de aplicações do mundo real
- É fácil de se entender, sendo amplamente usada em cursos introdutórios de programação.

# Desvantagens

- Possui difícil legibilidade
- Ainda se foca em **como** a tarefa deve ser feita e não em **o que** deve ser feito
- Tende a gerar códigos confusos, onde tratamento dos dados são misturados com o comportamento do programa



# Exemplos

- C
- Pascal
- Algol
- Cobol
- Fortran

# Referências Bibliográficas

SEBESTA, Robert W. **Conceitos de linguagens de programação**. Porto Alegre, Bookman, 2011.

TUCKER, Allen. **Linguagens de programação princípios e paradigmas**. Porto Alegre, AMGH, 2014.