

# O que torna uma linguagem imperativa?

- Programas escritos em linguagens de programação imperativas consistem em:
  - Um estado do programa;
  - Instruções que alteram o estado do programa
- Instruções do programa são "imperativas", no sentido gramatical dos verbos imperativos que expressam um comando.

**Primeiro faça isto, depois faça aquilo.**

# História das Linguagens Imperativas


- Primeiras linguagens imperativas: linguagens de montagem (assembly);
- 1954-1955: Fortran (**FOR**mula **TRAN**slator):
  - John Backus desenvolvido para IBM 704.
- Final de 1950: Algol (**ALGO**rithmic **L**anguage);
- 1958: Cobol (**C**ommon **B**usiness **O**riented **L**anguage), desenvolvido por uma comissão do governo, Grace Hopper muito influente.



# Elementos das linguagens imperativas

- Definições de tipo de dados;
- Declarações de variáveis:
  - São normalmente "tipadas" estaticamente ou dinamicamente:
    - Tipos básicos de dados (e.g., int, float, boolean, char);
    - Tipos de dados compostos (structs, arrays).
- Expressões e declarações de atribuição;
- Declarações de controle de fluxo (geralmente estruturados);
  - Desvios condicionais, laços;
  - As declarações são comandos e sua ordem é fundamental para a execução correta;
- Escopos lexicais e blocos:
  - Objetivo: proporcionar localidade de referência.
- Declarações e definições de procedimentos e funções (isto é, blocos parametrizados);
- Manipulação de erros e exceções;
- Comandos de E/S;
- Estruturas de dados.

# Características das linguagens imperativas

- São Turing completas se suportarem inteiros, operadores aritméticos básicos, atribuições, sequenciamento, laços e desvios condicionais;
- O estado do programa é mantido em variáveis associadas à localizações de memória:
  - Endereço e um valor de armazenamento;
- O valor da variável pode ser acessado direta e indiretamente, e pode ser alterado através de um comando de atribuição;
  - `a = 1;`
  - `&a = 1;`
- Uma atribuição introduz uma dependência de ordem no programa:
  - o valor de uma variável é diferente antes e depois de um comando de atribuição;
  - `print(a) => 1;`
  - `a = 2;`  dependência de ordem
  - `print(a) => 2;`



# Características das linguagens imperativas

- Repetição ou laço ou loop:
  - Usada extensivamente para processar valores ou acumular um valor em uma variável específica; ou
  - Usada para varrer uma sequência de localizações de memória tal como vetores.
- Funções:
  - Algoritmos que especificam como processar um intervalo de valores, a partir de um valor de domínio, com uma série de passos prescritos;
- Comandos:
  - Semelhantes às instruções de máquina nativas de um hardware de computador tradicional
    - o modelo de von Neumann-Eckley;
- O paradigma imperativo é predominante nas LPs, pois tais linguagens são mais fáceis de traduzir para uma forma adequada para execução na máquina;
- Usa-se abstração procedural e refinamento gradual para melhorar um programa imperativo.

# Gráfico de fluxo

- Usado para modelar programas imperativos;
- Com base nas três declarações de controle que são essenciais para se obter toda a capacidade de uma máquina de Turing;
- Precursor da UML e outras técnicas modernas;
- Criado para descrever fluxos de processos em geral.

