

# INTRODUÇÃO À PROGRAMAÇÃO

PROF<sup>a</sup>. M.Sc. JULIANA H Q BENACCHIO

# Linguagens de Programação

- O algoritmo auxilia o desenvolvimento do programas, pois é nele que os programadores vão representar a solução proposta para ser testada e avaliada
- Quando o algoritmo está finalizado, ou pronto, resta simplesmente transformá-lo em um programa computacional, respeitando toda lógica envolvida na solução testada e avaliada

# Linguagens de Programação

---

- Realmente, um bom algoritmo só precisará ser traduzido para a linguagem de programação desejada, respeitando as suas regras de sintaxe e semântica

# Linguagens de Programação

- O período de aprofundamento no conhecimento relacionado ao problema em questão, sua proposta de solução, os testes e a manutenção da mesma, ocorrem durante os estudos, pesquisas e elaboração do algoritmo restando assim, simplesmente a reescrita da solução encontrada para uma linguagem de programação computacional

# Linguagens de Programação

- As linguagens de programação, assim como as de comunicação, possuem regras sintáticas e semânticas que devem ser obedecidas, pois as instruções computacionais não podem oferecer possibilidades de interpretação para o computador, que somente executa o que lhe é solicitado

# Linguagens de Programação

- Existem várias linguagens de programação, mas em todas elas a lógica relacionada a proposta construída no algoritmo poderá ser implementada para atender ao objetivo almejado
  - Assembly      - Smalltalk      - Prolog      - C#
  - ALGOL          - Clipper        - B            - Java
  - BCPL           - Pascal         - Lua          - Visual BASIC
  - BASIC          - Ada            - Lisp         - C
  - COBOL         - Fortran        - C++         - PHP

# Processos de Tradução

- Um programa escrito por um programador em uma linguagem de programação de alto nível (chamado **código fonte**) é um conjunto de instruções que é clara para programadores, mas não para computadores.
- Ou seja, os computadores entendem única e exclusivamente suas linguagens nativas, as linguagens de máquina.
- Programas em linguagem de alto nível precisam ser traduzidos para linguagem de máquina para poderem ser submetidos ao computador e processados.

# Processos de Tradução

- Uma linguagem de programação pode ser convertida, ou traduzida, em código de máquina por compilação ou interpretada por um processo denominado interpretação. Em ambas ocorre a tradução do código fonte para código de máquina.
- Existem dois tipos fundamentais de tradutores:
  - Interpretadores
  - Compiladores

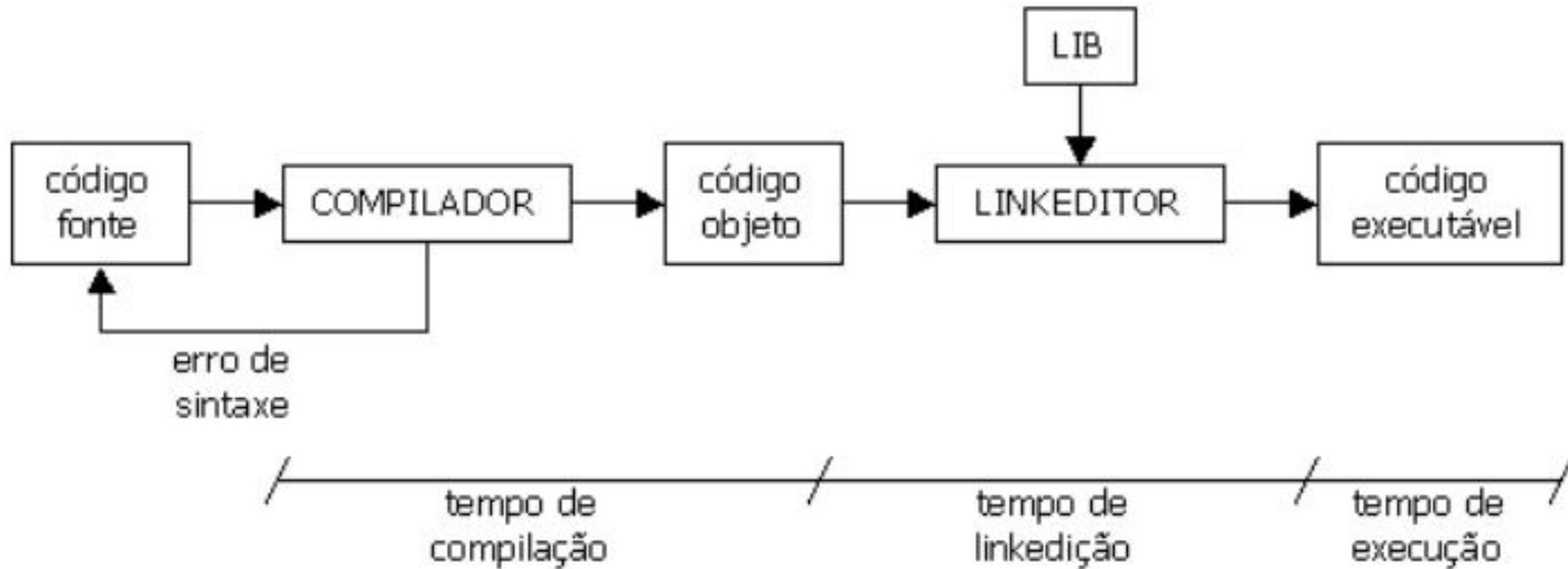
# Linguagens Compiladas

- São aquelas que se utilizam de um programa chamado **Compilador** para converter o código fonte em linguagem de máquina
- De forma simples, podemos dizer que os compiladores atuam como tradutores, que leem os comandos escritos em uma determinada linguagem de programação e os escrevem em linguagem de máquina.

# Linguagens Compiladas

- Neste processo, o código convertido fica gravado na forma de um programa executável (armazenado em um ou mais arquivos).
- Um programa compilado, portanto, pode ser executado inúmeras vezes sem que seja necessário realizar novamente o processo de compilação.
- Exemplos: Pascal, C, C++

# Linguagens Compiladas



# Linguagens Interpretadas

- Nas linguagens interpretadas, **não** existe a criação de um programa executável como ocorre nas linguagens compiladas.
- Nestas linguagens, os programas são diretamente executados pelos interpretadores, que vão convertendo os comandos de código fonte em linguagem de máquina, à medida em que o programa é executado

# Linguagens Interpretadas

- Como não é gerado um programa executável gravado em disco, a execução do programa é feita sempre pelo interpretador.
- Exemplos: BASIC, Perl, PHP,

# Linguagem C

- A linguagem C, desenvolvida por Dennis Ritchie, é uma linguagem de programação que tem sua origem em outras duas linguagens anteriores:
  - Linguagem B desenvolvida por Ken Thompson
  - Linguagem BCPL desenvolvida por Martin Richards

# Linguagem C



Ken Thompson e Dennis Ritchie (da esquerda para direita), os criadores das linguagens B e C, respectivamente

# Linguagem C

---

- A linguagem C foi criada por Dennis Ritchie, em 1972, no centro de Pesquisas da Bell Laboratories
- Sua primeira utilização importante foi a reescrita do Sistema Operacional UNIX, que até então era escrito em Assembly

# Linguagem C

- Em meados de 1970 o UNIX saiu do laboratório para ser liberado para as universidades
- Foi o suficiente para que o sucesso da linguagem atingisse proporções tais que, por volta de 1980, já existiam várias versões de compiladores C oferecidas por várias empresas, não sendo mais restrita apenas ao ambiente UNIX, porém compatíveis com vários outros sistemas operacionais

# Linguagem C

- O C é uma linguagem de propósito geral, sendo adequada à programação estruturada
- Pode ser considerada como uma linguagem de nível médio, pois possui instruções que a tornam ora uma linguagem de alto nível e estruturada como o Pascal, se assim se fizer necessário, ora uma linguagem de baixo nível pois possui instruções tão próximas da máquina, que só a linguagem Assembly possui

# Linguagem C

- Foi desenvolvida por programadores para programadores tendo como meta características de flexibilidade e portabilidade, pois não é “amarrada” a um sistema operacional ou a uma máquina particular
- Muito utilizada para escrever compiladores, analisadores léxicos, bancos de dados, editores de texto, etc.

- Suas principais características são:
  - Simples e de aprendizado fácil;
  - Uso amplamente difundido;
  - Baseada em construções simples, que usam os recursos da máquina de forma eficiente;
  - Oferece recursos de modularização necessários ao desenvolvimento de aplicações de grande porte;
  - Disponível em várias plataformas