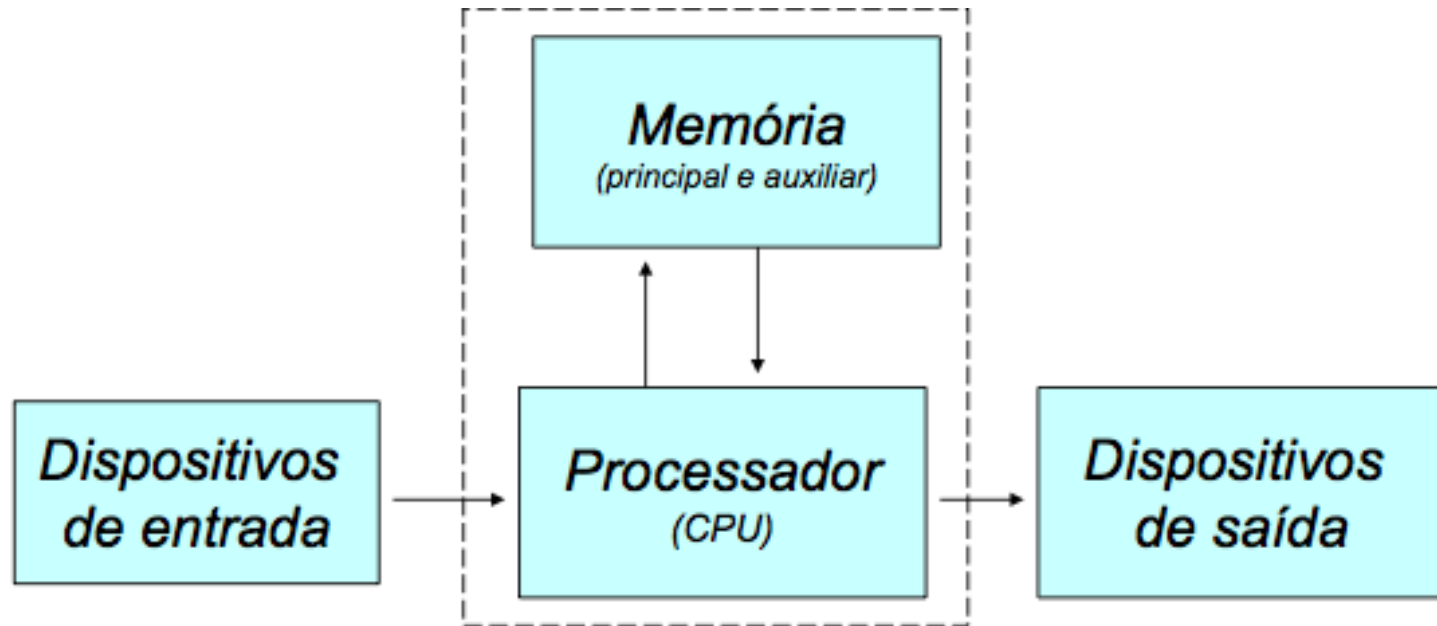


INTRODUÇÃO À PROGRAMAÇÃO

José Ricardo da Silva Junior

COMPONENTES DE UM COMPUTADOR



- **Processador (CPU):** Interpreta e executa as instruções
- **Memória (RAM (Random Access Memory), Disco):** Armazena os dados e as instruções
- **Dispositivos de entrada:** teclado, rato, leitor óptico, etc.
- **Dispositivos de saída:** monitor (ou ecrã), impressora, etc.



TIPOS DE MEMÓRIAS

- Central ou Principal (RAM)
 - Volátil (perde-se quando o computador é desligado)
 - Área de trabalho
 - Guarda temporariamente o programa e os dados, enquanto o programa está em execução
- Auxiliar ou secundária
 - Permanente
 - Guarda programas e dados/resultados
 - Inclui discos rígidos, pen-drives, CD-Rom's, etc.



CAPACIDADE DE MEMÓRIA

- Expressa no número de bytes que consegue armazenar

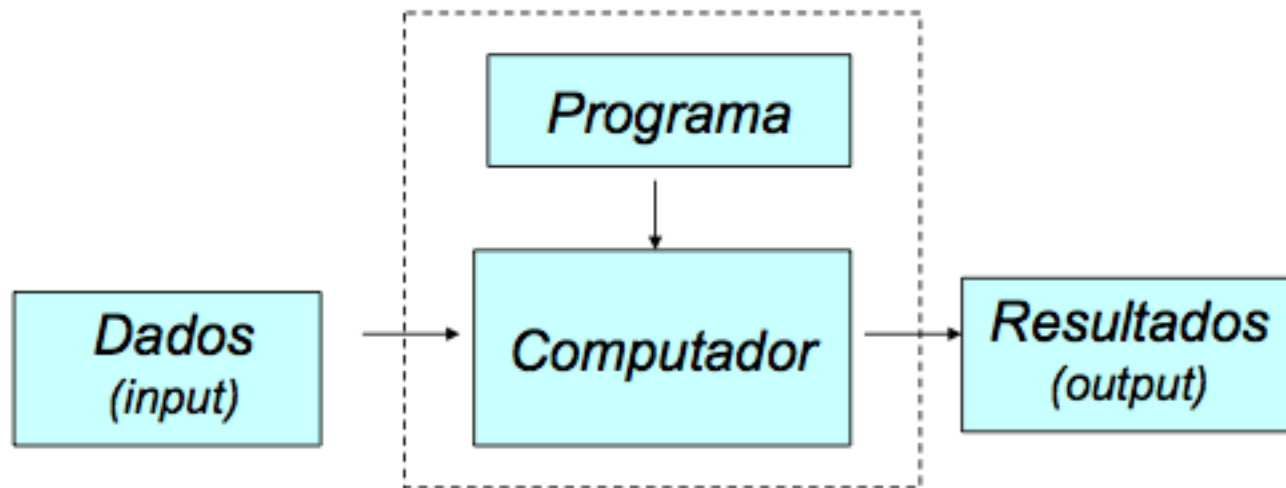
Unidade	Símbolo	No. De bytes
kilobyte	Kb	$2^{10}=1024$
megabyte	Mb	2^{20} (>1 milhão)
gigabyte	Gb	2^{30} (>1 bilhão)

- *Bit* é um dígito binário (0 ou 1)
- *Byte* é um conjunto de 8 bits



COMPONENTES COMPUTADOR

- **Software**: coleção de programas.
- **Hardware**: conjunto de componentes físicos
- **Programa**: conjunto de instruções, escritas em um código compreendida pelo computador que especificam como o trabalho deve ser executado



SISTEMA OPERACIONAL

- Conjunto de programas responsável pela interface entre o hardware do computador (disco, memória, unidade de processamento, mouse, teclado,...) e os outros programas (compiladores, editores de texto, programas de desenho, jogos, ...)
- Ambiente onde os utilizadores podem desenvolver seus programas e executá-los sem se preocupar com os detalhes de hardware específicos de cada máquina.
- Podemos dizer que é o programa “supervisor” de todas as operações do computador.



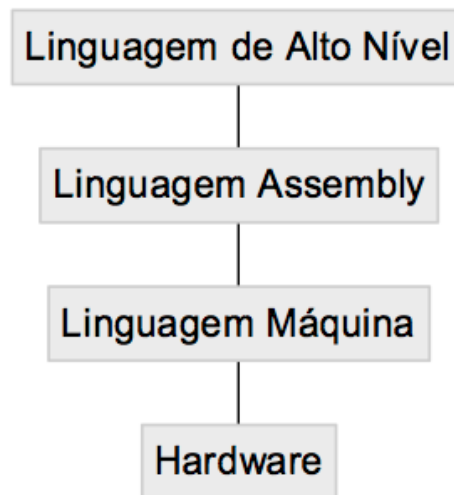
LINGUAGEM DE PROGRAMAÇÃO

- É um sistema formal de regras de descrição, com sintaxe e semântica bem definidas, capaz de representar um programa de computador.



CLASSIFICAÇÃO DAS LINGUAGENS

- Quanto ao grau de abstração:
 - Linguagem de programação de baixo nível
 - Assembly
 - Linguagem de programação de alto nível
 - Pascal, Fortran



LINGUAGEM DE MÁQUINA

- Menos natural para as pessoas
- Composta só com 0's e 1's
- Diretamente “entendida” pela máquina
- Não portátil (dependente do hardware)

Código de Máquina

111110100101001010010000000000001001000000001100



LINGUAGEM DE ALTO NÍVEL

- Mais próximas da linguagem natural
- Usam palavras, números e símbolos matemáticos
- O código é “portável” (independente do hardware)
- Não são diretamente “entendidas” pela máquina



LINGUAGEM ASSEMBLY

- A linguagem assembly foi desenvolvida para superar algumas das dificuldades da linguagem de máquina.
- Códigos e símbolos mnemônicos (fáceis de recordar) são utilizados para representar operações (como somar ou mover) e localizações de armazenamento.
 - Por exemplo, A significa "Add" (Somar) e L representa "Load" (Carregar).



LINGUAGEM ASSEMBLY

- A linguagem é a de mais baixo nível porque suas instruções correspondem às instruções em linguagem de máquina para um determinado computador e um microprocessador específico.
- Cada instrução em linguagem assembly corresponde a uma única instrução em linguagem de máquina.



LINGUAGEM ASSEMBLY

- Exemplo

Código de Máquina

11111010010100101001000000000001001000000001100

Código Assembly

AP TOTALA, VALORB



LINGUAGEM ASSEMBLY

- Embora a linguagem assembly seja mais fácil de usar do que a linguagem de máquina pura, ela ainda é extremamente difícil de aprender e exige programadores altamente capacitados.
- É uma linguagem utilizada principalmente para a escrita de software de sistemas operacionais, quando programas altamente detalhados e sensíveis à linguagem de máquina de um determinado computador têm de ser elaborados.



CLASSIFICAÇÃO

○ Quanto ao paradigma da sintaxe:

○ **Imperativo**

- Fortran, assembler
 - Procedimental
 - Pascal, C
 - Linguagem orientada a objetos
 - C++, Java, Python

○ **Declarativo**

- Funcional
 - Lisp, Haskell
- Lógico
 - Prolog



PARADIGMAS DE PROGRAMAÇÃO

○ Imperativo

- Permite descrever a resolução de um problema através de uma sequência de tarefas elementares (comandos) que definem o processo a seguir e permitem a mudança de estado de variáveis.

○ Orientado à Objetos

- O programa é organizado com base em objetos, que contêm não só as estruturas de dados mas também as funções que atuam sobre eles.
- A computação num sistema de objectos decorre da troca de mensagens entre os vários objetos, resultando na alterações nos seus estados internos e, conseqüentemente, no estado global do programa.



PARADIGMAS DE PROGRAMAÇÃO

○ Funcional

- Um programa é uma declaração de funções, que descrevem como computar um resultado a partir dos seus argumentos.
- As funções são aplicadas recursivamente ou por composição.
- A parte algorítmica e procedimental é (idealmente) suprimida.
- Modela-se apenas as formulações matemáticas da computabilidade.



PARADIGMAS DE PROGRAMAÇÃO

- **Lógico**
- Um programa é um conjunto de afirmações e regras que descrevem o conhecimento dos objetos e relações entre eles num determinado domínio.



LINGUAGENS DE PROGRAMAÇÃO

- Cada linguagem de programação foi projetada para resolver um determinado tipo de problema.
- É importante conhecer os pontos fortes e as limitações de cada uma dessas linguagens e de outras ferramentas de programação para poder escolher a linguagem mais apropriada para escrever o software.



CLASSIFICAÇÃO

- Quanto à estrutura dos tipos:
 - Fracamente tipada
 - Smaltalk
 - Fortemente tipada
 - Java, C
 - Dinamicamente tipada
 - Python



CLASSIFICAÇÃO

- Quanto ao modelo de execução:
 - **Interpretada:** linguagem onde o programa não é executado diretamente pelo SO ou processador.
 - **Compilada:** linguagem de programação onde o código fonte é executado diretamente pelo SO ou pelo processador, após ser traduzido, através de um processo chamado compilação.



CLASSIFICAÇÃO

- Interpretada



- Compilada

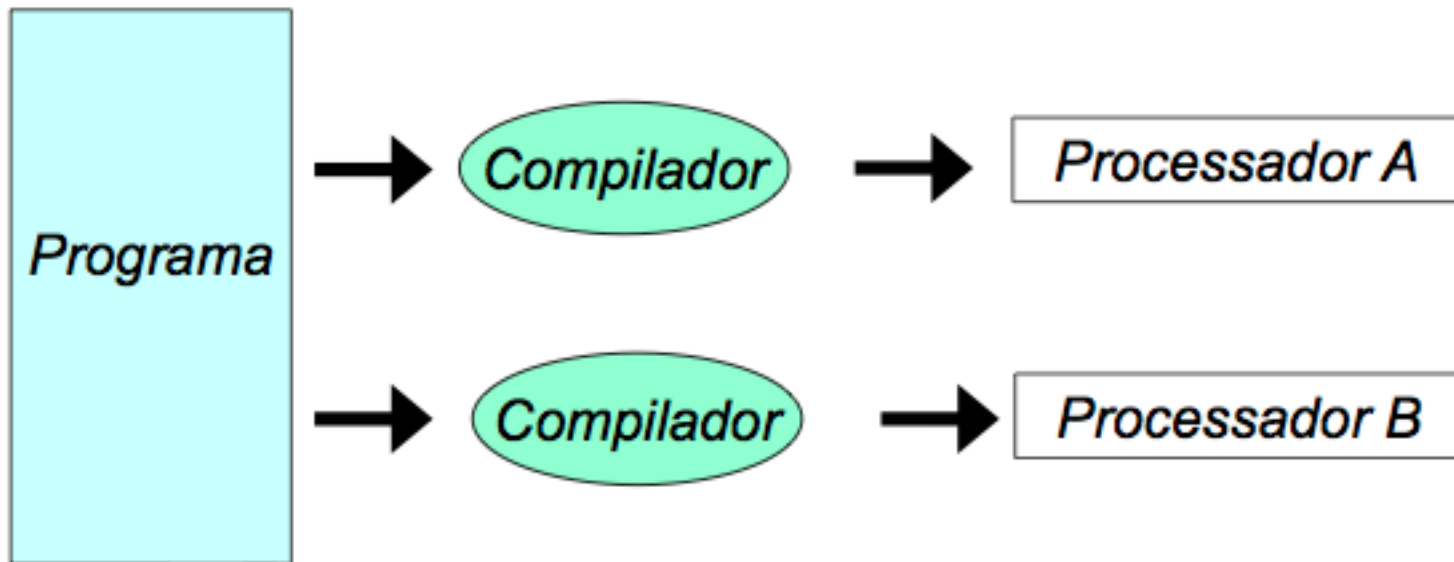


PROCESSO DE COMPILAÇÃO

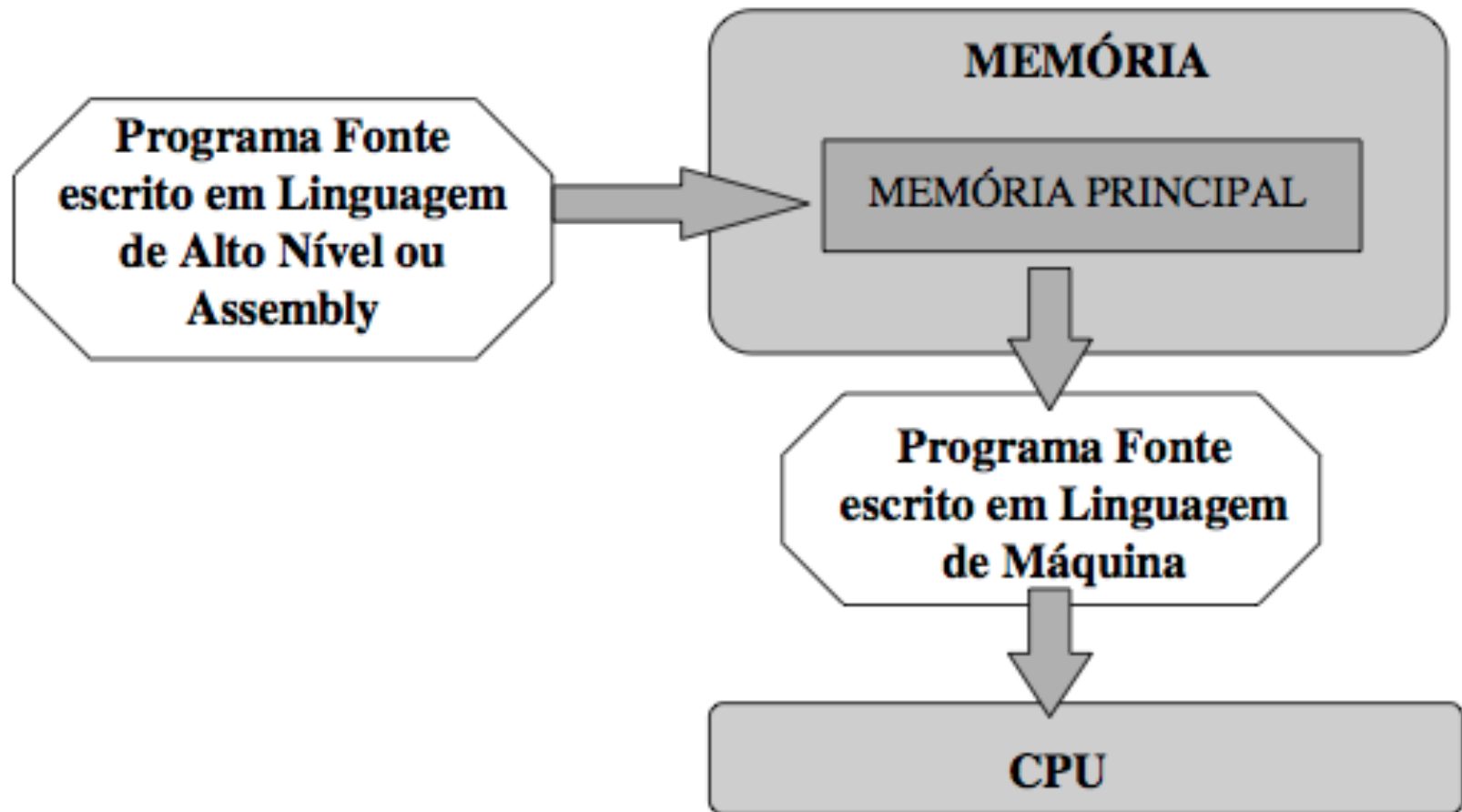
- **Compilador** é um programa especializado que verifica a correção sintática de um programa escrito numa linguagem de alto nível e o traduz para um programa escrito numa linguagem de baixo nível (i.e. a linguagem compreendida pelo computador que executa o programa).
- **Código fonte vs. Código objeto** – o programa escrito numa linguagem de alto nível diz-se código fonte e o programa escrito numa linguagem de baixo nível diz-se código objeto.
- **Compilar um programa** significa traduzir o código fonte para código objecto (sequências de 0's e 1's) através da execução do (programa) compilador.
 - Para fazer a tradução o compilador tem de entender a linguagem de programação e a linguagem máquina do microprocessador onde o programa vai ser executado.
 - Os compiladores tradicionais são específicos para uma linguagem de programação e um tipo de computador



PROCESSO DE COMPILAÇÃO



PROCESSO DE COMPILAÇÃO



LINGUAGEM C

- Desenvolvido sob o patrocínio da BellLinguagem CLLaboratories da AT&T no início dos anos 70
 - Desenvolvida para o sistema operacional UNIX
 - Inicialmente foi chamada de Linguagem B
 - O próprio UNIX foi desenvolvido em C.



LINGUAGEM C

- A Linguagem B mostrou-se muito limitada, prestando-se apenas para certas classes de problemas.
 - Fato sentido quando houve a necessidade de reescrever o UNIX. B era considerada lenta.
- Dennis Ritchie (Bell Labs) ficou encarregado de projetar uma nova linguagem, sucessora do B, que viria então, a ser chamada de Linguagem C.



LINGUAGEM C

- A linguagem C buscou manter o "contato com o computador real" e ainda sim dar ao programador novas condições para o desenvolvimento de programas em áreas diversas, como comercial, científica e de engenharia.
- Por 10 anos a sintaxe (o formato) da linguagem C seguia aquela fornecida com o UNIX versão 5.0 do Bell Labs.
 - A principal documentação deste padrão encontra-se na publicação "The C Programming Language", de Brian Kernighan e Dennis Ritchie (K&R), 1986, tida como a "bíblia da linguagem C".



LINGUAGEM C

- Característica interessante e importante era a portabilidade
 - Os programas-fonte criados para executar em um tipo de computador podiam ser transportados e recompilados em outros sem grandes problemas.
 - Com ela, uma empresa que desenvolve um programa pode fazê-lo executar em diferentes computadores sem ter um elevado custo cada vez que isto for feito.
- Em 1985, ANSI (*American National Standards Institute*) estabeleceu um padrão oficial de C o chamado "C ANSI".



EXERCÍCIOS

1. Qual a diferença entre linguagens de baixo nível e de alto nível? Como a linguagem C se encontra com relação a esse aspecto?
2. A que se refere o termo programação estruturada?

