



Estrutura de Dados

Ricardo José Cabeça de Souza

www.ricardojcsouza.com.br

ricardo.souza@ifpa.edu.br

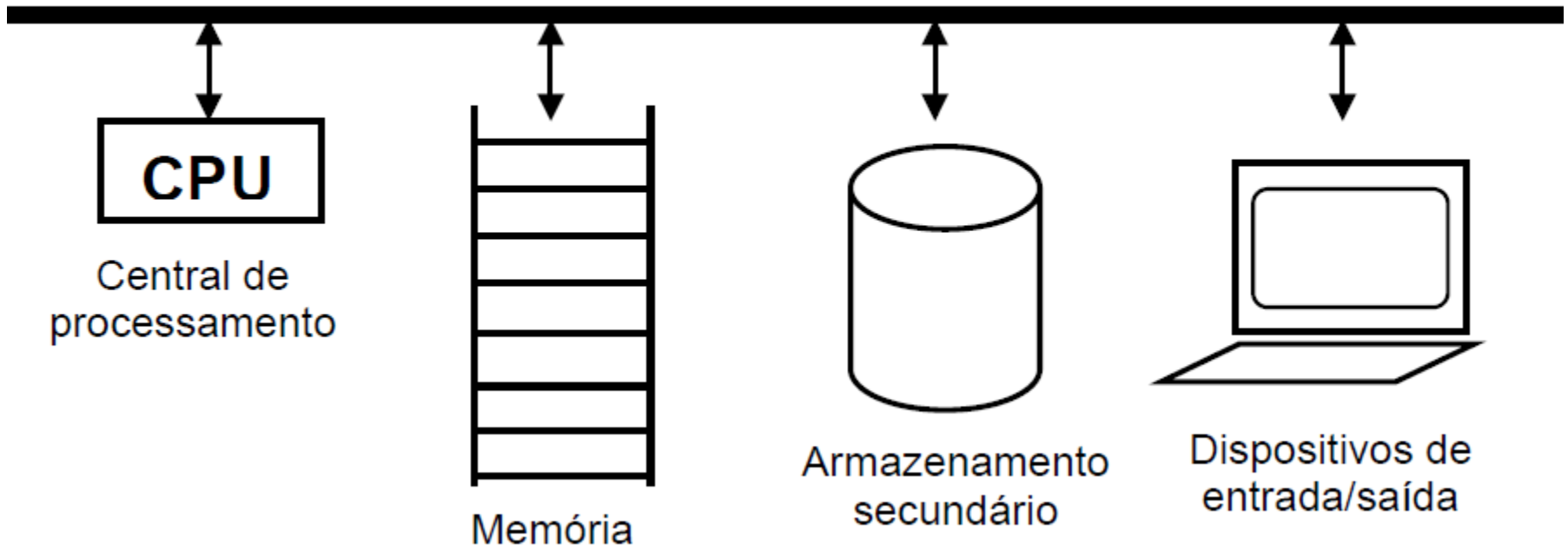
Parte 1



CONCEITOS

- **MODELO DE UM COMPUTADOR**

Canal de comunicação (*BUS*)





CONCEITOS

- **MODELO DE UM COMPUTADOR**
- **O canal de comunicação (conhecido como *BUS*)**
 - Meio para a transferência de dados entre os diversos componentes
- **Na memória principal**
 - São armazenados os programas e os dados no computador
 - Tem acesso randômico, o que significa se pode endereçar (isto é, acessar) diretamente qualquer posição da memória
 - Memória não é permanente(volátil)
 - Dados são armazenados enquanto o programa está sendo executado
 - Após o término do programa, a área ocupada na memória fica disponível para ser usada por outras aplicações



CONCEITOS

- **MODELO DE UM COMPUTADOR**
- **A área de armazenamento secundário**
 - Representada por um disco (disco rígido, pen driver, etc.)
 - Tem a vantagem de ser permanente
 - Tem um custo mais baixo do que a memória principal
 - Acesso aos dados é bem mais lento
- **Dispositivos de entrada e saída**
 - Dispositivos de entrada (por exemplo, teclado, *mouse*) permitem passarmos dados para um programa
 - Dispositivos de saída permitem que um programa exporte seus resultados, por exemplo em forma textual ou gráfica usando monitores ou impressoras



CONCEITOS

- **ARMAZENAMENTO DE DADOS E PROGRAMAS NA MEMÓRIA**
 - A memória do computador é dividida em unidades de armazenamento chamadas *bytes*
 - Cada byte é composto por 8 *bits*, que podem armazenar os valores zero ou um
 - Todas as informações (programas, textos, imagens, etc.) são armazenadas usando uma codificação numérica na forma binária
 - Na representação binária, os números são representados por uma sequência de zeros e uns
 - Por exemplo, o número decimal 5 é representado por 101, pois $1*2^2 + 0*2^1 + 1*2^0$ é igual a 5
 - Não é possível endereçar diretamente um bit



CONCEITOS

- **ARMAZENAMENTO DE DADOS E PROGRAMAS NA MEMÓRIA**
 - Para ser possível armazenar uma sequência de caracteres, que representa o texto, atribui-se a cada caractere um código numérico
 - Por exemplo, pode-se associar ao caractere 'A' o código 65, ao caractere 'B' o código 66, e assim por diante
 - Se todos os caracteres tiverem códigos associados (inclusive os caracteres de pontuação e de formatação), podemos armazenar um texto na memória do computador como uma sequência de códigos numéricos



CONCEITOS

- **ARMAZENAMENTO DE DADOS E PROGRAMAS NA MEMÓRIA**
 - Um computador só pode executar programas em linguagens de máquina
 - Cada programa executável é uma sequência de instruções que o processador central interpreta, executando as operações correspondentes
 - Esta sequência de instruções também é representada como uma sequência de códigos numéricos
 - Os programas ficam armazenados em disco e, para serem executados pelo computador, devem ser carregados (transferidos) para a memória principal
 - Uma vez na memória, o computador executa a sequência de operações correspondente



CONCEITOS

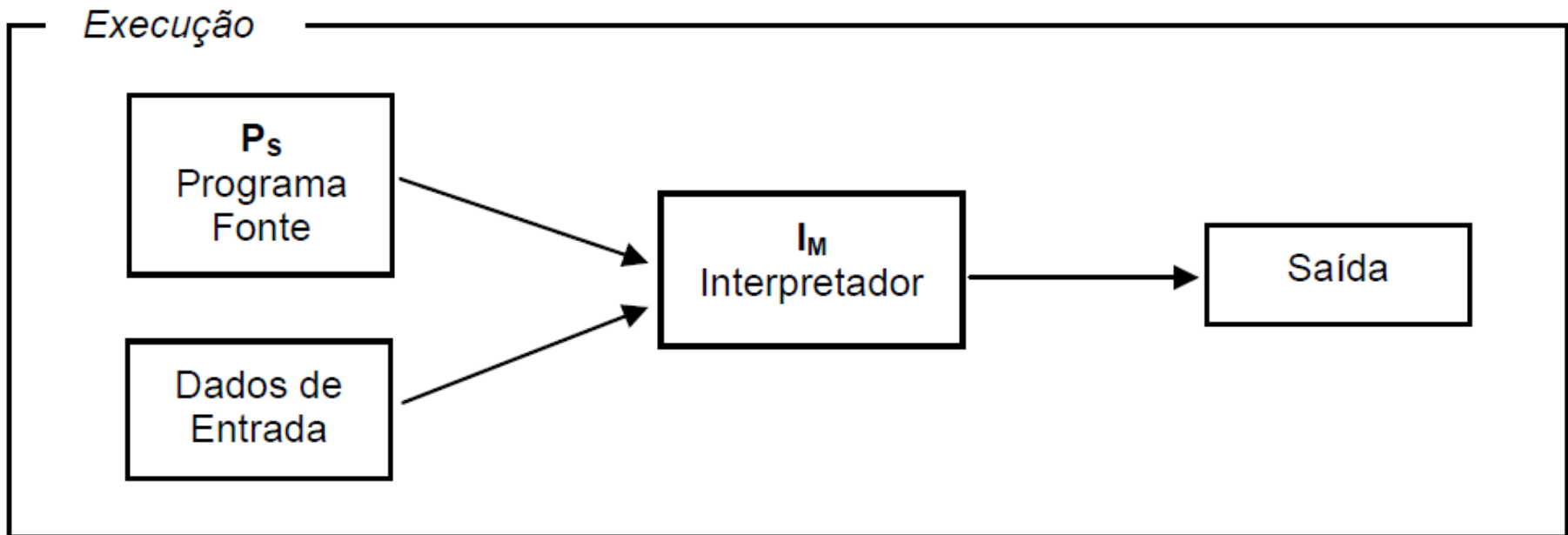
- **INTERPRETAÇÃO VERSUS COMPILAÇÃO**

- Qualquer computador não pode executar programas em sua forma original, apenas programas na linguagem de máquina
- No caso da interpretação, um programa interpretador (I_M), escrito em M , lê o programa P_S escrito em S e simula cada uma de suas instruções, modificando os dados do programa da forma apropriada
- No caso da compilação, um programa compilador (C_M), escrito em M , lê o programa P_C , escrito em C , e traduz cada uma de suas instruções para M , escrevendo um programa P_M cujo efeito é o desejado
- Como consequência deste processo, P_M , por ser um programa escrito em M , pode ser executado em qualquer máquina com a mesma linguagem de máquina M , mesmo que esta máquina não possua um compilador.

CONCEITOS



- Interpretação

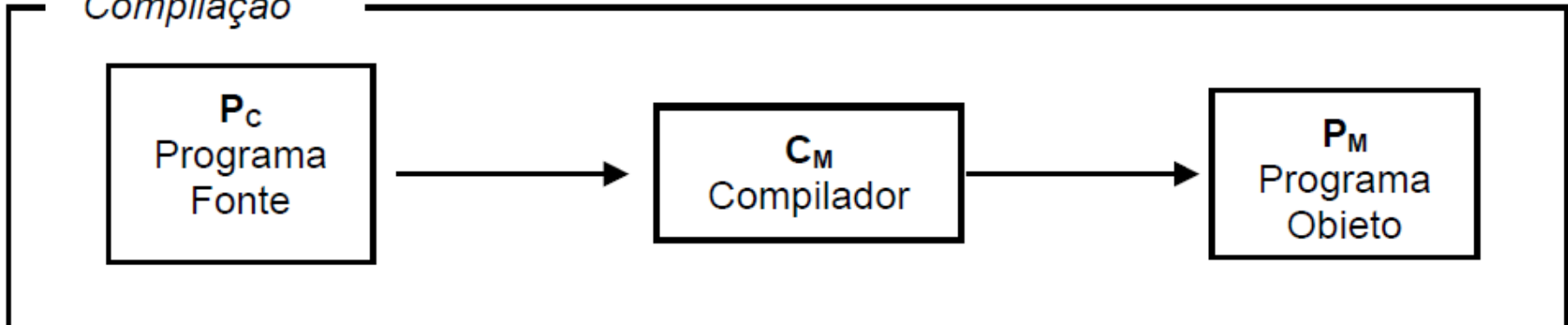




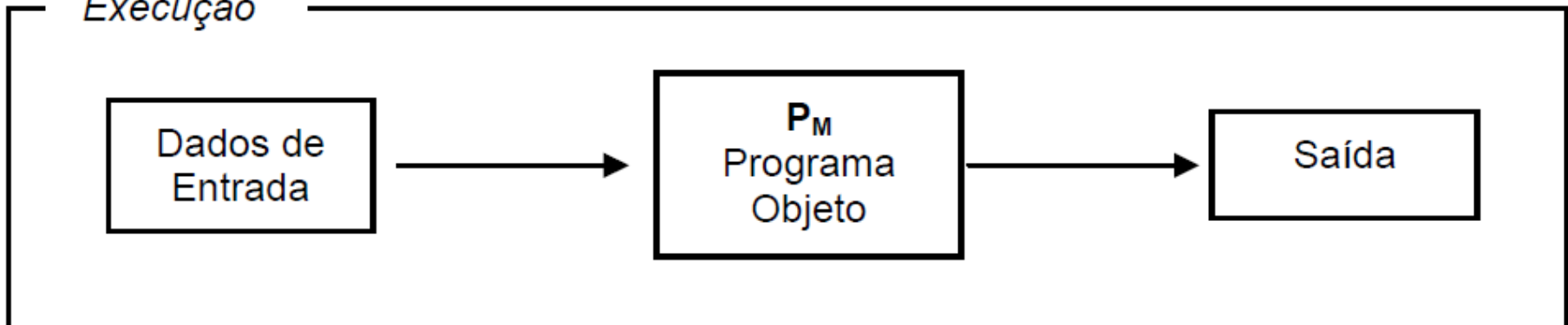
CONCEITOS

- **Compilação**

Compilação



Execução





CONCEITOS

- **EXEMPLO DE CÓDIGO EM C**

- Consideremos um programa que tem a finalidade de converter valores de temperatura dados em Celsius para Fahrenheit

```
/* Programa para conversão de temperatura */  
  
#include <stdio.h>  
  
float converte (float c)  
{  
    float f;  
    f = 1.8*c + 32;  
    return f;  
}
```



CONCEITOS

- **EXEMPLO DE CÓDIGO EM C**

```
int main (void)
{
    float t1;
    float t2;

    /* mostra mensagem para usuario */
    printf("Digite a temperatura em Celsius: ");

    /* captura valor entrado via teclado */
    scanf("%f",&t1);

    /* faz a conversao */
    t2 = converte(t1);

    /* exhibe resultado */
    printf("A temperatura em Fahrenheit é: %f\n", t2);

    return 0;
}
```



CONCEITOS

- **COMPILAÇÃO DE PROGRAMAS EM C**

- Desenvolvimento programas em uma linguagem como C, precisamos de, no mínimo, um editor e um compilador
- Com o editor de textos, escrevemos os programas fontes, que são salvos em arquivos
- Com o compilador, transformamos os programas fontes em programas objetos, em linguagem de máquina, para poderem ser executados
- Os programas fontes são, em geral, armazenados em arquivos cujo nome tem a extensão “.c”
- Os programas executáveis possuem extensões que variam com o sistema operacional: no Windows, têm extensão “.exe”; no Unix (Linux), em geral, não têm extensão
- Com o compilador **gcc**, na linha de comando do sistema operacional, digitar:

```
> gcc -o prog prog.c
```



CONCEITOS

- **COMPILAÇÃO DE PROGRAMAS EM C**
 - Desenvolvimento programas em uma linguagem como C, precisamos de, no mínimo, um editor e um compilador
 - Com o editor de textos, escrevemos os programas fontes, que são salvos em arquivos
 - Com o compilador, transformamos os programas fontes em programas objetos, em linguagem de máquina, para poderem ser executados
 - Os programas fontes são, em geral, armazenados em arquivos cujo nome tem a extensão “.c”
 - Os programas executáveis possuem extensões que variam com o sistema operacional: no Windows, têm extensão “.exe”; no Unix (Linux), em geral, não têm extensão



CONCEITOS

- **COMPILAÇÃO DE PROGRAMAS EM C**

- Com o compilador **gcc**, na linha de comando do sistema operacional, digitar:

- > gcc -o prog prog.c

- Execução

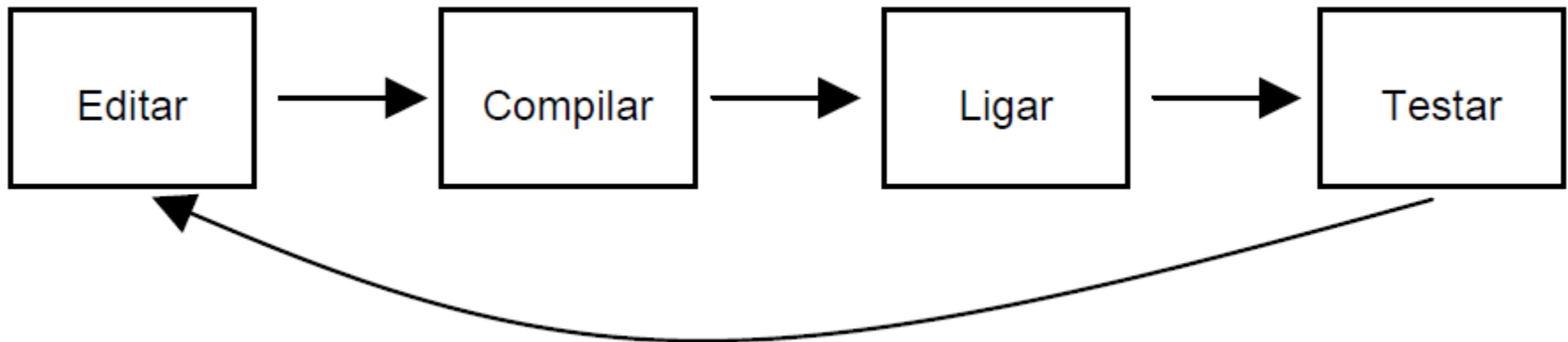
- > ./prog



CONCEITOS

- **VERIFICAÇÃO E VALIDAÇÃO**

- Programas podem conter (e, em geral, contêm) erros, que precisam ser identificados e corrigidos





CONCEITOS

- **VERIFICAÇÃO E VALIDAÇÃO**
- Este ciclo pode ser realizado usando programas (editor, compilador, ligador) separados ou empregando um “ambiente integrado de desenvolvimento” (*Integrated Development Environment*, ou **IDE**)
- **IDE** é um programa que oferece janelas para a edição de programas e facilidades para abrir, fechar e salvar arquivos e para compilar, ligar e executar programas

Estrutura de Dados



- **REFERÊNCIAS**
- Tenenbaum, Aaron M. Langsam, Yedidyah, Augenstein, Moshe J. **Estruturas de dados usando C**. São Paulo : MAKRON *Books*, 1995.
- Veloso, Paulo. et. al. **Estrutura de dados**. Rio de Janeiro: Campus, 1986.
- Moraes, Celso Roberto. **Estrutura de dados e algoritmos**. 2. ed. São Paulo: Futura, 2003.
- Celes, W. Rangel, J. L. **Curso de Estrutura de Dados**. PUC-Rio, 2002.
- W. Celes, R. Cerqueira, J.L. Rangel. **Introdução a Estruturas de Dados - com técnicas de programação em C**. Rio de Janeiro: Campus, 2004.