

Tópicos Especiais à Programação

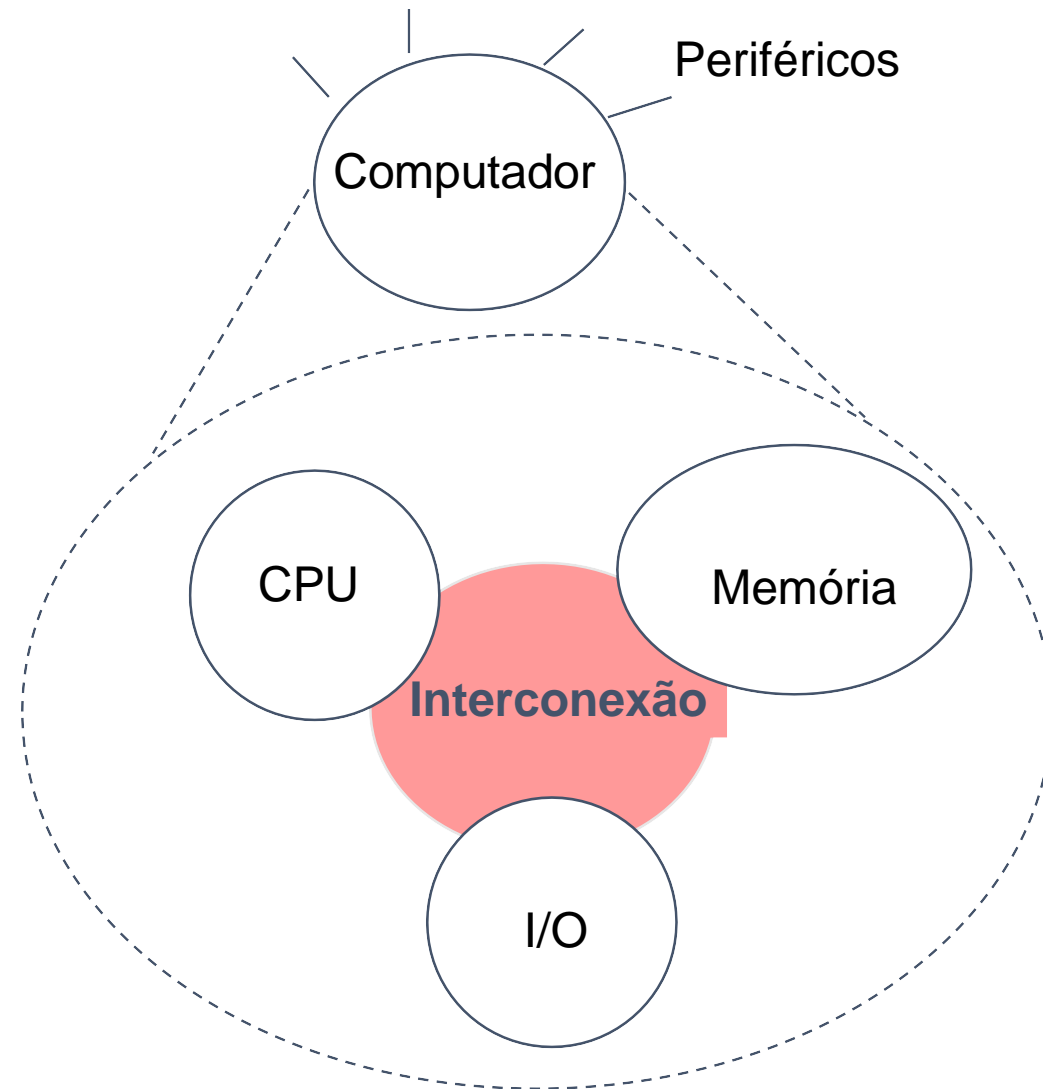
Prof. Me. Nisston Moraes Tavares de Melo

prof2279@iesp.edu.br

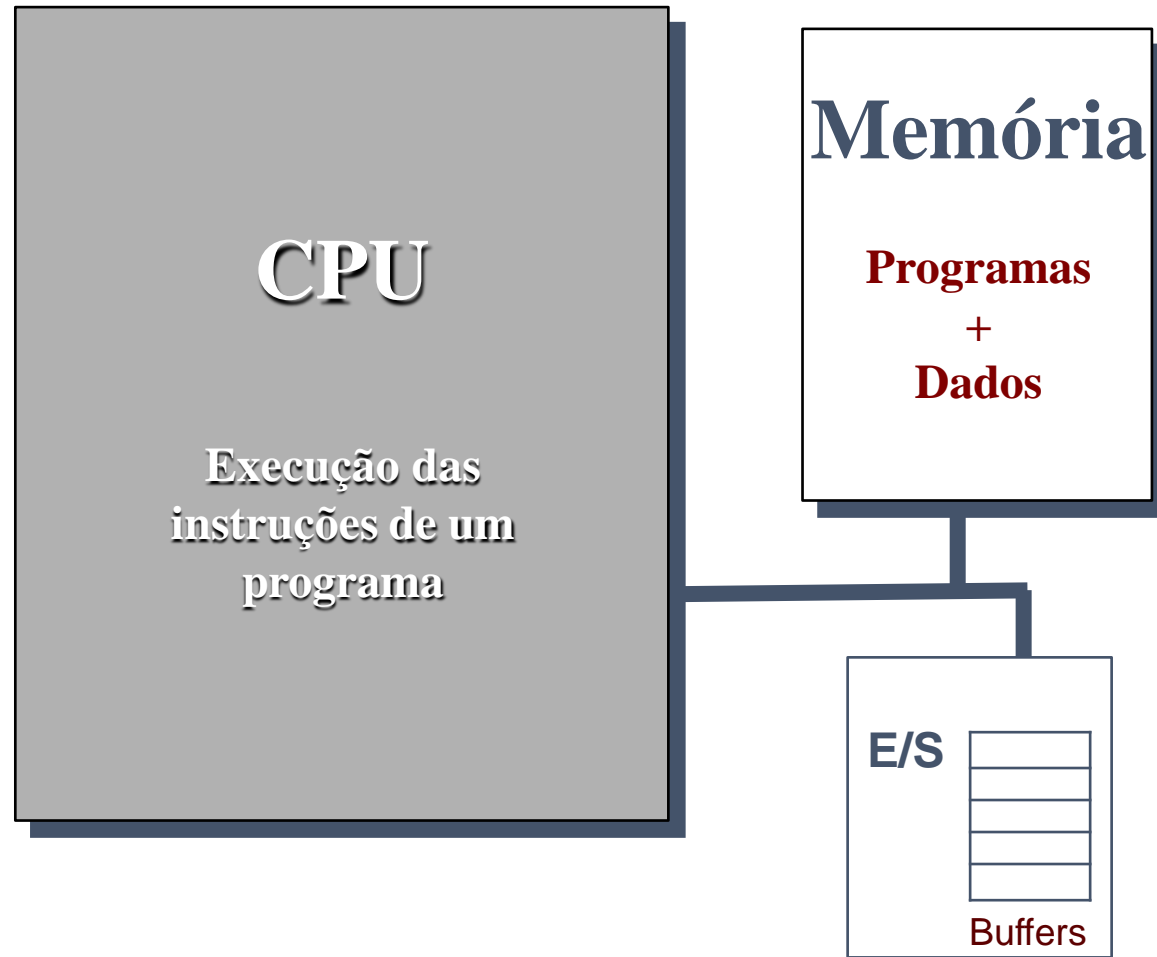
Objetivos da disciplina

- Noções básicas de comandos para os sistemas linux e batch;
- Possua o conhecimento básico para utilizar o github, realizando uma conexão usuário-servidor;
- Utilizar as boas práticas das metodologias de desenvolvimento Devops;
- Conceitos de versionamentos;
- Estruture um fluxo de sprint, planning, tendo por acompanhamento o uso do kanban.
- Conhecimento das diversas áreas básicas da tecnologia.

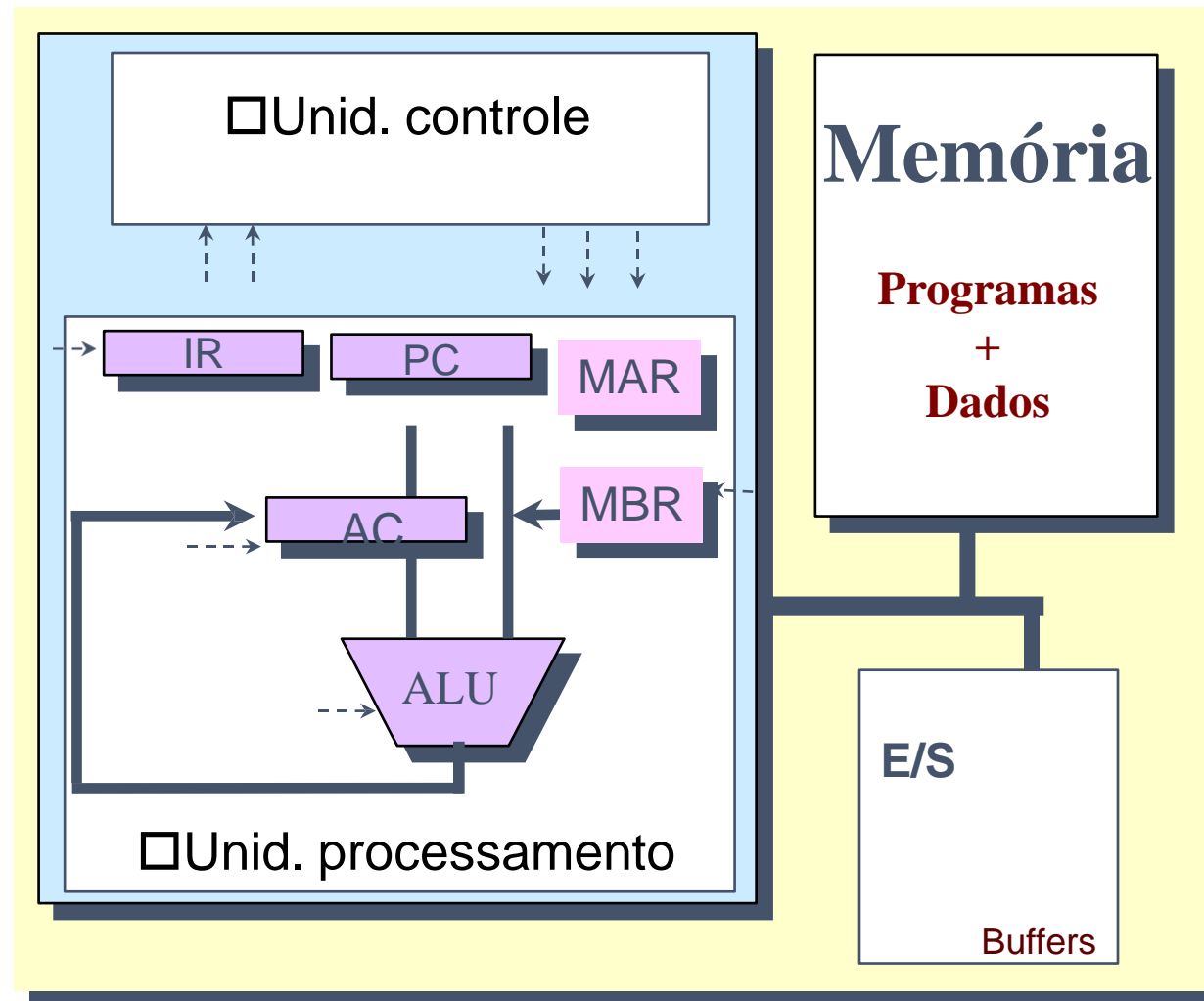
Estrutura de um computador



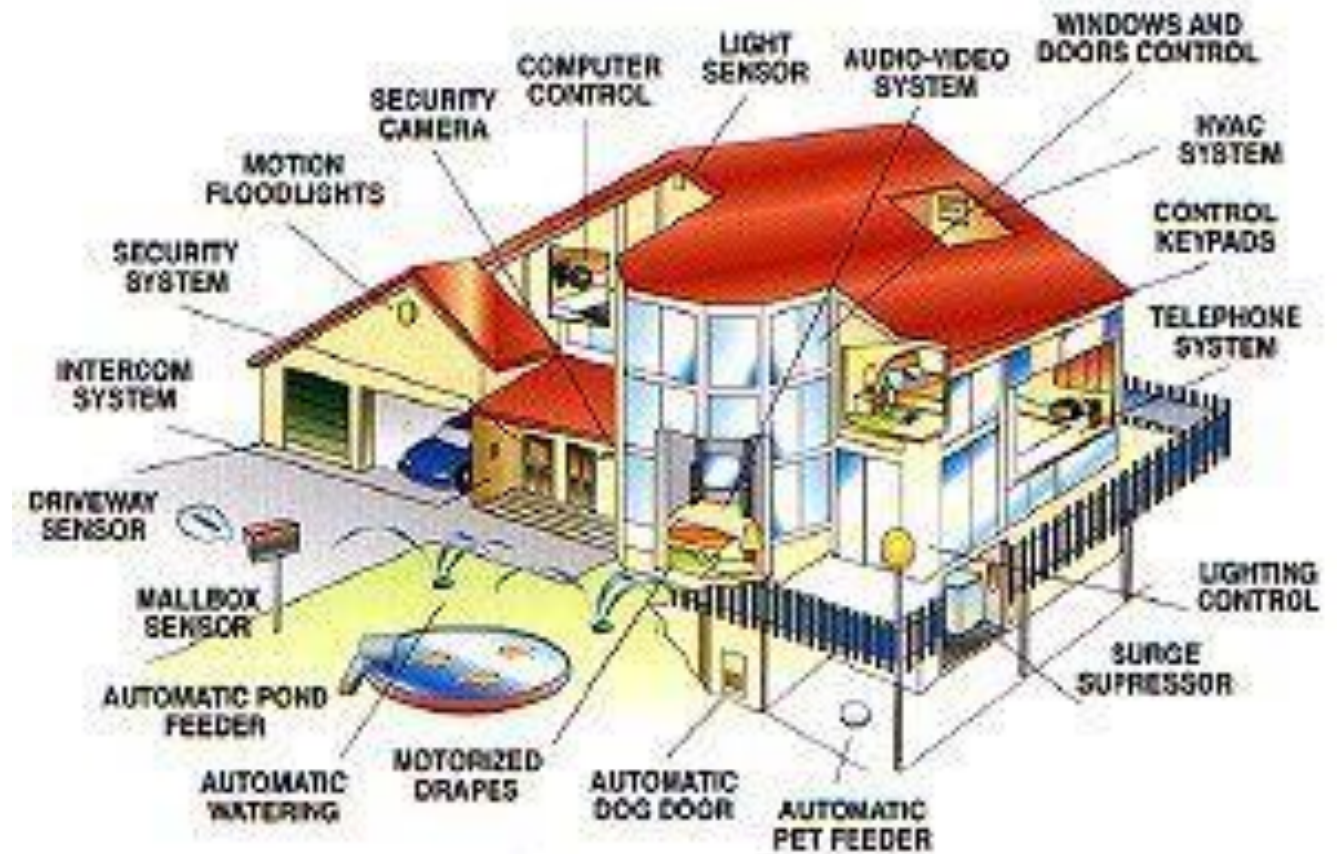
Componentes de um computador



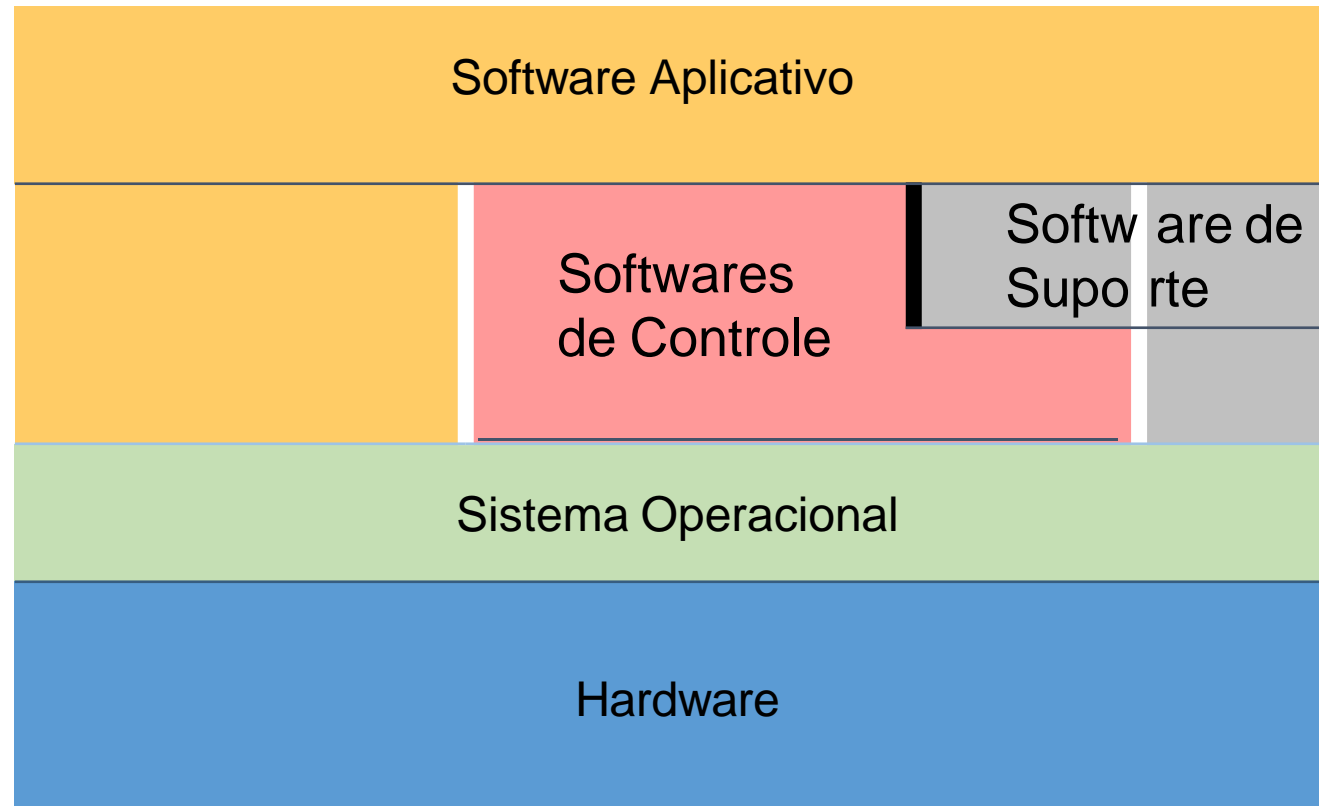
Componentes de um computador



**Computadores
estão
presentes nos mais
diversos
equipamentos**



Hardware & Software



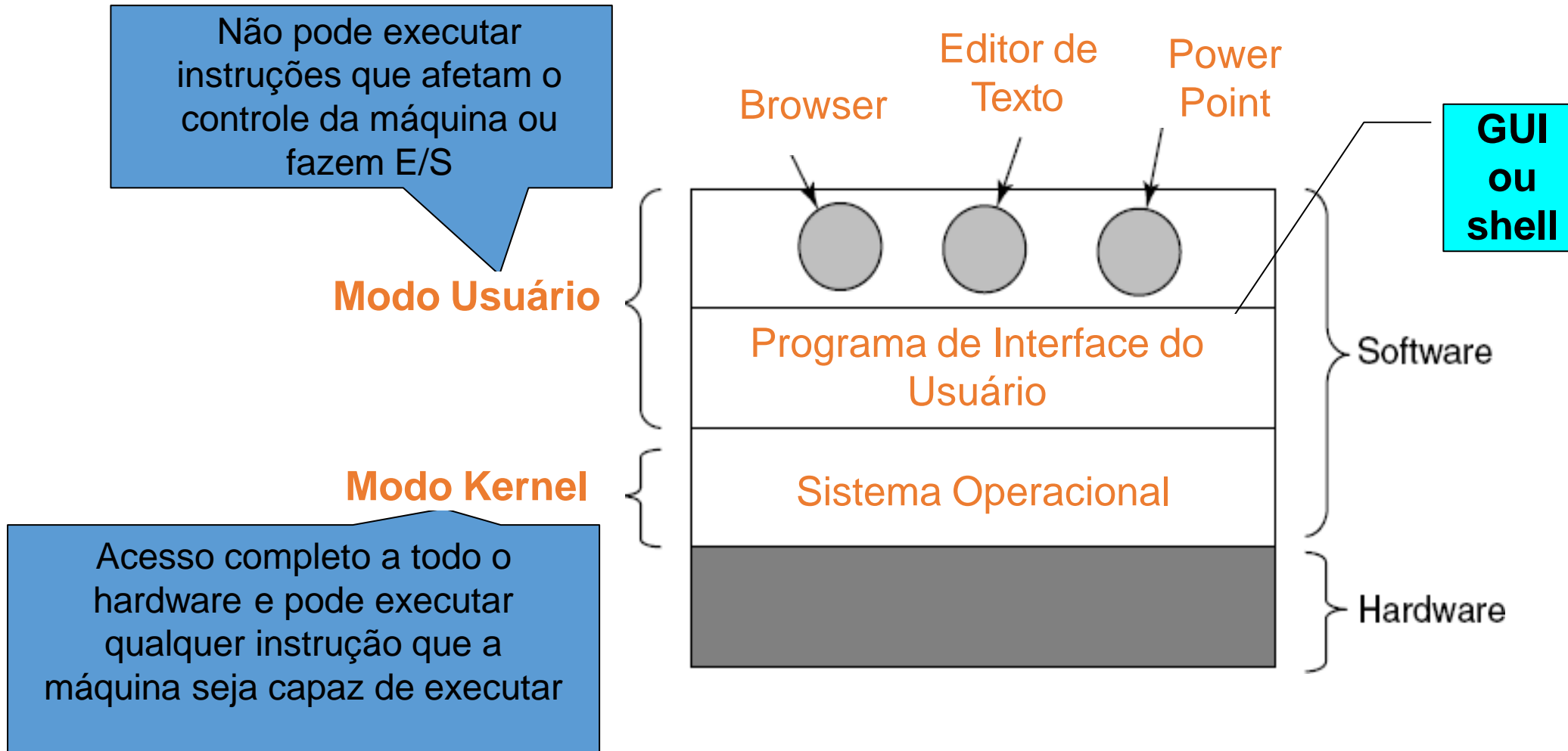
Hardware

São todos os componentes palpáveis de um dispositivo eletrônico, como placas, memória, processador, teclado, monitor, etc. O hardware não se limita apenas ao PC, se referindo também aos itens físicos que compõem celulares, tablets, smart TVs, entre outros aparelhos.

Software

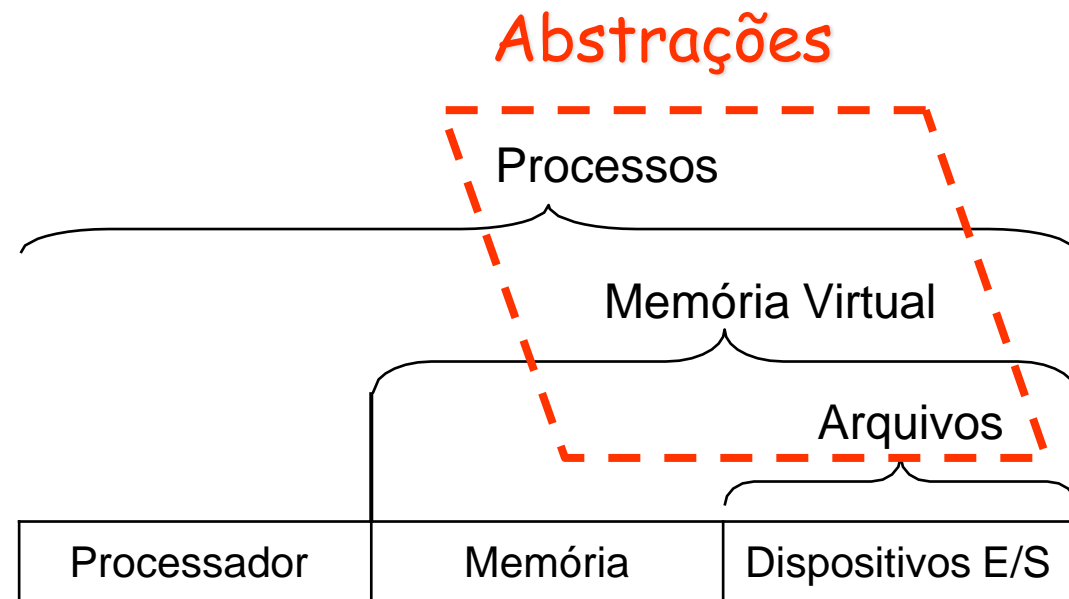
Software é uma sequência de instruções escritas para serem interpretadas por um computador para executar tarefas específicas. Também pode ser definido como os programas, dados e instruções que comandam o funcionamento de um computador, smartphone, tablet e outros dispositivos eletrônicos.

Sistema Computacional em Camadas

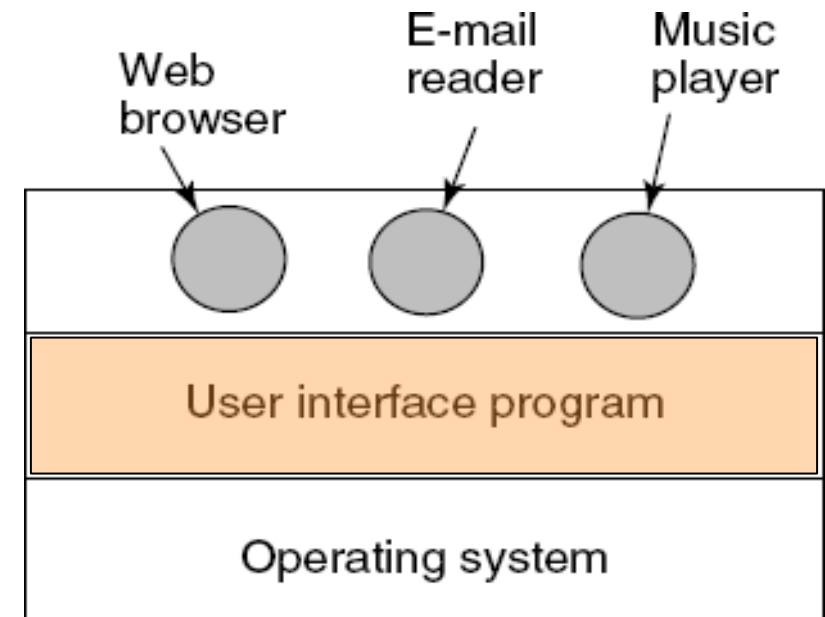
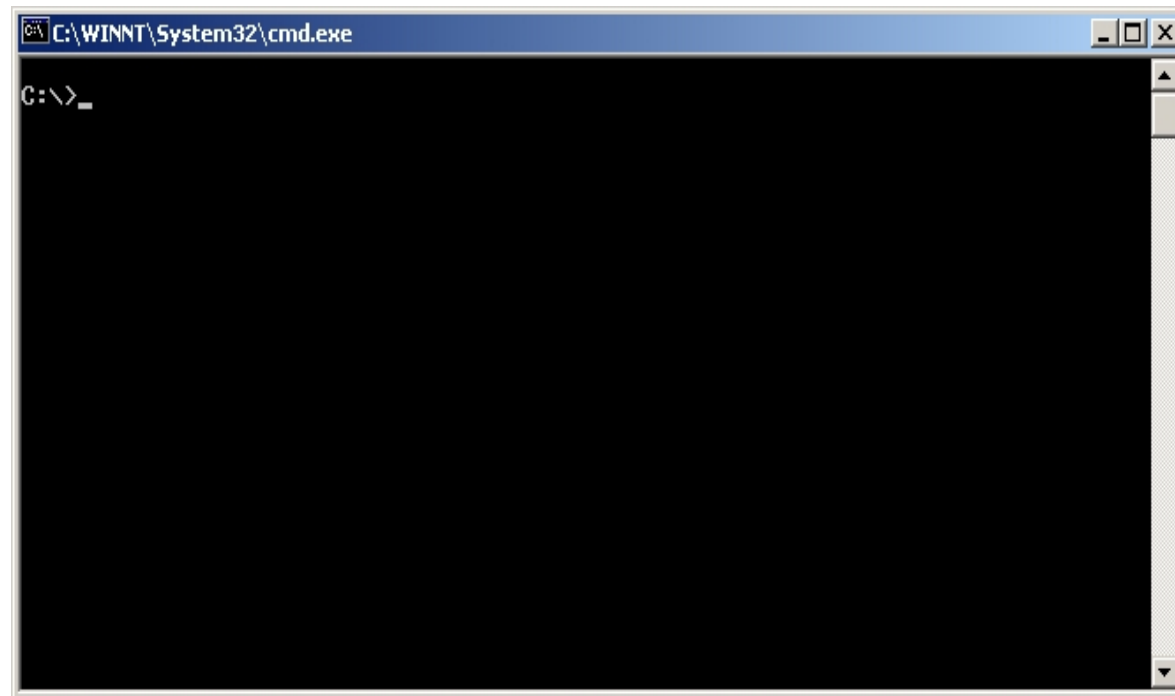


Sistema Computacional em Camadas

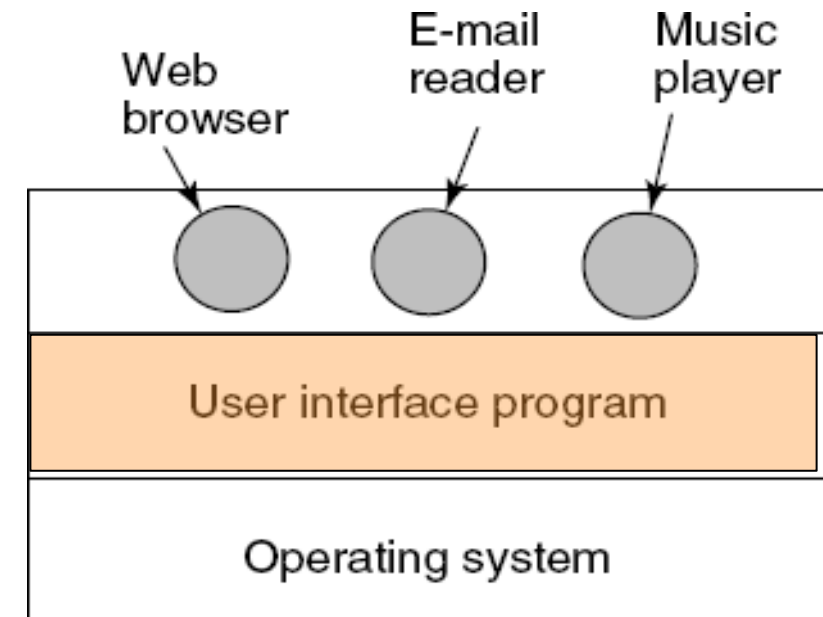
Sistemas operacionais fazem com que o hardware, que tem interfaces difíceis, se torne mais acessível por meio de abstrações mais fáceis e simples.



SO: Interface de Usuário - Shell



SO: Interface de Usuário - GUI – *Graphical User Interface*



O S.O. como Gerenciador de Recursos

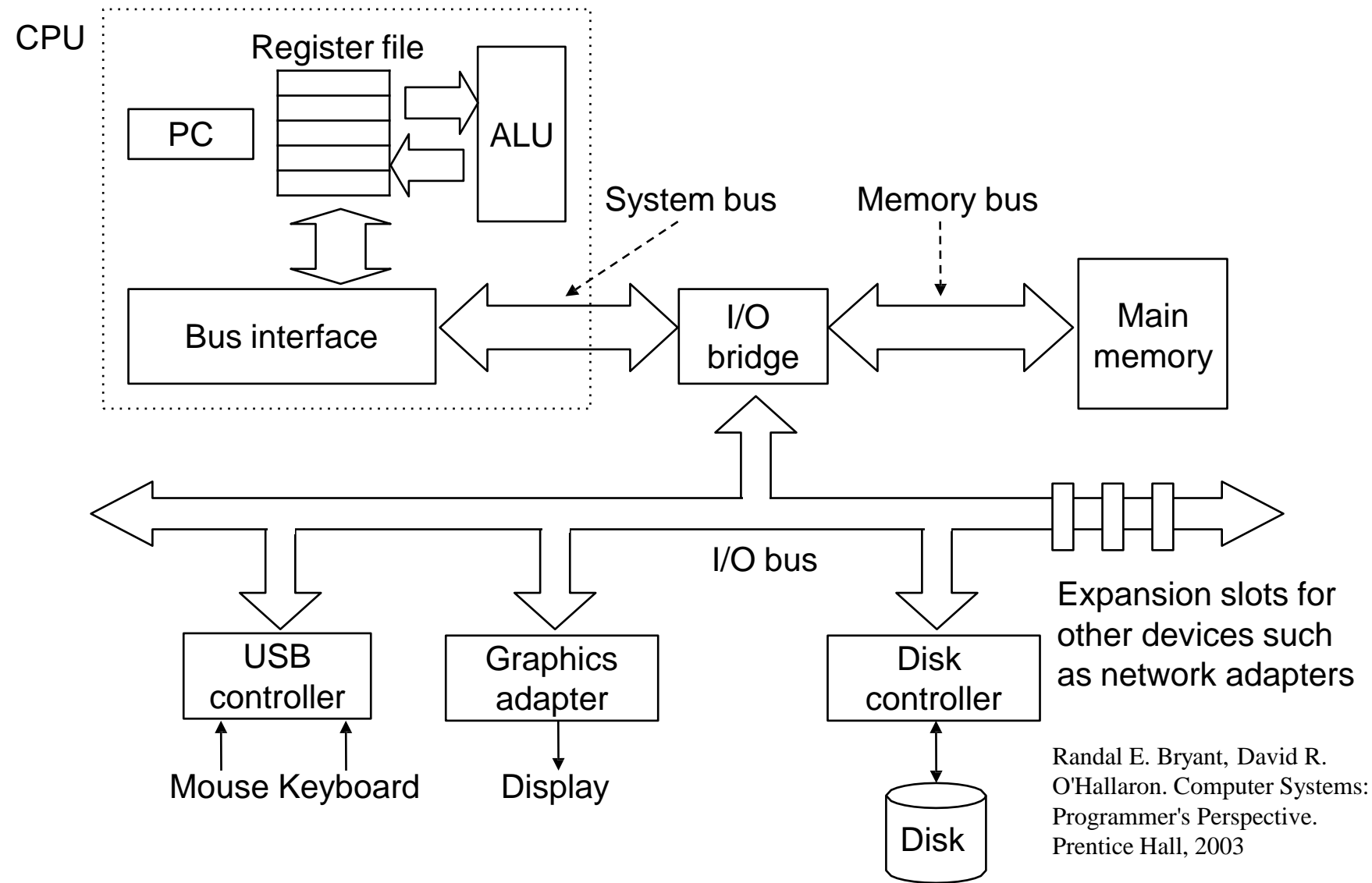
- **Gerencia** e **protege** memória, dispositivos de E/S e outros recursos (hardware)
- Permite o compartilhamento de recursos
 - no tempo (time-sharing)
 - Ex.: múltiplos programas compartilham o processador (executam) ao “mesmo tempo”
 - no espaço
 - Ex.: dados de diferentes usuários/arquivos compartilhem o espaço em disco

Tipos de Sistemas Operacionais

- Sistemas operacionais de computadores de grande porte (*mainframe*)
- Sistemas operacionais de servidores / redes
- Sistemas operacionais de multiprocessadores (paralelismo)
- Sistemas operacionais de computadores pessoais
- Sistemas operacionais de dispositivos portáteis/ móveis (ex. celulares)
- Sistemas operacionais de tempo-real
- Sistemas operacionais embarcados
- Sistemas operacionais de cartões inteligentes
- Sistemas operacionais de sensores

Hardware

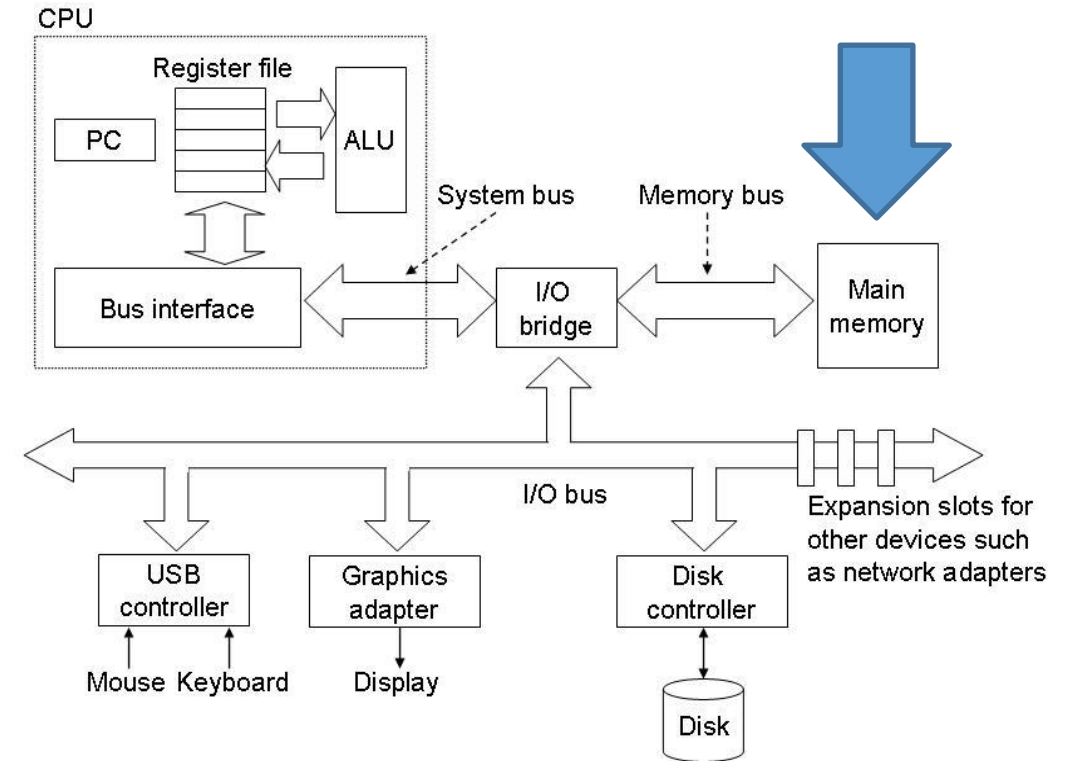
Hardware



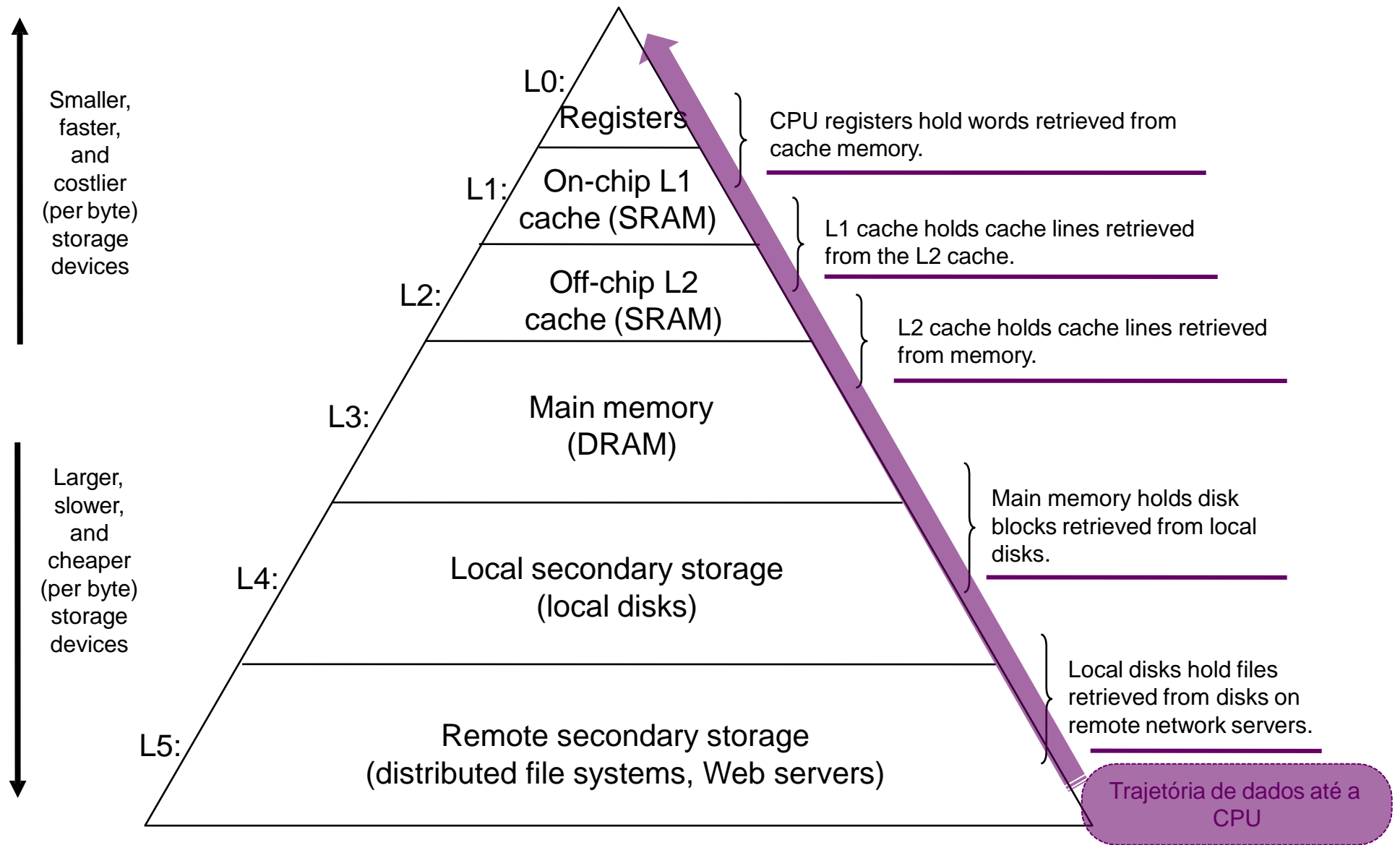
Randal E. Bryant, David R. O'Hallaron. Computer Systems: A Programmer's Perspective. Prentice Hall, 2003

Memória

- Logicamente, a memória principal corresponde a um enorme vetor (array) de bytes
 - cada posição tem um endereço único (índice do vetor)
- Os registradores da CPU muitas vezes são usados para armazenar endereços de memória
 - Assim, o número de bits em cada registrador **limita** o número de **posições de memória endereçáveis**
 - Ex.: 8 bits \Rightarrow 256 posições...



Hierarquia de Memória

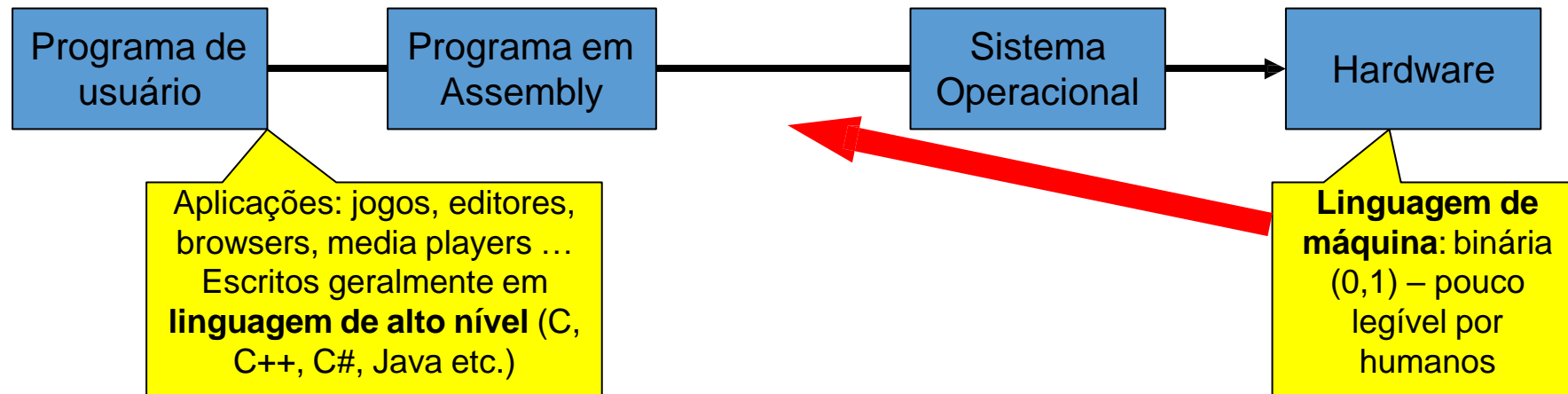


Software

Executando um programa

Programa Executável

- “Conhecendo mais sobre o que está ‘por baixo’ do programa, você pode escrever programas mais eficientes e confiáveis”
- Abstrações em um sistema de computação:

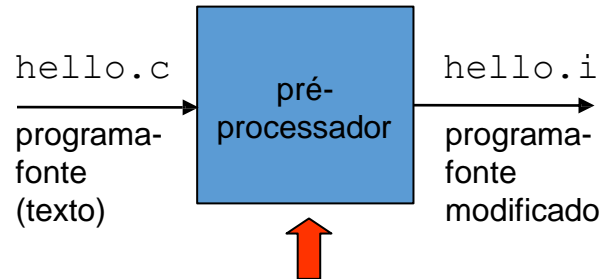


- A linguagem de montagem (Assembly) é um **mapeamento direto da linguagem de máquina**, mas que introduz várias “facilidades” (ou “menos dificuldades”) para o programador

Gerando um executável

```
unix> gcc -o hello hello.c
```

```
1. #include <stdio.h>
2. int main()
3. {
4.     printf("hello, world\n");
5. }
```

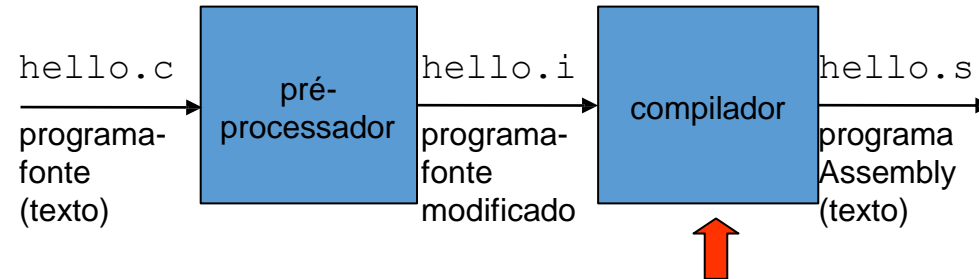


- Modifica o programa em C de acordo com diretivas começadas com #
 - Ex.: `#include <stdio.h>` diz ao pré-processador para ler o arquivo `stdio.h` e inseri-lo no programa fonte
- O resultado é um programa expandido em C, normalmente com extensão `.i`, em Unix

Gerando um executável

```
unix> gcc -o hello hello.c
```

```
1. #include <stdio.h>
2. int main()
3. {
4.     printf("hello, world\n");
5. }
```

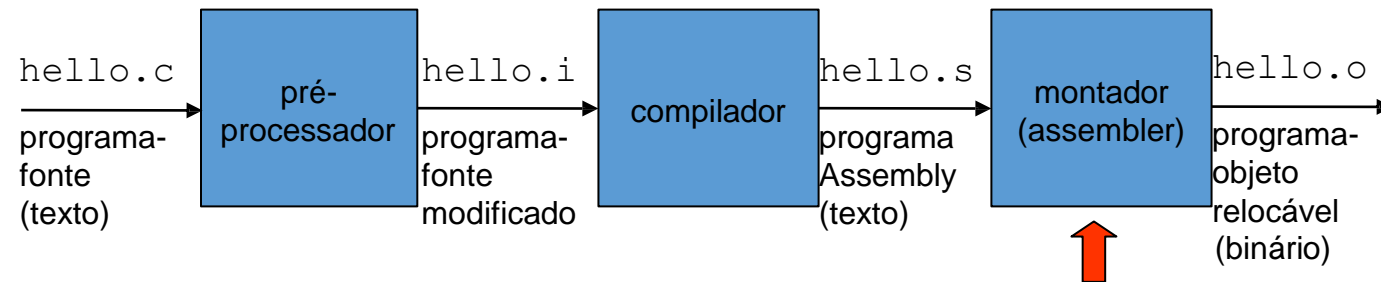


- Compilador traduz o programa .i em um programa em Assembly
 - É o formato de saída comum para os compiladores nas várias linguagens de programação de alto nível
 - i.e., programas em C, Java, Fortran, etc vão ser traduzidos para a mesma linguagem Assembly

Gerando um executável

```
unix> gcc -o hello hello.c
```

```
1. #include <stdio.h>
2. int main()
3. {
4.     printf("hello, world\n");
5. }
```

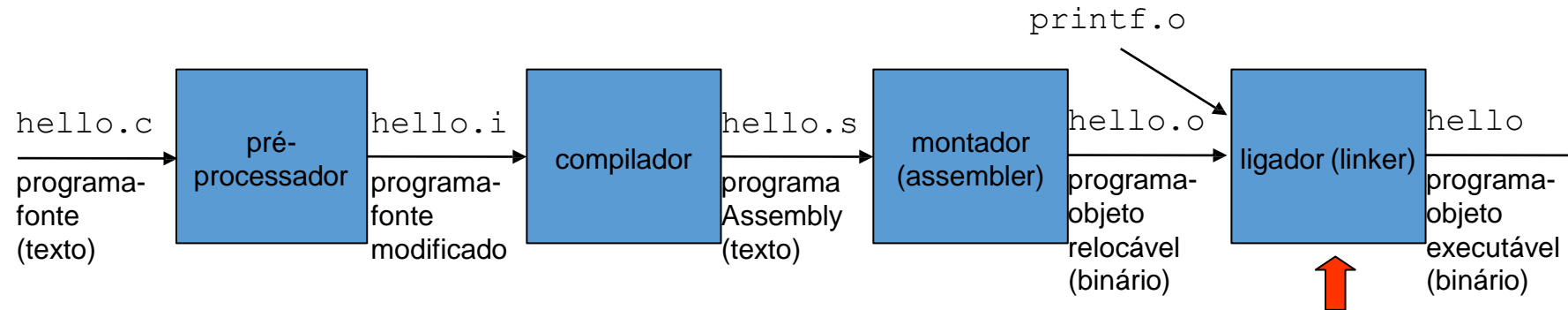


- Montador (Assembler) transforma o programa em Assembly em um programa binário em linguagem de máquina (chamado programa-objeto)
 - Os módulos de programas, compilados ou montados, são armazenados em um formato intermediário (“Programa-Objeto Relocável” – extensão .o)
- Endereços de acesso e a posição do programa na memória ficam **indefinidos**

Gerando um executável

```
unix> gcc -o hello hello.c
```

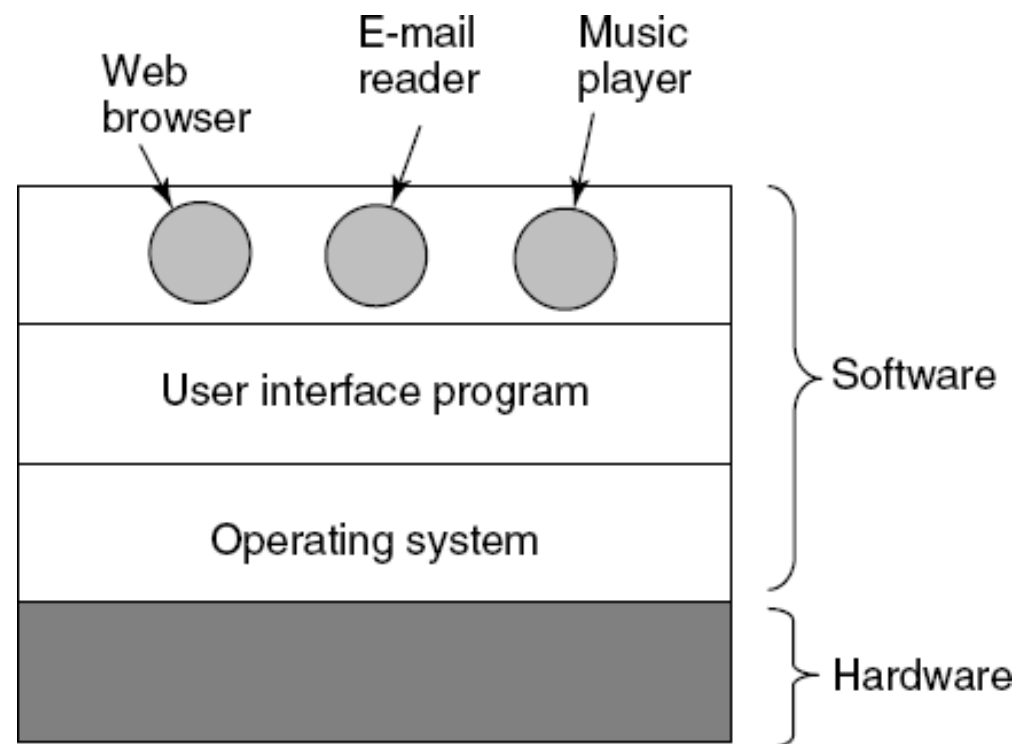
```
1. #include <stdio.h>
2. int main()
3. {
4.     printf("hello, world\n");
5. }
```



- O ligador (linker) gera o programa executável a partir do `.o` gerado pelo assembler
 - No entanto, pode haver funções-padrão da linguagem (ex., `printf`) que não estão definidas no programa, mas em outro arquivo `.o` pré-compilado (`printf.o`).
 - O ligador faz a junção dos programas-objeto necessários para gerar o executável

















Mais de um programa em execução

- Múltiplos processos vs. um (ou [poucos] mais) processador(es) \Rightarrow **como pode???**



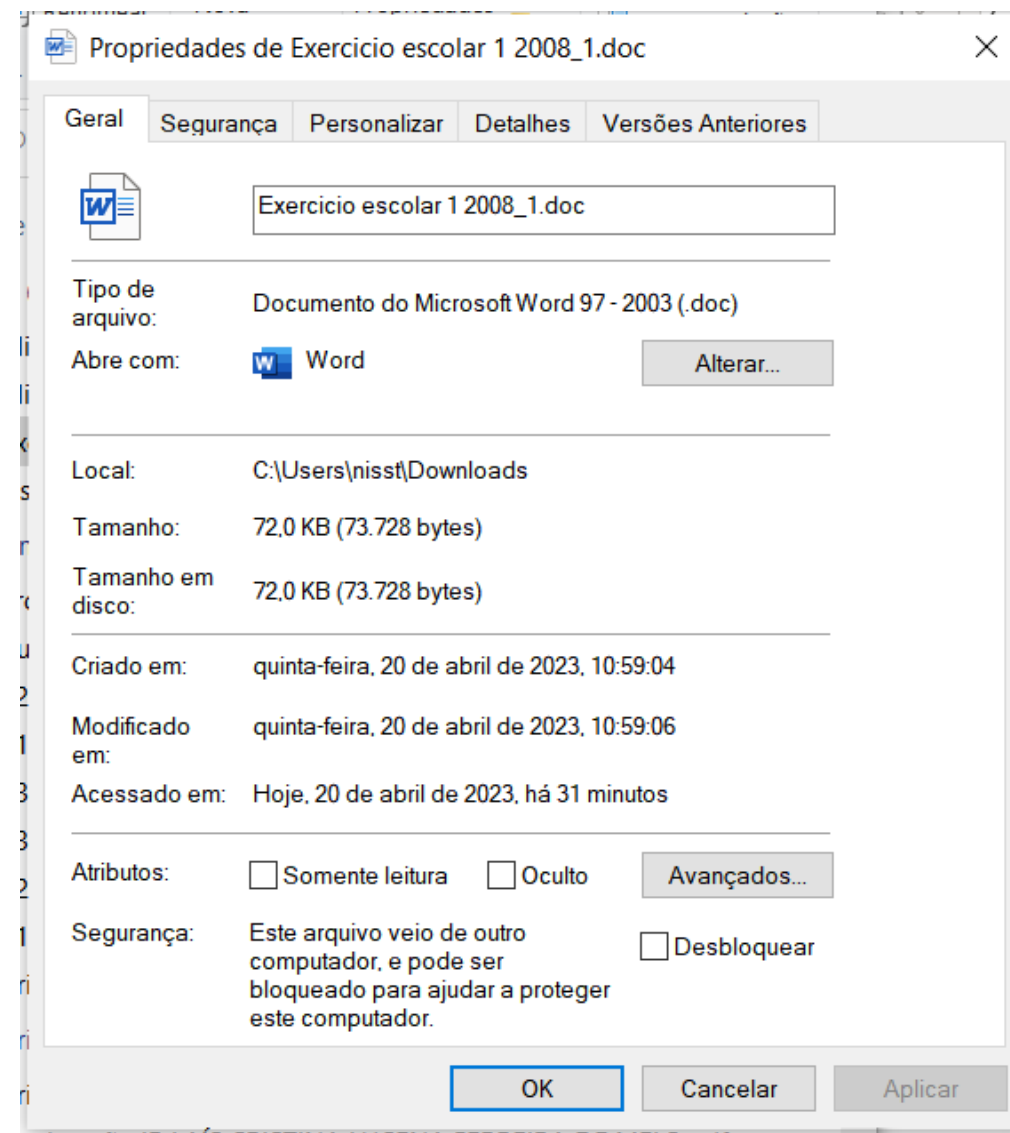
Arquivo

- Nome_do_Arquivo.Extensão
 - Tamanho do nome
 - Tamanho da extensão
 - Tipos de extensão

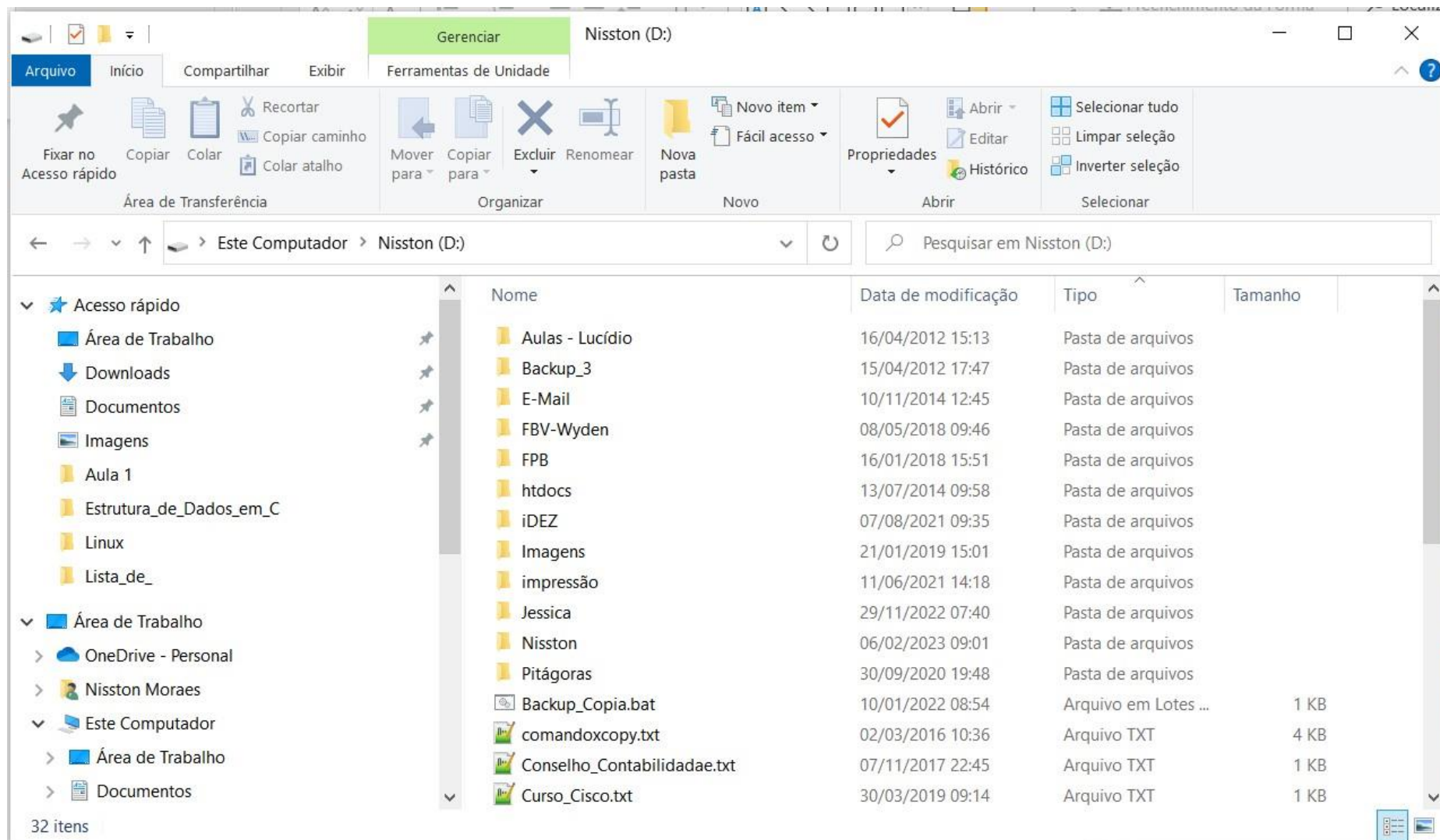
Nome	Data de modificação	Tipo	Tamanho
 Backup_Copia.bat	10/01/2022 08:54	Arquivo em Lotes ...	1 KB
 comandocopy.txt	02/03/2016 10:36	Arquivo TXT	4 KB
 Conselho_Contabilidadae.txt	07/11/2017 22:45	Arquivo TXT	1 KB
 Curso_Cisco.txt	30/03/2019 09:14	Arquivo TXT	1 KB
 Informações e Configurações.txt	14/04/2023 16:08	Arquivo TXT	44 KB
 registroback.txt	10/01/2022 09:08	Arquivo TXT	8 KB
 Evidências_Ações_2018_1.docx	02/04/2018 09:54	Documento do Mi...	12 KB
 Formulario.docx	28/10/2016 16:05	Documento do Mi...	12 KB
 Formulario_Carlos.docx	28/10/2016 18:50	Documento do Mi...	12 KB
 Formulario_Maria.docx	28/10/2016 18:50	Documento do Mi...	12 KB
 msdia80.dll	02/12/2006 00:37	Extensão de aplica...	884 KB
 Consolidada_2013.pdf	31/10/2013 23:06	Microsoft Edge PD...	116 KB
 Curso de Fotografia.pdf	15/10/2016 07:25	Microsoft Edge PD...	9.009 KB
 DARF1.pdf	12/04/2017 17:19	Microsoft Edge PD...	76 KB
 DARF2.pdf	12/04/2017 17:19	Microsoft Edge PD...	76 KB
 Entrega_Documentação.pdf	05/09/2017 10:04	Microsoft Edge PD...	217 KB

Estrutura dos arquivos

- Arquivo
 - Criando
 - Renomeando
 - Apagando
 - Propriedades
 - Geral
 - Segurança
 - Acesso
- Pasta
 - Criando
 - Renomeando
 - Apagamento
 - Propriedades
 - Geral
 - Segurança
 - Acesso

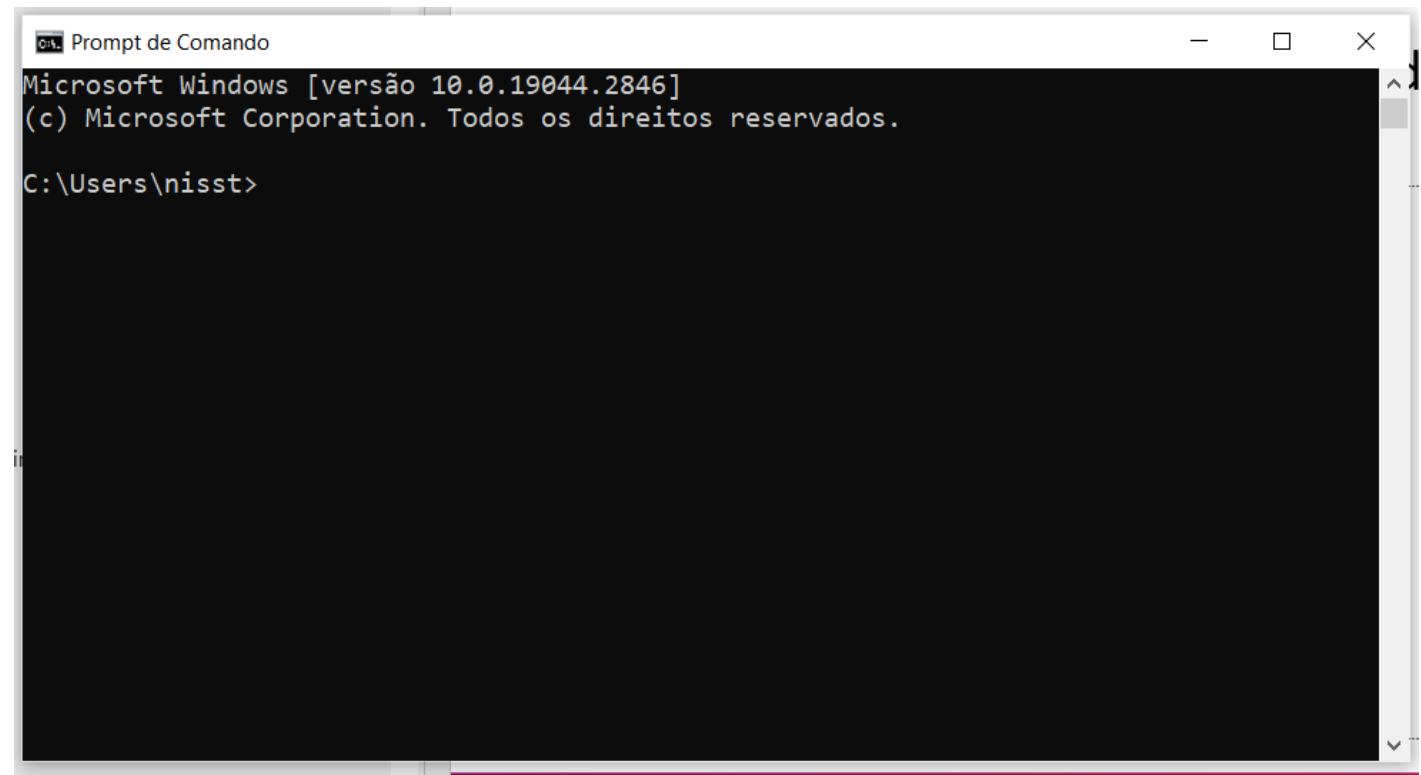


Windows



Abrindo um terminal Shell no Windows

- Digite o comando CMD na pesquisa do Windows

A screenshot of a Windows Command Prompt window. The title bar reads "Prompt de Comando". The window content shows the following text: "Microsoft Windows [versão 10.0.19044.2846]", "(c) Microsoft Corporation. Todos os direitos reservados.", and "C:\Users\nisst>". The window has a standard Windows interface with minimize, maximize, and close buttons in the top right corner.

```
Microsoft Windows [versão 10.0.19044.2846]
(c) Microsoft Corporation. Todos os direitos reservados.

C:\Users\nisst>
```

Comandos Windows

dir -- > listando arquivos e pastas
mkdir -- > criar nova pasta
rmdir -- > remover uma pasta
cd -- > entrar em uma pasta
cd .. -- > retornar para pasta anterior
copy -- > copia arquivo
xcopy --> copia arquivo e pasta
ren --> altera o nome de um arquivo
print --> imprimir um arquivo



Exercício

Linux

Comandos Linux

ls -- > listando arquivos e pastas
df -- > espaço usado no disco rígido
top -- > uso da memória
mkdir -- > criar nova pasta
cd -- > entrar em uma pasta
cd .. -- > retornar para pasta anterior
cat -- > abre um arquivo
cp -- > copia arquivo
rm -- > remover arquivo
rmdir -- > remover diretório
mv -- > move um arquivo



Exercício

Links de vídeos

- <https://youtu.be/T7lCM3l7vAQ>
- <https://youtu.be/9JCElq7svL4>
- <https://youtu.be/YB-Fp0oD45E>
- <https://youtu.be/LMoHEbCFtew>
- <https://youtu.be/QZ2nyxzZXPY>
- <https://youtu.be/uoORbQHDtk0>



Dúvidas?

Referência Bibliográfica

- Livro
 - TANENBAUM, Andrew S.; WOODHULL, Albert S. Sistemas operacionais. Grupo A, 2008. E-book. ISBN 9788577802852. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788577802852/>. Acesso em: 20 abr. 2023.
 - JR., Ramiro S C.; LEDUR, Cleverson L.; MORAIS, Izabelly S. Sistemas operacionais. Grupo A, 2019. E-book. ISBN 9788595027336. Disponível em: <https://integrada.minhabiblioteca.com.br/#/books/9788595027336/>. Acesso em: 20 abr. 2023.



uniesp

Centro Universitário