

# Sistemas Operacionais

## Gestão de arquivos - o conceito de arquivo

Prof. Carlos Maziero

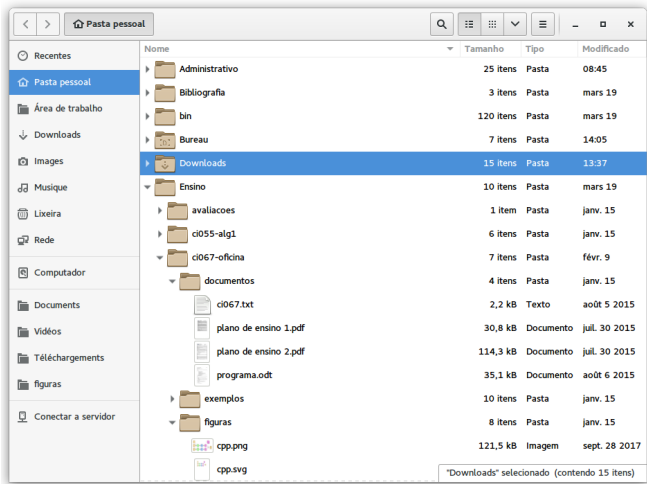
DInf UFPR, Curitiba PR

Agosto de 2020

# Conteúdo

- 1 Arquivos e sistemas de arquivos
- 2 Atributos e operações
- 3 Formatos de arquivos
- 4 Identificação de conteúdo
- 5 Arquivos especiais

# Arquivos e diretórios



The screenshot shows a Windows File Explorer window titled "Pasta pessoal". The left sidebar shows the navigation pane with "Downloads" selected. The main pane displays a list of folders and files in the "Downloads" folder. A status bar at the bottom indicates "Downloads" is selected and contains 15 items.

Nome	Tamanho	Tipo	Modificado
Administrativo	25 itens	Pasta	08:45
Bibliografia	3 itens	Pasta	mars 19
bin	120 itens	Pasta	mars 19
Bureau	7 itens	Pasta	14:05
Downloads	15 itens	Pasta	13:37
Ensino	10 itens	Pasta	mars 19
avaliacoes	1 item	Pasta	janv. 15
ci055-alg1	6 itens	Pasta	janv. 15
ci067-oficina	7 itens	Pasta	févr. 9
documentos	4 itens	Pasta	janv. 15
ci067.txt	2,2 kB	Texto	août 5 2015
plano de ensino 1.pdf	30,8 kB	Documento	juil. 30 2015
plano de ensino 2.pdf	114,3 kB	Documento	juil. 30 2015
programa.odt	35,1 kB	Documento	août 6 2015
exemplos	10 itens	Pasta	janv. 15
figuras	8 itens	Pasta	janv. 15
cpp.png	121,5 kB	Imagem	sept. 28 2017
cpp.svg			

"Downloads" selecionado (contendo 15 itens)

# Arquivos

## Arquivo

Sequência de bytes armazenada em um dispositivo não-volátil, com um nome que permita sua localização.

- Unidade básica de armazenamento de dados de usuário
- Extremamente versáteis em conteúdo e capacidade
  - Texto, código, planilha, imagens, música, vídeo, ...
  - De alguns bytes a Gigabytes
- Organizados em estruturas hierárquicas (*diretórios*)

# Sistemas de arquivos

## Sistema de arquivos

Organização física e lógica dos arquivos e diretórios dentro de um dispositivo físico.

- **Imensa** estrutura de dados armazenada no dispositivo
- Geralmente uma árvore ou grafo
- Milhões de nós variados (arquivos, diretórios, links, ...)
- Várias implementações: NTFS, FAT, Ext4, HPFS, ...

# Conteúdo, atributos e operações



Conteúdo:

- Pepinos, água, temperos

Atributos:

- Informações da etiqueta

Operações:

- Abrir, fechar
- Tirar, por, ...

# Atributos de arquivos

Cada arquivo possui um conjunto de **atributos**:

- Nome
- Tipo
- Tamanho
- Datas
- Proprietário
- Permissões
- Localização
- etc.

```

1 $ stat slides.pdf
2
3   File: slides.pdf
4   Size: 527546      Blocks: 1032      IO Block: 4096  regular file
5 Device: 812h/2066d Inode: 1195571   Links: 1
6 Access: (0644/-rw-r--r--)  Uid: (1000/ maziero)  Gid: (1000/ maziero)
7 Access: 2020-08-31 10:14:43.760611368 -0300
8 Modify: 2020-08-31 10:14:43.664610398 -0300
9 Change: 2020-08-31 10:14:43.664610398 -0300
10 Birth: -
  
```

# Operações sobre arquivos

- **Criar:** alocar entrada para ele no sistema de arquivos
- **Abrir:** preparar o SO para a usar o arquivo:
  - Verificar se o arquivo existe
  - Verificar as permissões de acesso
  - Localizar seu conteúdo no dispositivo
  - Criar descritores no núcleo e na aplicação
- **Ler:** transferir dados do arquivo para a memória
- **Escrever:** transferir dados da memória para o arquivo
- **Fechar:** liberar as estruturas criadas ao abri-lo
- **Remove:** eliminar o arquivo do sistema de arquivos
- **Mudar atributos:** mudar nome, proprietário, etc.



# Formatos de arquivos

Todo arquivo contém uma **sequência de bytes**.

O conteúdo de um arquivo é interpretado pela aplicação.

Algumas estruturas internas de arquivos são frequentes:

- Sequências de bytes
- Sequências de registros
- Arquivos de texto
- Arquivos executáveis

# Sequências de bytes

Estrutura básica usada em **todos** os arquivos.

Cabe às aplicações definir uma semântica aos bytes.

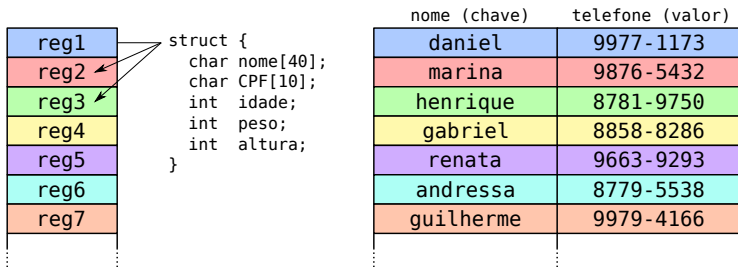
Padrões são definidos para informações usuais:

- Documentos: PDF, RTF, ODT, DOCX, ...
- Imagens: JPG, PNG, GIF, ...
- Áudio: MP3, WAV, OGG, FLAC, ...
- Desenhos: DXF, SVG, ...

Quem interpreta o formato do conteúdo é **a aplicação**.

# Arquivos de registros

- Sequências de registros de tamanho fixo ou variável.
- Podem ser indexados (chave/valor).
- Suportados pelo núcleo (OpenVMS).
- Suportados por bibliotecas (Berkeley DB, SQLite).



# Arquivos de texto

Formato de arquivo mais popular:

- Código-fonte, scripts, páginas HTML, configurações, etc

Estrutura geral:

- Arquivo: sequência de linhas de caracteres.
- Os caracteres são codificados (ASCII, UTF-8, ...).
- As linhas têm tamanho variável (geralmente pequeno).
- Linhas separadas por caracteres de controle.
- Estrutura depende do sistema operacional.

Este formato é reconhecido por muitas aplicações.

# Arquivos de texto em UNIX

Linhas separadas por `0x0a (\n)`:

```

1  int_main()↓
2  {↓
3  printf("Hello, world\n");↓
4  exit(0);↓
5  }↓

```

```

1  0000  69 6e 74 20 6d 61 69 6e 28 29 0a 7b 0a 20 20 70
2      i n t _ m a i n ( ) \n { \n _ _ p
3  0010  72 69 6e 74 66 28 22 48 65 6c 6c 6f 2c 20 77 6f
4      r i n t f ( " H e l l o , _ w o
5  0020  72 6c 64 5c 6e 22 29 3b 0a 20 20 65 78 69 74 28
6      r l d \ n " ) ; \n _ _ e x i t (
7  0030  30 29 3b 0a 7d 0a
8      0 ) ; \n } \n

```

# Arquivos de texto em Windows

Linhas separadas por `0x0d0a (\r\n)`:

```

1  int_main() ←↓
2  { ←↓
3  printf("Hello, world\n"); ←↓
4  exit(0); ←↓
5  } ←↓

```

```

1  0000  69 6e 74 20 6d 61 69 6e 28 29 0d 0a 7b 0d 0a 20
2      i n t _ m a i n ( ) \r \n { \r \n \
3  0010  20 70 72 69 6e 74 66 28 22 48 65 6c 6c 6f 2c 20
4      \ p r i n t f ( " H e l l o , \
5  0020  77 6f 72 6c 64 5c 6e 22 29 3b 0d 0a 20 20 65 78
6      w o r l d \ n " ) ; \r \n \ \ e x
7  0030  69 74 28 30 29 3b 0d 0a 7d 0d 0a
8      i t ( 0 ) ; \r \n } \r \n

```

# Arquivo de código

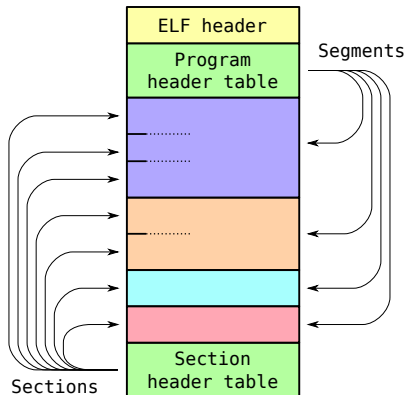
Usado em executáveis e bibliotecas compiladas

- Estruturado em seções a carregar em RAM
- Código, tabelas de símbolos, dependências e configurações
- Estrutura depende do sistema operacional

Formatos mais comuns:

- **ELF** (*Executable and Linking Format*): sistemas UNIX
- **PE** (*Portable Executable*): sistemas Windows
- **COFF** (*Common Object File Format*): UNIX antigos

# O formato ELF



- *Header*: descreve o conteúdo
- *Section header table*: descreve as seções do conteúdo
- *Sections*: código binário, constantes, tabela de símbolos, tabela de relocações, ...
- *Program header table*: informações usadas ao carregar na memória
- *Segments*: conteúdo a carregar em cada área de memória



# Identificação de conteúdo

Como indicar o conteúdo de um arquivo?

- Usar parte do nome do arquivo (“extensão”);
  - `praia.jpg`, `entrevista.mp3`
  - Padrão 8.3 introduzido pelo DOS nos anos 1980
- “Magic numbers”:
  - Bytes predefinidos no início do conteúdo
- Atributos adicionais (“tipo” do arquivo)
  - MacOS 9: *File Type* e *Creator Application* (4 bytes cada)

# Magic Numbers

Tipo de arquivo	bytes iniciais
Imagem PGM ascii	P2\n
Documento PDF	%PDF
Imagem GIF	GIF89a
Imagem JPEG	0xFF D8 FF
Música MIDI	MThd
Classes Java	0xCA FE BA BE
Arquivo ZIP	0x50 4B 03 04
Documento RTF	{\rtf1

# Tipos MIME

## RFC 2046: *Multipurpose Internet Mail Extensions*

Tipo/subtipo MIME	Significado
application/java-archive	Arquivo de classes Java
application/msword	Documento do Microsoft Word
audio/mpeg	Áudio em formato MP3
image/png	Imagem em formato PNG
text/csv	Texto em formato CSV
text/html	Texto HTML
text/plain	Texto puro
text/rtf	Texto em formato RTF ( <i>Rich Text Format</i> )
text/x-csrc	Código-fonte em C

Usado no MacOS, BeOS, anexos de e-mail e no HTTP

# Arquivos especiais

- Abstração de dispositivos de baixo nível:
  - `/dev/ttyS0`: porta de comunicação serial
  - `/dev/sda1`: partição de disco
- Abstração de interfaces do núcleo:
  - `/proc/cpuinfo`: informações sobre processadores
  - `/proc/3754/maps`: mapa de memória do processo 3754
- Canais de comunicação: *sockets* de rede, *pipes*
- Abstrações diversas:
  - `/dev/random`: fonte de bytes aleatórios
  - `/dev/null`: “buraco negro” de dados

SOs *Plan 9* e *Inferno*: todas as entidades são arquivos!