

Sistemas Operacionais



Aula 16

Sistemas de Arquivos

Prof. Ms. Alex Lima

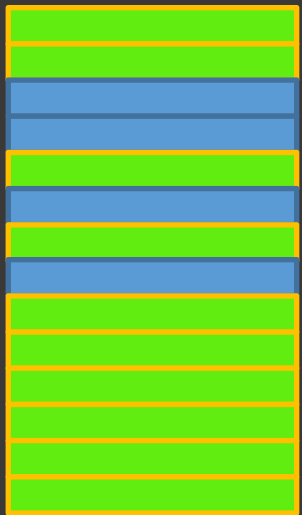
- 01 Memória virtual**
- 02 Mapeamento**
- 03 Paginação**
- 04 Tabela de páginas**
- 05 Memória associativa**

Armazenamento de dados

- **Armazenamento na memória principal**
 - Processos são temporários.
 - A memória principal é volátil e limitada.
 - Processos são propensos a falhas críticas (deadlocks).
- **Arquivos**
 - Arquivos armazenam permanentemente.
 - Memória secundária armazena de forma permanente e possui maior capacidade.
 - Arquivos podem ser compartilhados simultaneamente.

Visão geral

Memória principal



Processo



Programa



Memória externa



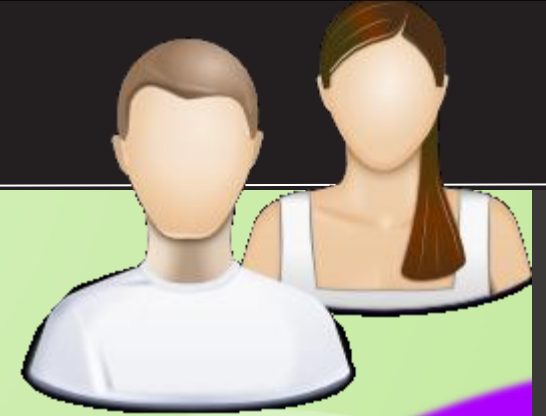
Arquivo

Arquivo

- Conceito
 - **Arquivo** é uma unidade lógica de informação formadas por processos.
 - **Função**: Organização lógica de informações armazenadas de forma persistente.
 - **Sistema de arquivos** são os módulos do sistema operacional responsáveis pelo gerenciamento de arquivos.
 - **Função**: Gerenciar como os arquivos são estruturados, nomeados, acessados, usados, protegidos, implementados e armazenados.

Arquivo

- Conceito



Usuário

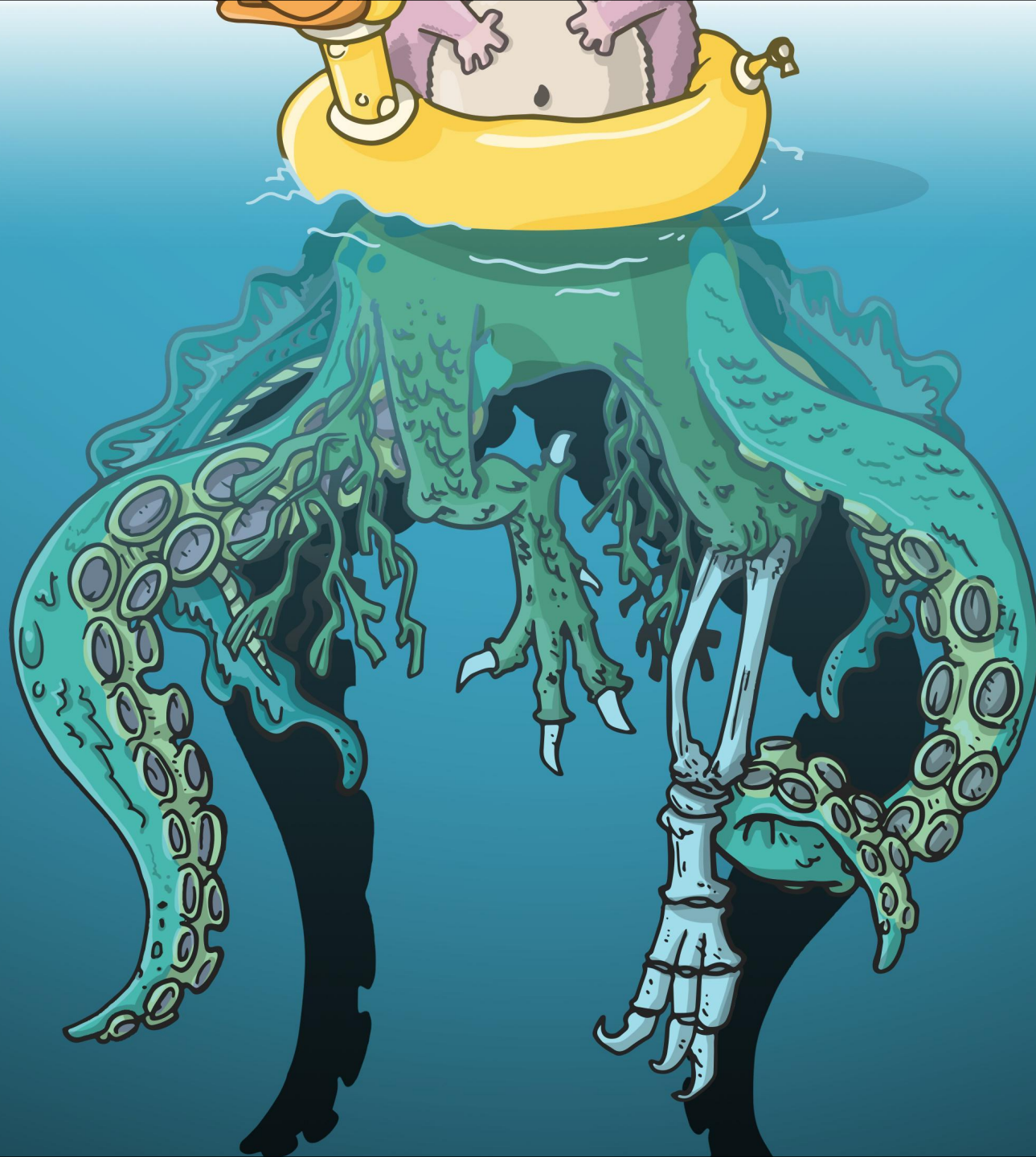
FRONT-END



Arquivo

- Conceito

Sistema



Arquivo

- Sistema operacional

Sistema Operacional	Sistema de arquivos
Linux	EXT3, EXT4, XFS, JFS
MacOS	HFS
Windows	FAT, NTFS
FreeBSD, OpenBSD	UFS
Sun Solaris	UFS
IBM AIX	JFS

Arquivo

- Nomeação de arquivos

- Windows

- FAT-16
 - FAT-32
 - NTFS
 - ReFS (Windows Server)

- Nomeação de arquivos

- UNIX

- UFS
 - FFS
 - Ext4
 - ReiserFS

Arquivo

- **Analizando o Sistema FAT**
 - O sistema FAT16 utiliza 16 bits para endereçar o espaço em disco.
 - $2^{16} = 65536$ clusters
 - Para clusters de 32k temos:
 - $65536 \times 32k = 2GB$ tamanho máximo da partição.

Atividade

- **Analizando o Sistema FAT**
 - Verifique a capacidade máxima de armazenamento do sistema com:
 - clusters de 4k, 8k, e 16k.
 - Faça os cálculos para FAT16 e FAT32.

Arquivo

- Analisando o Sistema NTFS

Tamanho do cluster	Maior volume	Arquivo maior
4 KB (tamanho padrão)	16 TB	16 TB
8 KB	32 TB	32 TB
16 KB	64 TB	64 TB
32 KB	128 TB	128 TB
64 KB (tamanho máximo)	256 TB	256 TB

Arquivo

- Nomeação

- O nome de um arquivo pode ser dividido em 02 partes.

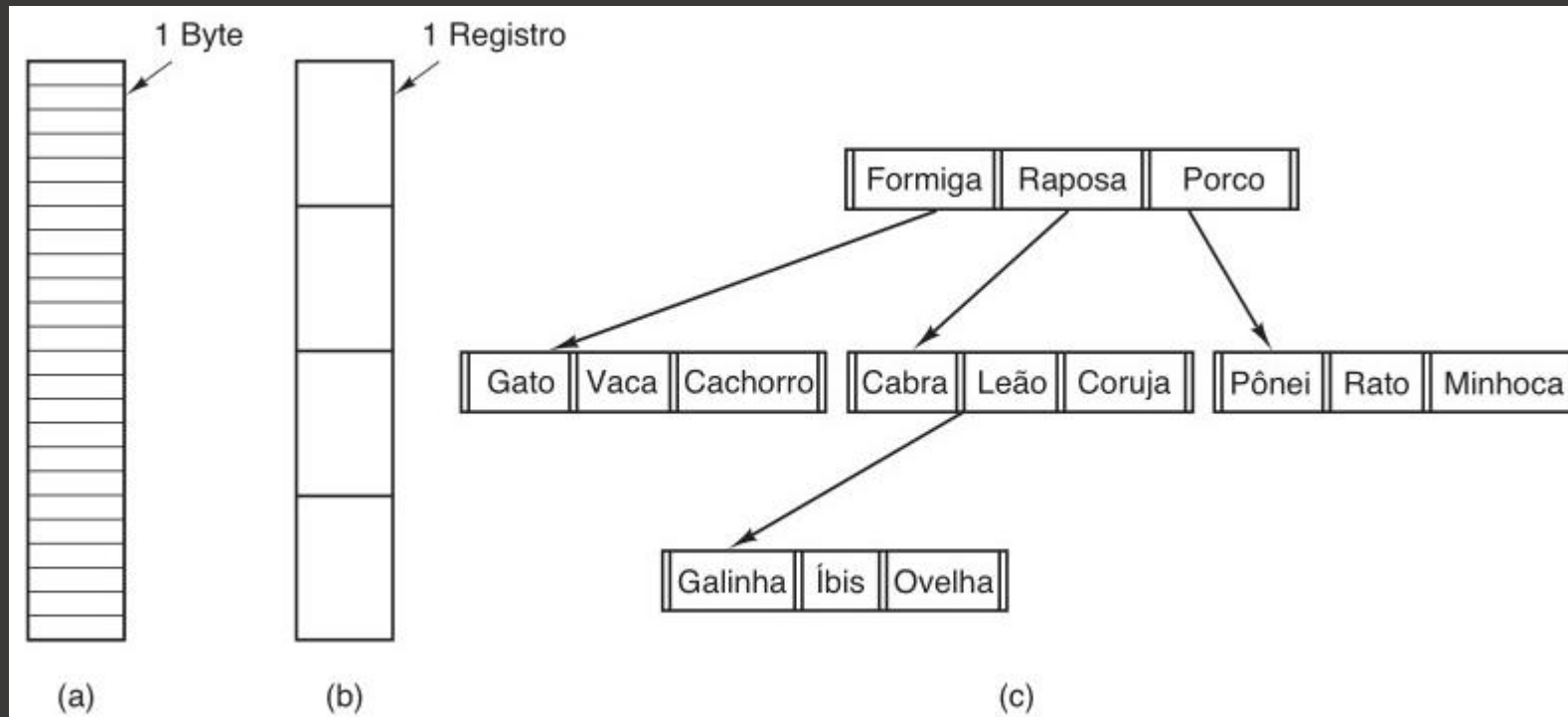
nome.extensão

- Indicam informações úteis ao usuário sobre o arquivo e seu formato.

Extensão	Significado
.bak	Cópia de segurança
.c	Código-fonte de programa em C
.gif	Imagem no formato Graphical Interchange Format
.hlp	Arquivo de ajuda
.html	Documento em HTML
.jpg	Imagem codificada segundo padrões JPEG
.mp3	Música codificada no formato MPEG (camada 3)
.mpg	Filme codificado no padrão MPEG
.o	Arquivo objeto (gerado por compilador, ainda não ligado)
.pdf	Arquivo no formato PDF (Portable Document File)
.ps	Arquivo PostScript
.tex	Entrada para o programa de formatação TEX
.txt	Arquivo de texto
.zip	Arquivo compactado

Arquivo

- Estrutura de arquivos
 - Arquivos podem ser estruturados de 3 formas principais



Arquivo

- Estrutura de arquivos
 - 1. Sequência de bytes
 - O arquivo é visto apenas como uma sequência de bytes.
 - Sua execução é determinada a nível de usuário.
 - Estrutura padrão adotada pelos Unix-like e Windows.

Arquivo

- Estrutura de arquivos
 - **3. Estrutura indexada**
 - O arquivo é visto apenas como uma sequência de bytes.
 - Os registros podem ser de tamanho fixo ou variável.
 - Utilizado em computadores de grande porte e de uso comercial.

Arquivo

- Tipos de arquivos
 - **1. Diretórios**
 - Arquivos do sistema para manter a estrutura do sistema de arquivos.
 - **2. Arquivos regulares**
 - São arquivos que mantêm as informações dos usuários.
 - **3. Arquivos especiais de bloco**
 - Arquivos usados para modelar discos.
 - **4. Arquivos especiais de caracteres**
 - Arquivos relacionados a dispositivos de I/O seriais como terminais, impressoras e redes.

Arquivo

- Tipos de arquivos

- 1. Arquivos regulares

- Arquivos ASCII

- Arquivos de texto
 - Legíveis ao usuário
 - Facilita o pipeline de processos

- Tipos de arquivos

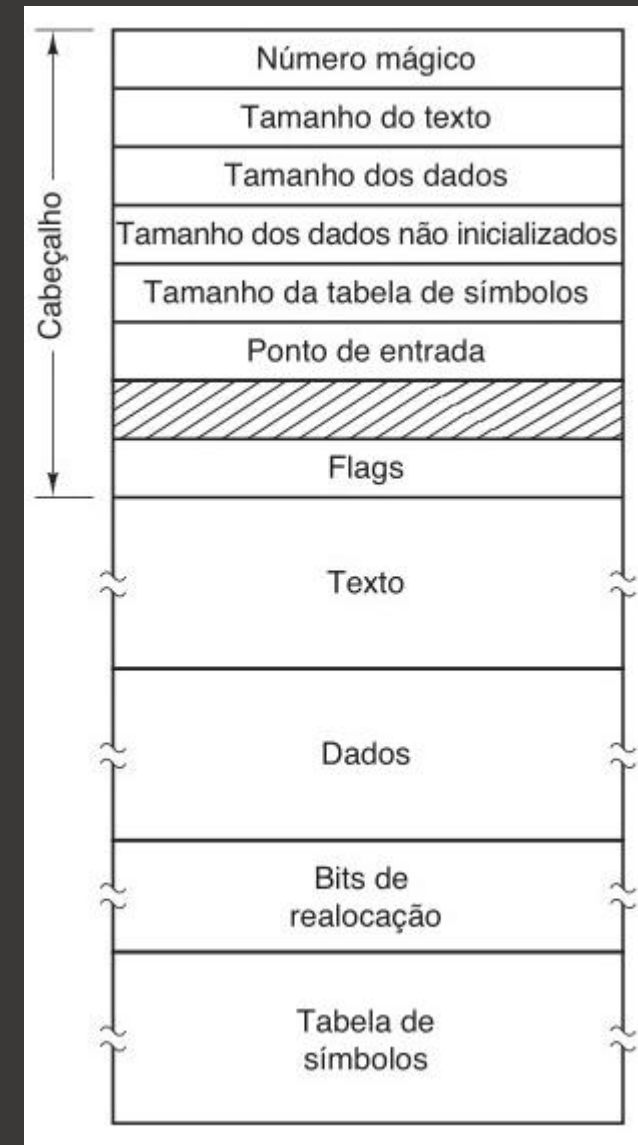
- 1. Arquivos regulares

- Arquivos binários

- Estrutura interna definida pelo programa associado.

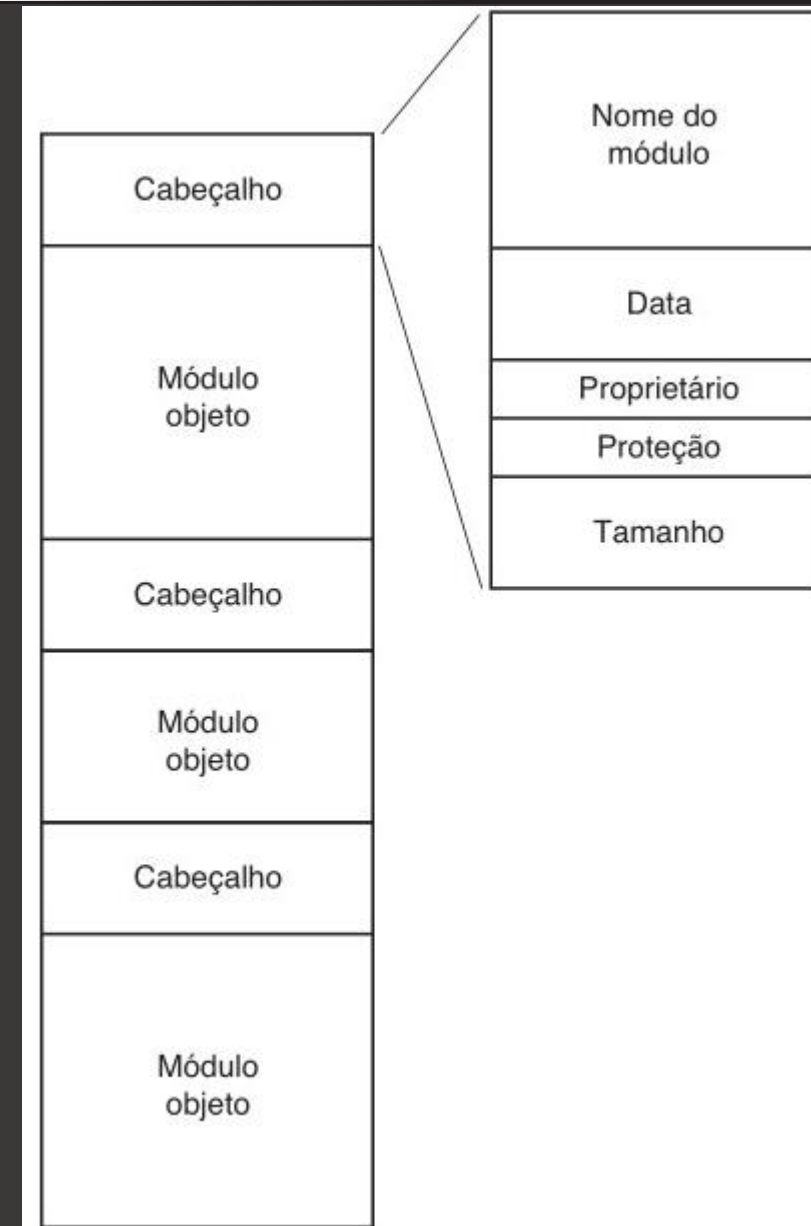
Arquivo

- Tipos de arquivos
 - **Arquivo binário executável**
 - **Número mágico** - Identificador de executável
 - Tamanho dos segmentos
 - Endreço inicial de alocação
 - Segmentos



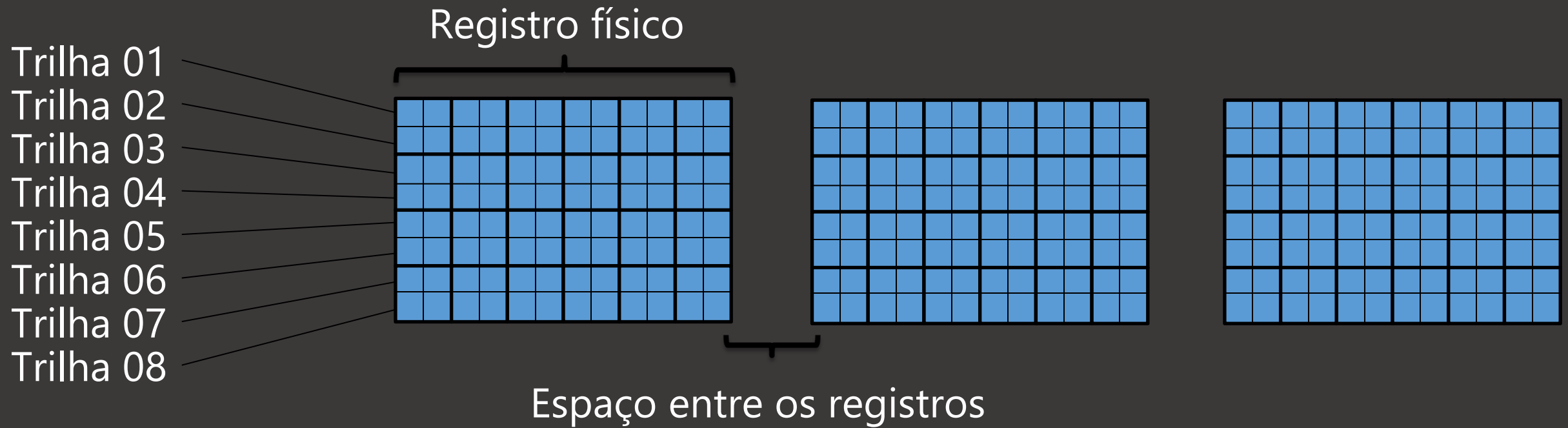
Arquivo

- Tipos de arquivos
 - **Arquivo binário (archieve)**
 - Conjunto de rotinas compiladas e não ligadas(linkage)
 - Cada rotina possui um cabeçalho
 - Cabeçalho
 - Informações sobre os módulos que compoem o arquivo.



Arquivo

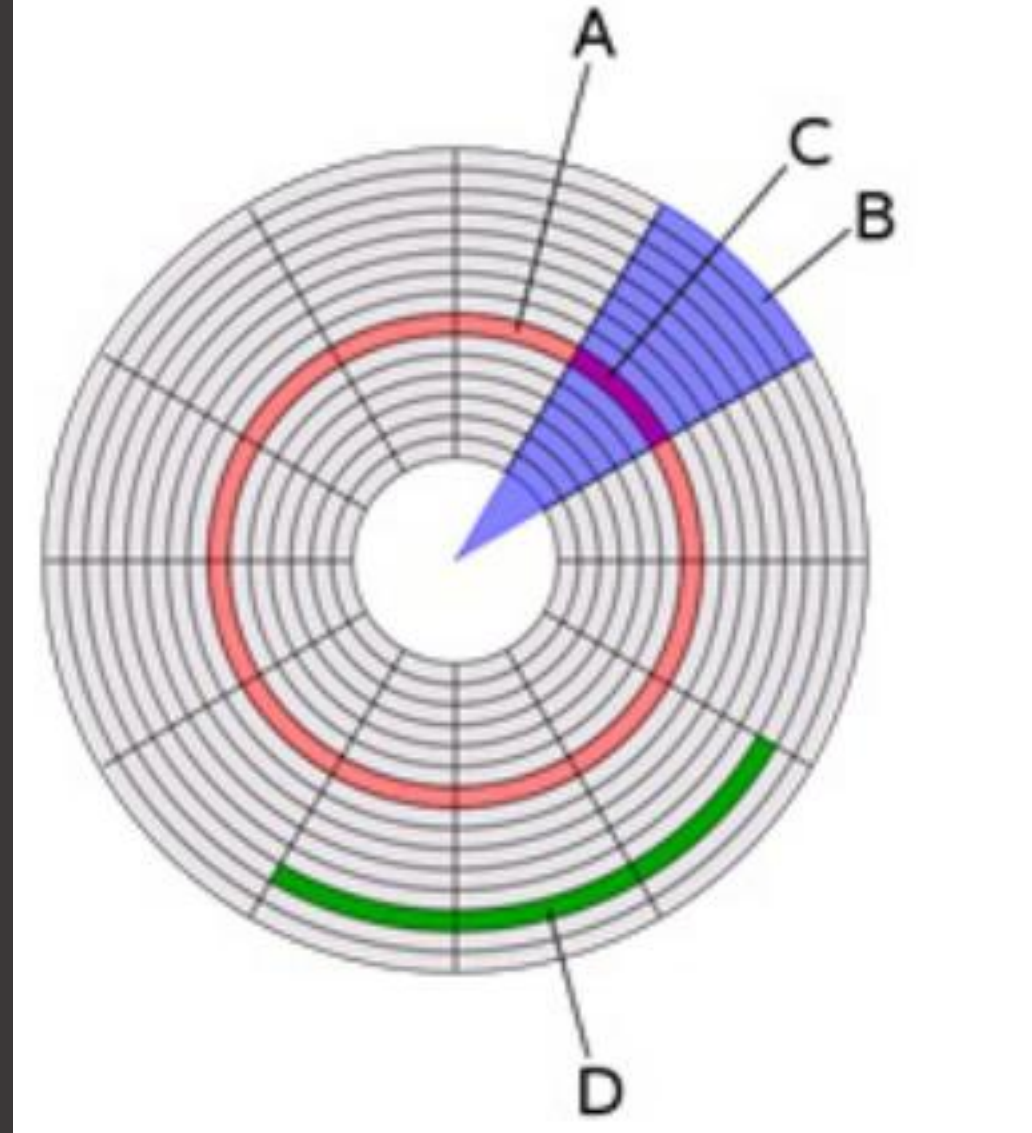
- Métodos de acesso
 - 1. Acesso sequencial



Arquivo

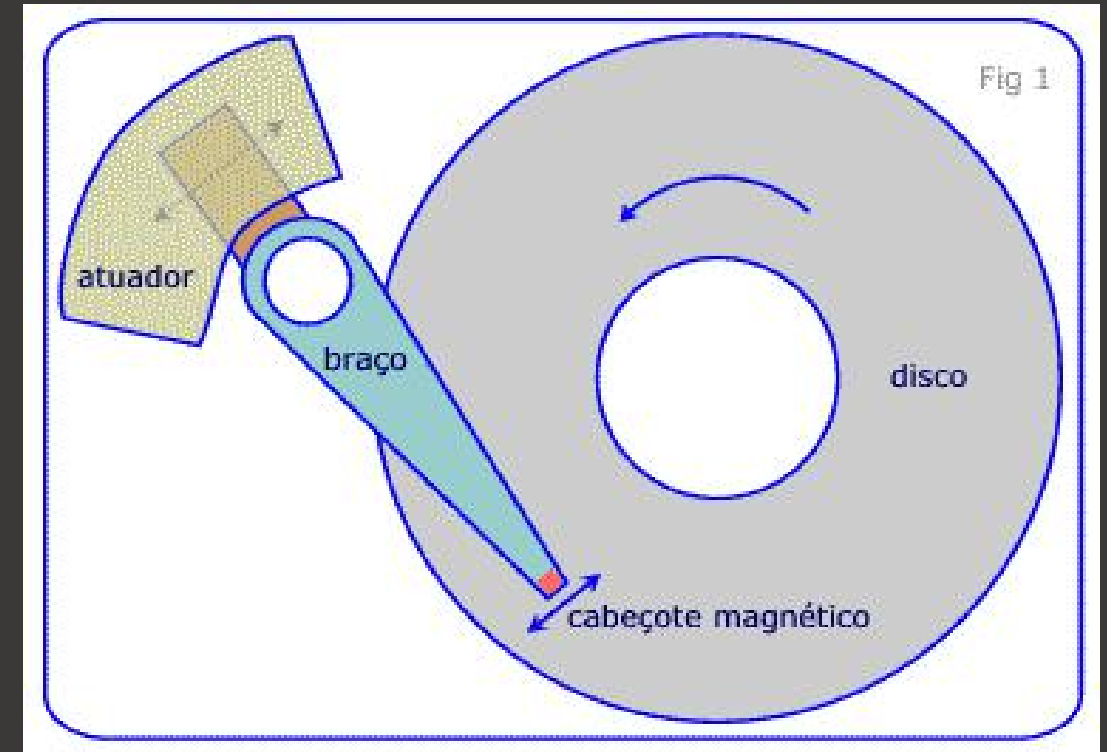
- Acesso ao disco
 - O disco é dividido em trilhas;
 - As trilhas são divididas em setores de 512 B;
 - O sistema de arquivos lida com conjuntos de setores, chamados *clusters*.

A – Trilha
B – Setor geométrico
C – Setor de trilha
D – Unidade de alocação



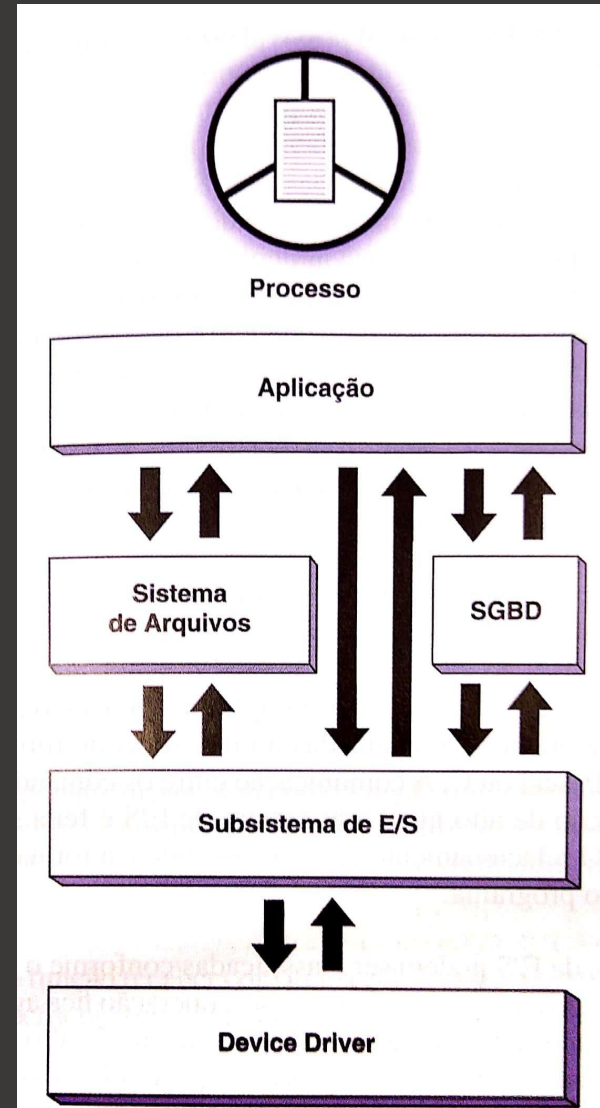
Arquivo

- Métodos de acesso
 - 2. Acesso direto
 - **ALERTA!** Não confundir método de acesso aleatório (usado em memória principal) com arquivos de acesso aleatório.
 - **Arquivos de acesso aleatório** são assim chamados por que podem ser acessados em qualquer ordem.
 - Método de acesso ao disco é acesso direto.



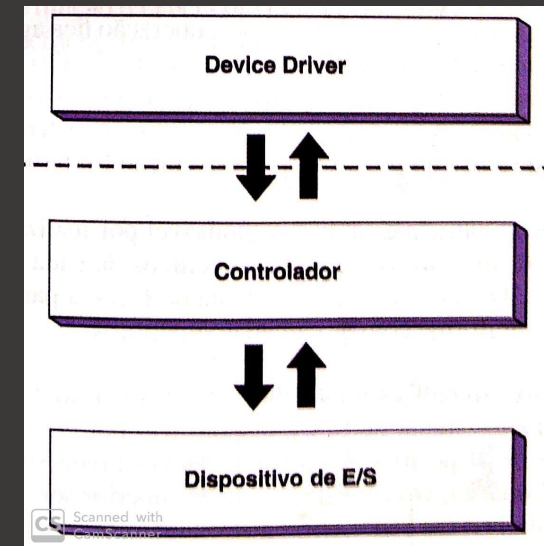
Device Driver

- Subsistema de E/S
 - O subsistema de E/S realiza a comunicação entre aplicações e dispositivos de entrada e saída.
 - O device driver, ou simplesmente driver, realiza a tradução dos comandos do subsistema de E/S para o controlador do dispositivo.



Device Driver

- Subsistema de E/S
 - As aplicações trabalham em modo usuário.
 - O sistema operacional e os drivers em modo kernel
 - O controlador e os dispositivos de E/S são dispositivos físicos.



Arquivo

- Propriedades de arquivos
 - 1. Atributos
 - Informações e características específicas de cada arquivo.
 - A lista de atributos varia para cada SO.

```
C:\WINDOWS\system32\cmd.exe

C:\Users\Alex>help attrib
Exibe ou altera atributos de arquivo.

ATTRIB [+R | -R] [+A | -A] [+S | -S] [+H | -H] [+O | -O] [+I | -I] [+X | -X] [+P | -P] [+U | -U]
[unidade:][caminho][nome_de_arquivo] [/S [/D]] [/L]

+ Define um atributo.
- Limpa um atributo.
R Atributo de arquivo somente leitura.
A Atributo de arquivo morto.
S Atributo de arquivo do sistema.
H Atributo de arquivo oculto.
O Atributo offline.
I Atributo de arquivo sem conteúdo indexado.
X Sem atributo de arquivo de remoção.
V Atributo de integridade.
P Atributo fixado.
U Atributo desafixado.
B Atributo de Blob SMR.
[unidade:][caminho][nome_de_arquivo]
Especifica um ou mais arquivos para o atributo processar.
/S Processa os arquivos correspondentes na pasta atual
e em todas as subpastas.
/D Inclui pastas no processamento.
/L Trabalha nos atributos do Link Simbólico versus
o destino do Link Simbólico

C:\Users\Alex>
```


Arquivo

- Operações em arquivos
 - O sistema operacional pode fornecer chamadas de sistema para: criar, gravar, ler, reposicionar, apagar e truncar arquivos
 - Criando um arquivo
 - Aloca-se espaço para o arquivo no sistema de arquivos
 - Cria-se uma entrada para o novo arquivo no diretório
 - Gravando um arquivo
 - O sistema mantém um ponteiro de gravação onde a próxima gravação será feita
 - O ponteiro de gravação é atualizado sempre que ocorre uma escrita

Arquivo

- Operações em arquivos
 - Lendo um arquivo
 - O sistema mantém um ponteiro de leitura demarcado onde ocorrerá a próxima leitura dentro do arquivo. Após a leitura, o ponteiro deve ser atualizado
 - Normalmente o ponteiro de gravação e o ponteiro de leitura são mantidos pelos SO's como um ponteiro só denominado ponteiro de posição corrente.
 - Reposicionando dentro de um arquivo
 - O ponteiro de posição corrente é alterado para um determinado valor
 - Essa operação também é conhecida como busca (seek) de arquivo

Arquivo

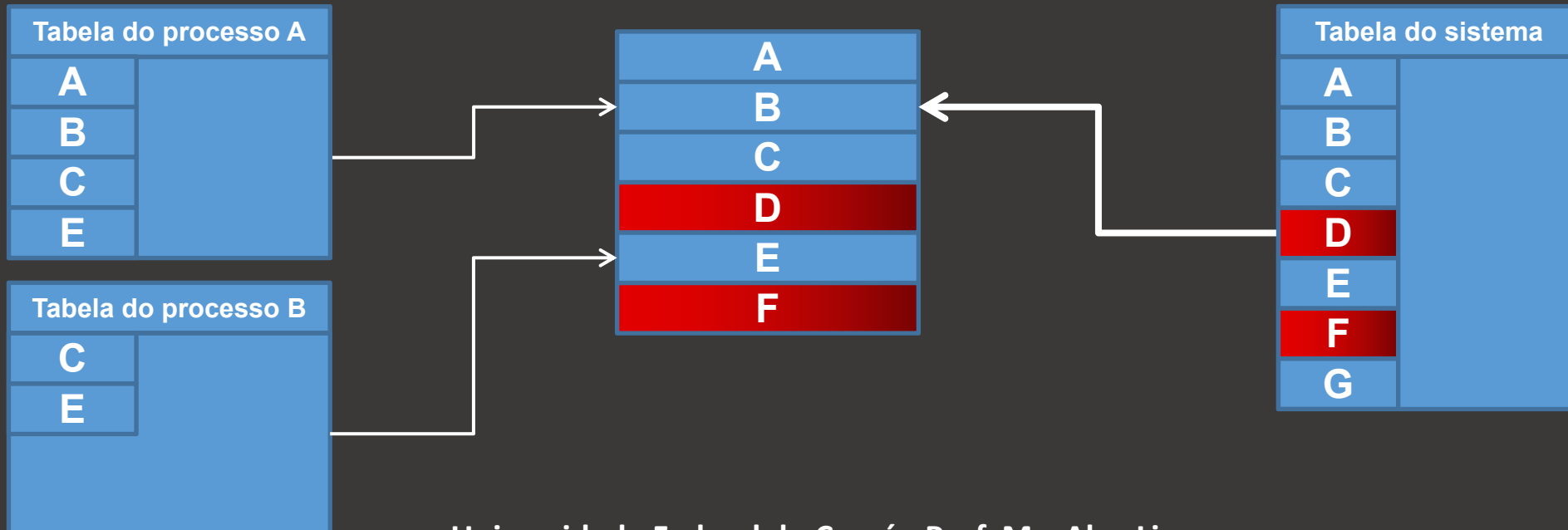
- Operações em arquivos
 - Apagando um arquivo
 - Procura-se no diretório o arquivo pelo nome
 - Libera-se o espaço alocado ao arquivo para que possa ser utilizado por outros
 - Apaga-se a entrada do arquivo no diretório

Arquivo

- Operações em arquivos
 - A maioria das operações envolve a busca do arquivo no diretório.
 - Para evitar uma busca constante, SO's possuem uma chamada de sistema `open()`
 - O SO mantém uma tabela de arquivos abertos contendo informações dos arquivos abertos
 - Quando uma operação em um arquivo é solicitada, o arquivo é especificado com um índice nessa tabela
 - Quando o arquivo não for mais necessário pelo processo, uma chamada de sistema `close()` remove a entrada do arquivo na tabela de arquivos abertos

Arquivo

- Operações em arquivos
 - Tabela de arquivos abertos
 - Arquivos podem ser utilizados por um ou mais processos.

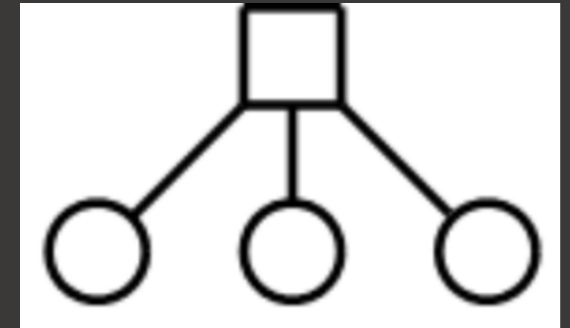


Diretório

- **Diretório** é uma estrutura de dados que contém entradas associadas aos arquivos em que cada entrada armazena informações como localização física, nome, organização e demais atributos.
 - Esquemas mais comuns para definir a estrutura lógica de um diretório
 - ✓ Diretório de um nível
 - ✓ Diretório de dois níveis
 - ✓ Diretórios estruturados em árvore
 - ✓ Diretórios como estruturas de grafo acíclico

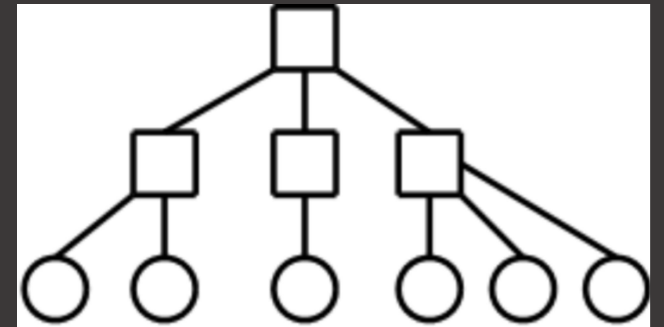
Diretório

- **Diretório de um nível**
 - Um único diretório contendo todos os arquivos
 - Utilizado nos primeiros computadores
 - **Vantagens**
 - Simplicidade
 - Capacidade de localizar os arquivos rapidamente
 - **Desvantagens**
 - Limitação na escolha de nomes de arquivos
 - Arquivos de vários usuários no mesmo lugar
 - Falta de organização



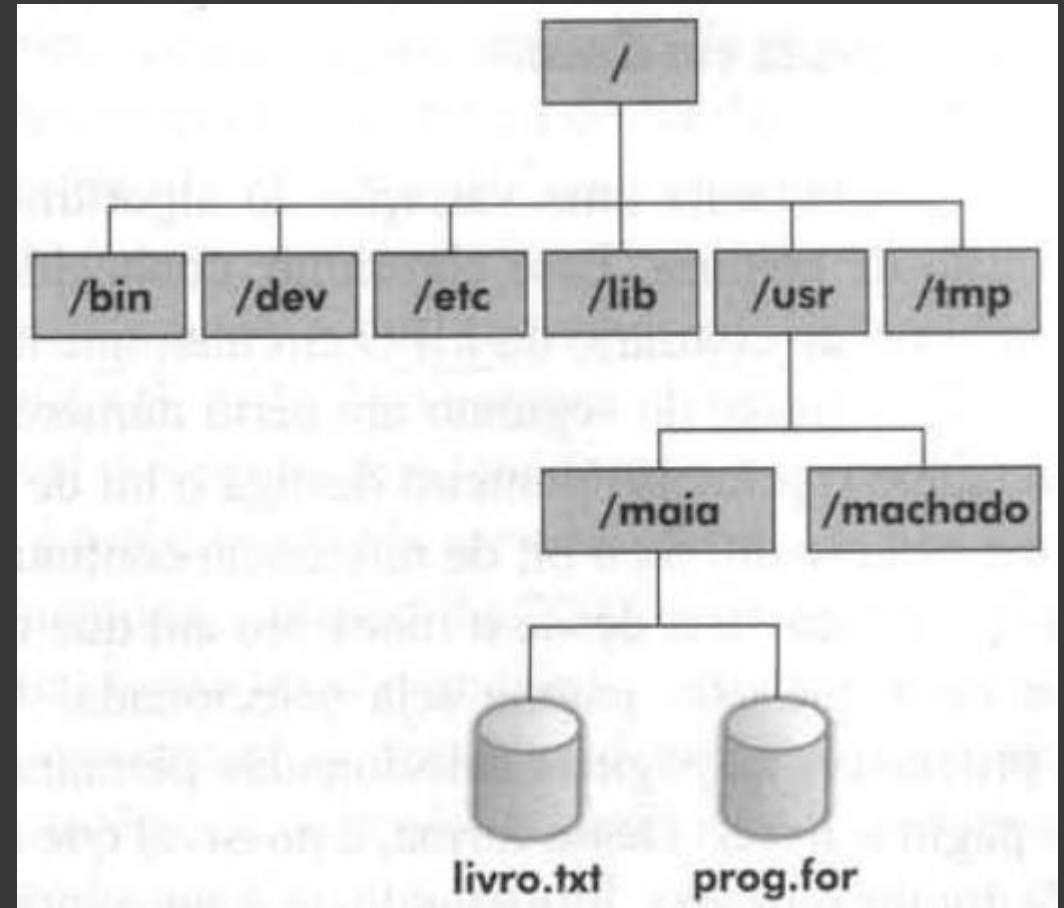
Diretório

- **Diretório de dois níveis**
 - Cada usuário tem seu próprio diretório de arquivos
 - **UFD = User File Directory**
 - Os diretórios de usuário são internos ao diretório de arquivos mestre
 - **MFD = Master File Directory**
 - Quando um usuário referencia um arquivo, apenas seu próprio UFD é pesquisado
 - Usuários diferentes podem ter arquivos com o mesmo nome.
 - Esta organização é considerada uma árvore de dois níveis.



Diretório

- **Diretório estruturado em árvore**
 - Organização de diretórios como uma árvore.
 - Permite que o usuário criem seus próprios subdiretórios e organizem seus arquivos hierarquicamente.
 - Um bit em cada entrada do diretório define a entrada como arquivo (0) ou como subdiretório (1).



Diretório

- **Gerenciamento de espaço livre em disco**
 - O gerenciamento de espaço livre do disco utiliza estruturas de dados para manter o controle sobre quais blocos do disco estão livres e quais estão ocupados.
- As soluções clássicas são:
 - ✓ Mapa de bits
 - ✓ Lista encadeada
 - ✓ Tabela de blocos livres

Diretório

- Gerenciamento de espaço livre em disco
 - O espaço livre em disco pode ser gerenciado por uma tabela denominada **Mapa de Bits (Bitmap)**.
 - Na **bitmap**, cada bloco é representado por um bit.
 - Bit 1 indica que o bloco está ocupado.
 - Bit 2 indica que o bloco está livre.

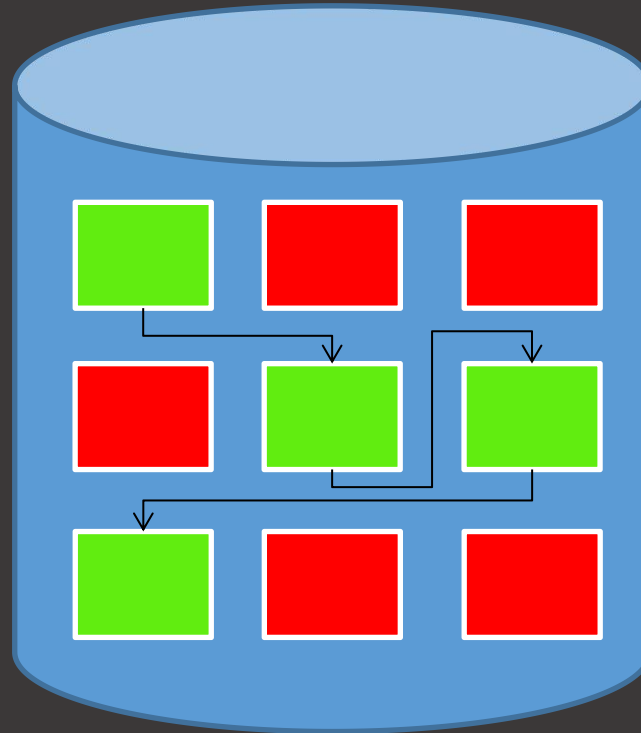
Trilha	Blocos
1	010101010101010101110
2	01010101101010101110
3	111110010101001101100
4	010001001000100010001
5	101010101010101010101

Diretório

- Gerenciamento de espaço livre em disco
 - Uma lista encadeada pode ser usada para conectar os blocos livres no disco.
 - Quando necessário, uma busca linear pode ser realizada para se encontrar espaços livres ou verificar se um dado espaço está ocupado.
 - Esta solução apresenta menor custo de memória que o *bitmap*.

Diretório

- Gerenciamento de espaço livre em disco
 - O espaço livre em disco pode ser gerenciado por uma lista encadeada.



Diretório

- Gerenciamento de espaço livre em disco
 - Tabela de blocos livres
 - A tabela armazena uma entrada para cada segmento de blocos livres.
 - A tabela armazena o número do primeiro bloco do segmento e o número de blocos livres.

Bloco	Contador
4	2
10	1
13	7
25	20
50	5

Diretório

- Gerenciamento de espaço livre em disco
 - Tabela de blocos livres



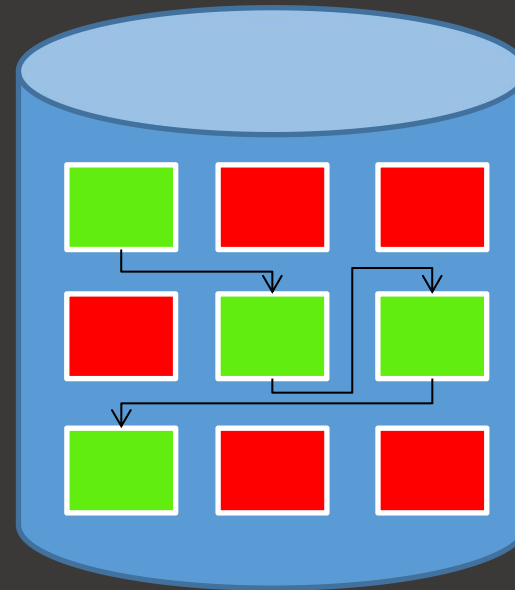
Bloco	Contador
5	4
12	2
15	2

Diretório

- Gerenciamento de espaço livre em disco

Trilha	Blocos
1	010101010101010101110
2	010101011010101011110
3	111110010101001101100
4	010001001000100010001
5	101010101010101010101

Bitmap



Lista encadeada

Bloco	Contador
5	4
12	2
15	2

Tabela de blocos livres

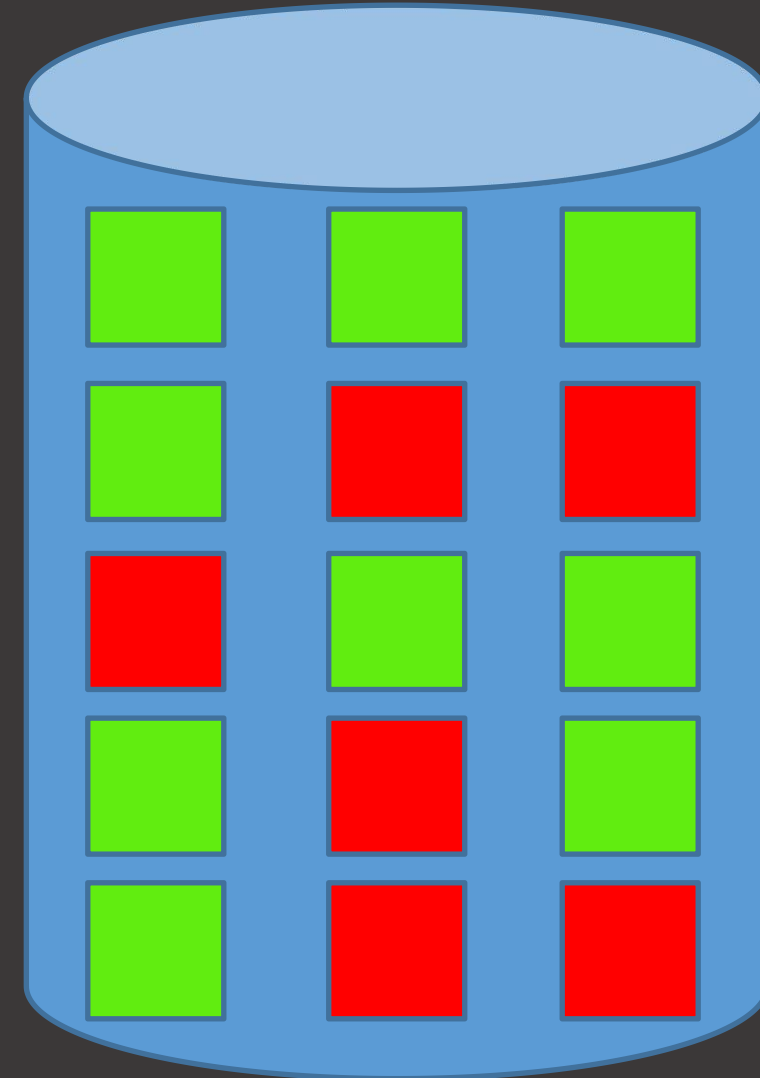
Diretório

- **Gerenciamento de alocação em disco**
 - Assim como o gerenciamento do espaço livre, o gerenciamento do espaço alocado em disco é de fundamental importancia ao SO.
 - As principais técnicas de gerenciamento de espaço alocado são:
 - Alocação contígua
 - Alocação encadeada
 - Alocação indexada

Diretório

- **Gerenciamento de alocação em disco**
- Alocação contígua
 - Os arquivos são alocados de forma contígua na memória externa.
 - O sistema localiza um arquivo por meio de uma tabela que contém o endereço do primeiro bloco e a sua extensão em blocos.

Arquivo	Bloco	Extensão
A.txt	4	3
B.jpg	10	1
C.mkv	13	2



Diretório

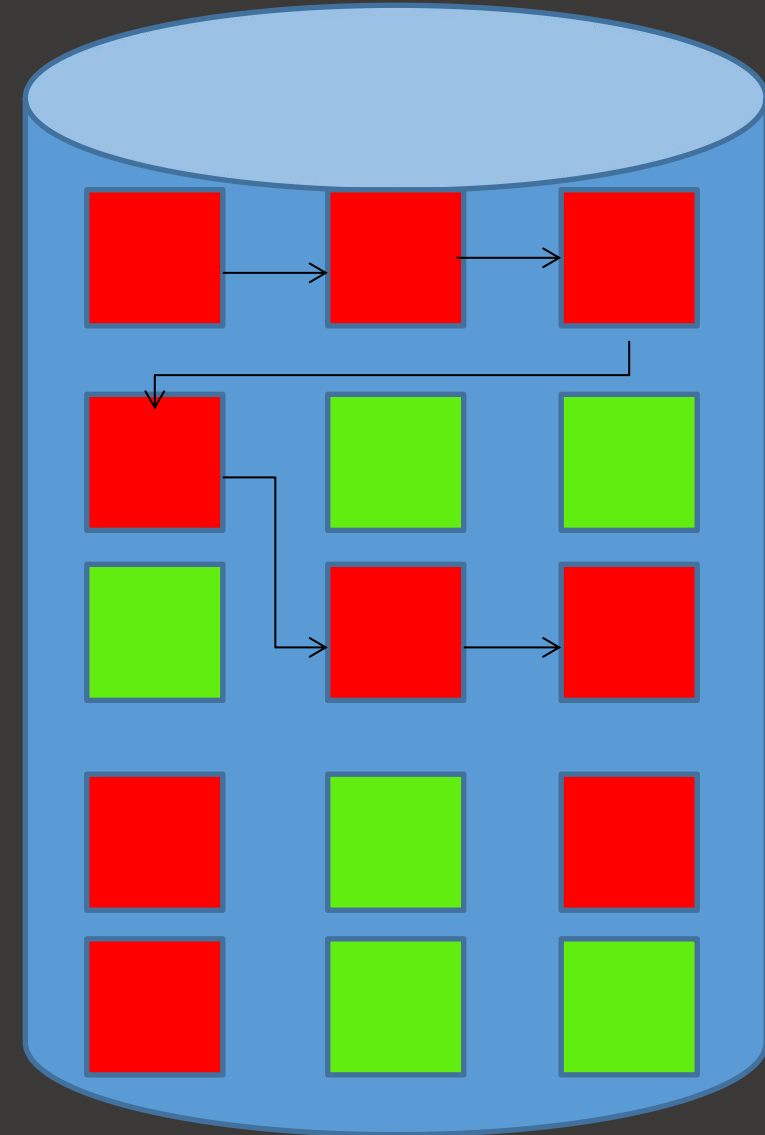
- Gerenciamento de alocação em disco
- Alocação contígua
 - Política de alocação
 - **First-fit** - Aloca no primeiro espaço livre disponível.
 - **Best-fit** - Aloca no menor espaço livre capaz de armazenar o arquivo.
 - **Worst-fit** - Aloca no maior espaço livre disponível.
- Caso haja espaço livre suficiente para o arquivo, mas este não seja contíguo, o disco pode ser desfragmentado para reorganizar o espaço no disco.

Diretório

- **Gerenciamento de alocação em disco**
- Alocação encadeada
 - Os arquivos são organizados como um conjunto de blocos conectados logicamente.
 - Cada bloco deve conter um ponteiro para o bloco seguinte.
 - **Do ponto de vista lógico**, a fragmentação não gera problema algum nesse sistema.
 - **Do ponto de vista físico**, arquivos fragmentados geram perda de desempenho.

Diretório

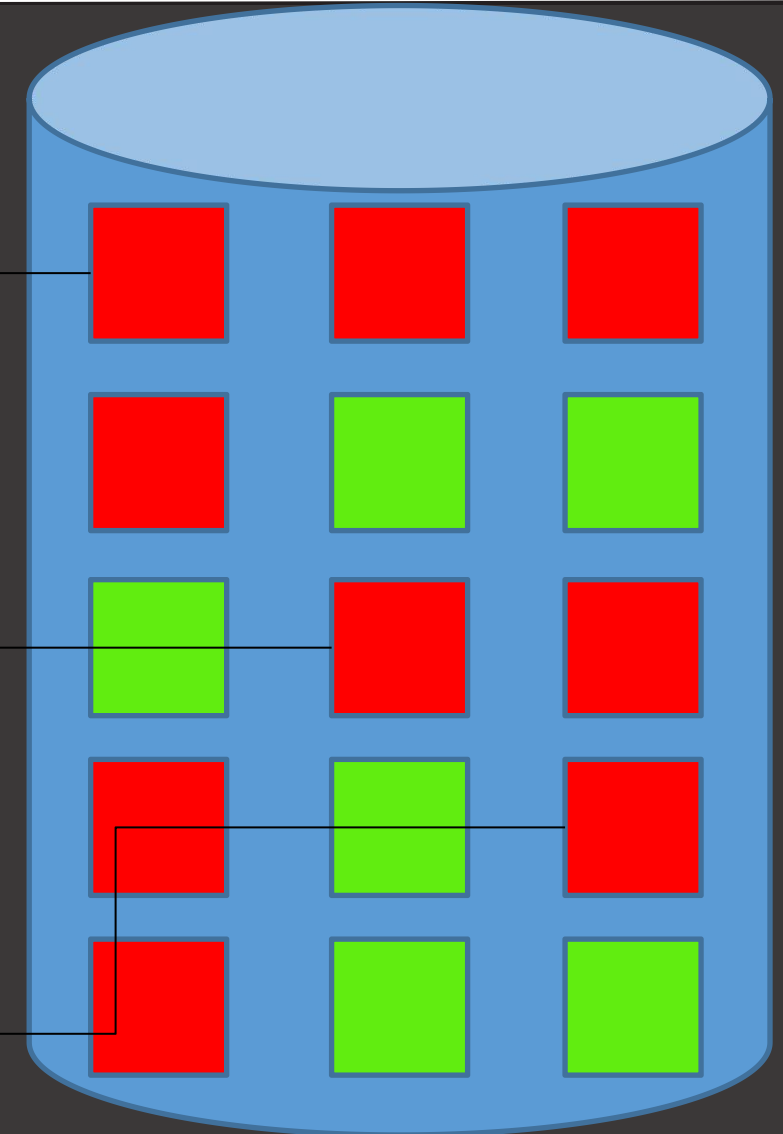
- Gerenciamento de alocação em disco
 - Alocação encadeada
 - A tabela usada na alocação contígua é substituída por ponteiros entre os blocos



Diretório

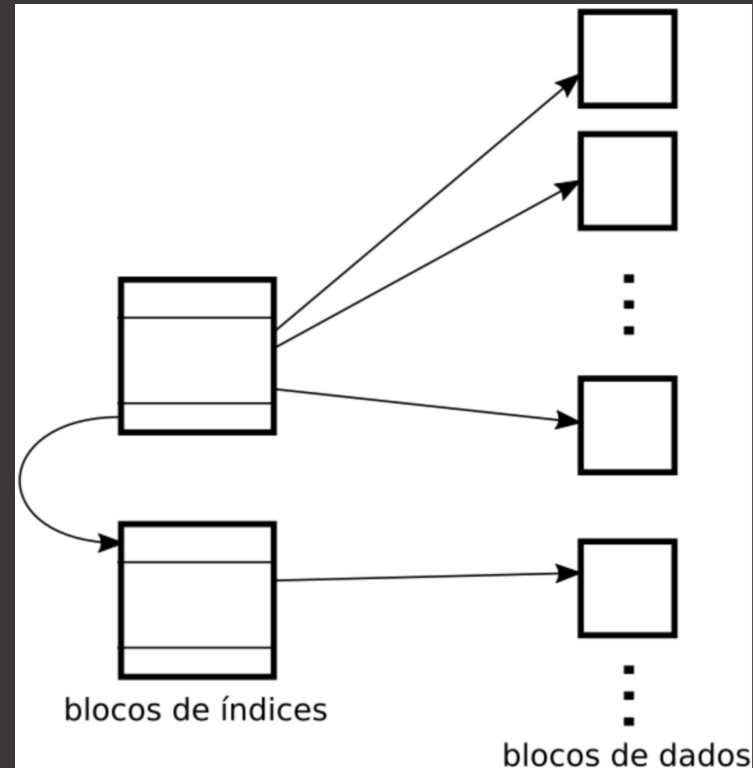
- Gerenciamento de espaço livre em disco
 - Alocação indexada
 - A proposta desta técnica é manter todos os segmentos de blocos do arquivo em uma estrutura chamada bloco de índice, também conhecido como i-node.

A.txt	↓
A.txt	←
A.txt	←



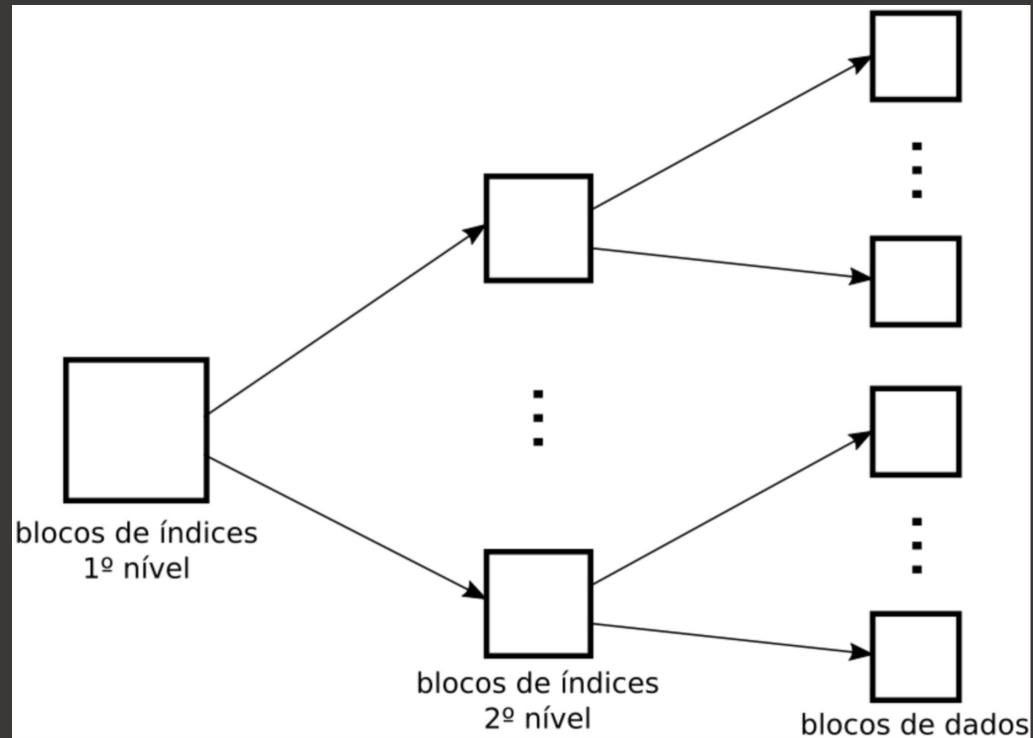
Diretório

- Gerenciamento de alocação em disco
- Alocação indexada
 - Solução 1 → Esquema encadeado



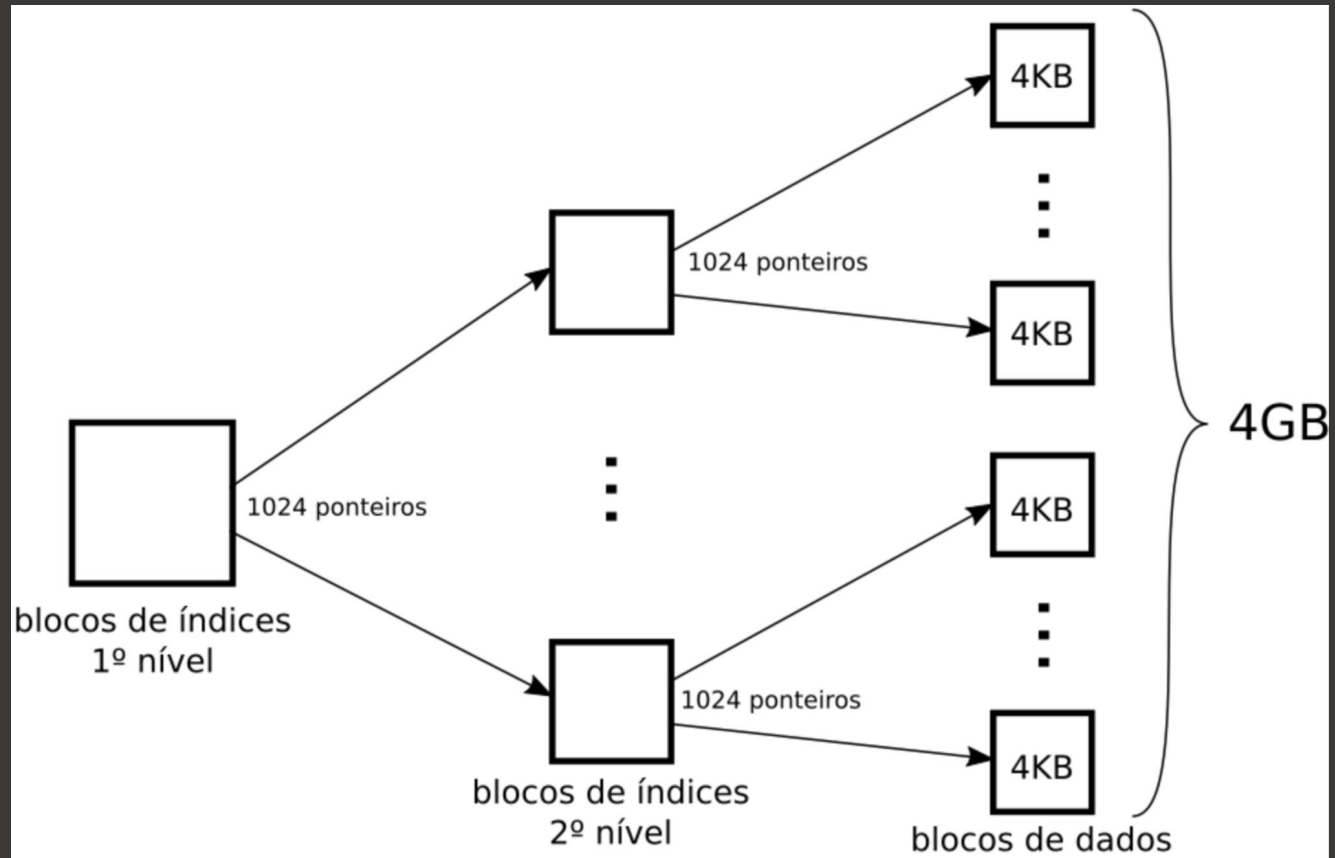
Diretório

- Gerenciamento de alocação em disco
- Alocação indexada
 - Solução 2 → Esquema multinível



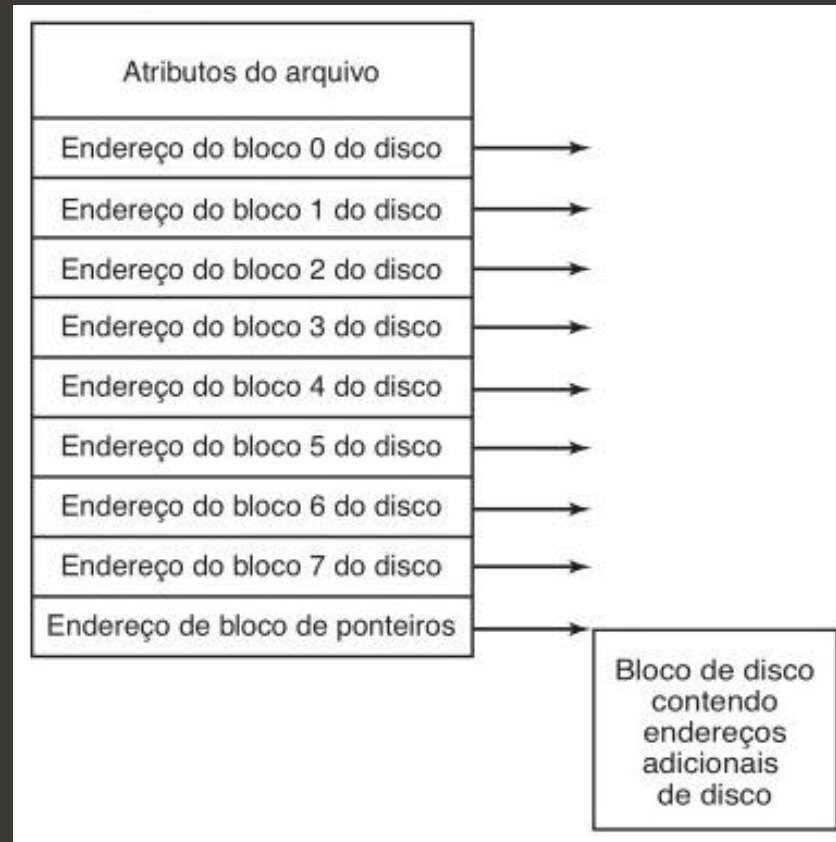
Diretório

- Gerenciamento de alocação em disco
- Alocação indexada
 - Solução 2 → Esquema multinível



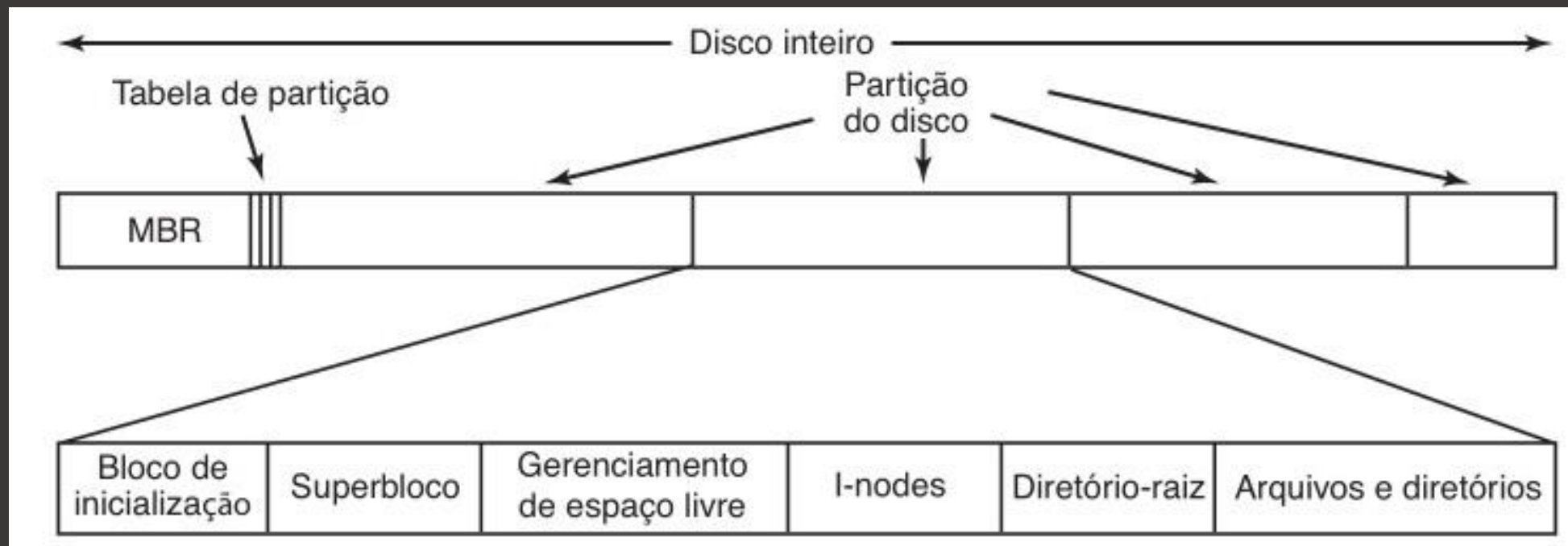
Diretório

- Gerenciamento de alocação em disco
- Alocação indexada
 - i-nodes de um nível



Sistema de arquivos

- Esquema de um sistema de arquivos



Atividade

- Gerenciamento de alocação em disco
- Alocação contígua
 - Onde o arquivo A deve ser armazenado segundo cada política de alocação?