

---

# MAC5753 - Sistemas Operacionais

Daniel Macêdo Batista

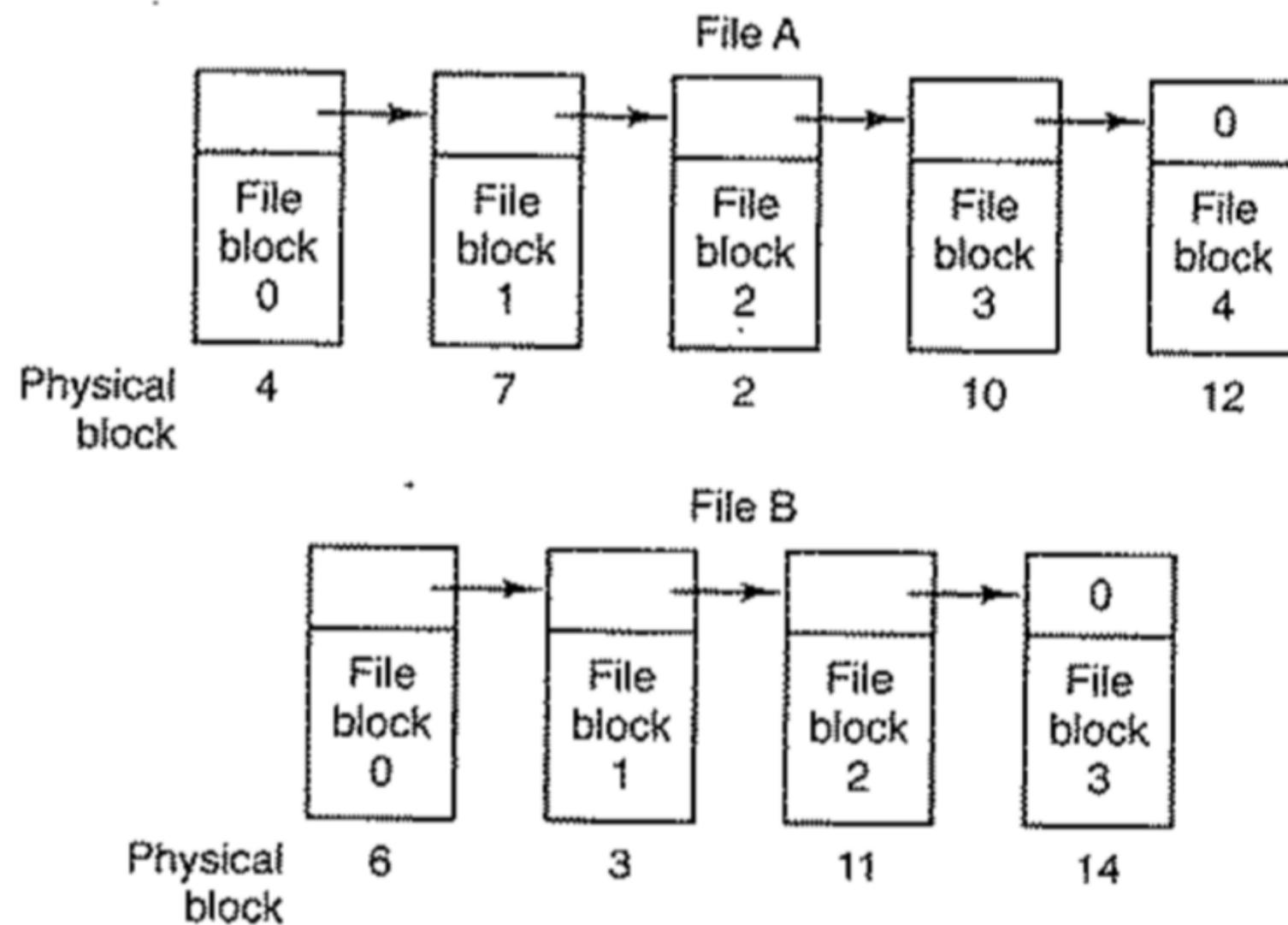
IME - USP, 22 de Outubro de 2020

## Armazenamento de arquivos

# Armazenamento de arquivos

# Alocação por lista encadeada

- O início de um bloco aponta para o próximo bloco e o resto do bloco contém os dados



# Alocação por lista encadeada

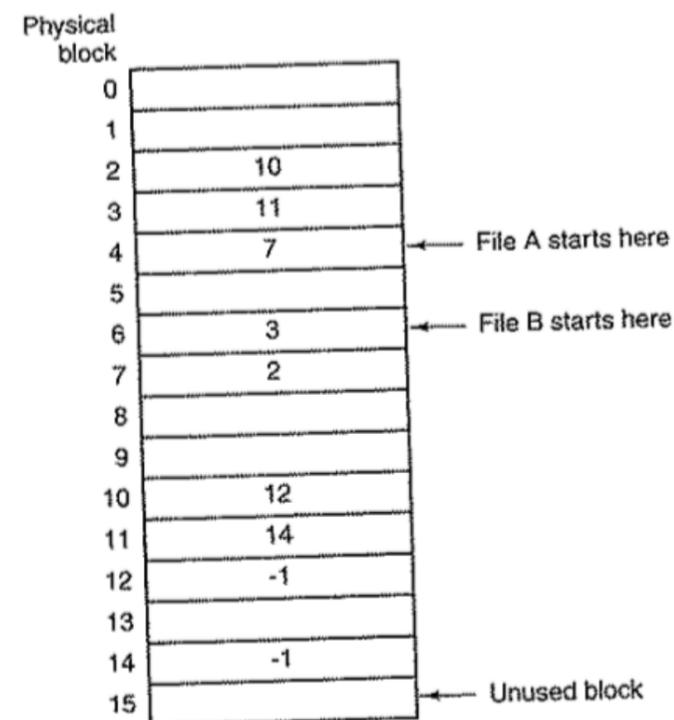
- Vantagens:
  1. Cada bloco do disco pode ser usado
  2. Não perde espaço com fragmentação
  3. A entrada para o arquivo no diretório onde ele está basta armazenar o endereço do primeiro bloco

# Alocação por lista encadeada

- Desvantagens:
  1. A leitura tende a ser bem lenta pois para chegar no bloco  $n$  é necessário navegar em todos os  $n - 1$  anteriores
  2. Perde parte do espaço do bloco para guardar o ponteiro para o próximo → Chamadas de sistemas preparadas para ler múltiplos do tamanho de 1 bloco podem ter desempenho ruim (teria que ler dois blocos ao invés de 1)

# FAT

- Seria possível eliminar as desvantagens da alocação por lista encadeada se o ponteiro de cada bloco no sistema de arquivos fosse armazenado em uma tabela na memória (O FAT faz isso)
- Na verdade vai ter lentidão inicial para carregar tudo na memória mas apenas no início



- Vantagens:
  1. Como no caso anterior, o disco inteiro fica disponível para ser usado
  2. O acesso aleatório fica mais fácil e rápido pois apesar de ter que percorrer os  $n-1$  blocos anteriores, a cadeia entre eles está na memória
  3. Como no caso anterior, as entradas no diretório só precisam armazenar um único número que vai ser o primeiro bloco de cada arquivo

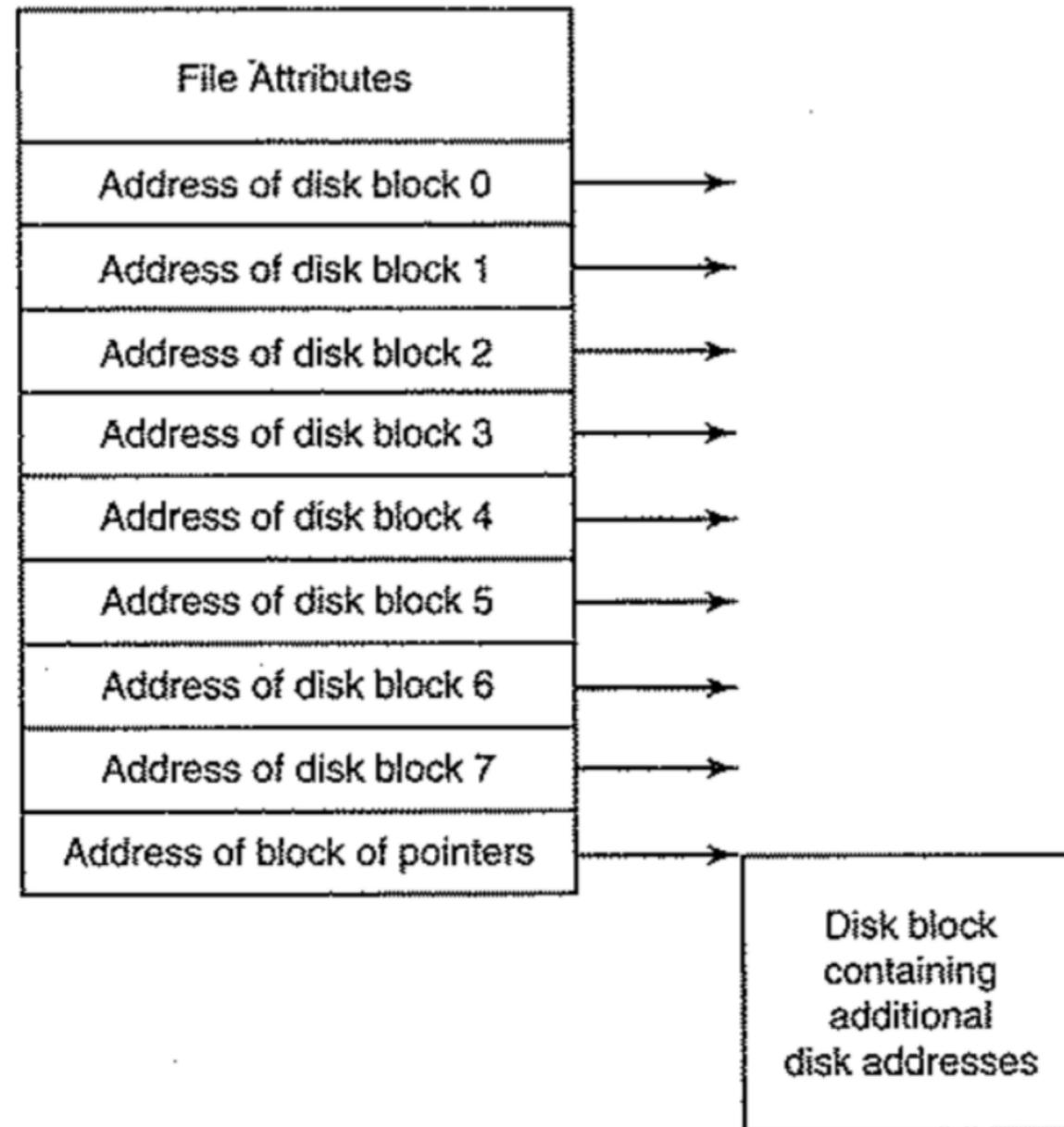
- Desvantagem:
  1. A tabela inteira tem que ficar guardada na memória o tempo inteiro (Exemplo: disco de 200GB com blocos de 1KB precisará de 200 milhões de entradas na tabela. Para endereçar 200 milhões de entradas com os ponteiros, precisa-se de 4 bytes por entrada, assim a tabela ocupará 800MB da memória inteira) → Não escala para discos grandes

# i-nodes

- Cada arquivo é associado a uma estrutura de dados chamada i-node (index-node).
- Um i-node lista os atributos e endereços de disco dos blocos do arquivo
- Dado um i-node é possível encontrar todos os blocos de um arquivo

# i-nodes

Armazenamento de arquivos



- Vantagens:
  1. Em relação ao FAT é que o i-node só precisa estar na memória quando o arquivo associado estiver aberto  $\rightarrow$  1 i-node ocupando  $n$  bytes e se o SO aceita até  $k$  arquivos abertos ao mesmo tempo, então usará  $n \times k$  bytes de memória
  2. O espaço em memória do FAT é proporcional ao tamanho do disco. O espaço em memória usando i-nodes é proporcional à quantidade máxima de arquivos abertos

- Desvantagens:
  1. O tamanho do i-node, para facilitar o acesso, deve ser considerado fixo, mas isso limita o tamanho máximo de um arquivo → pode resolver usando o último endereço de bloco apontando para um endereço de disco que tenha mais endereços de bloco