



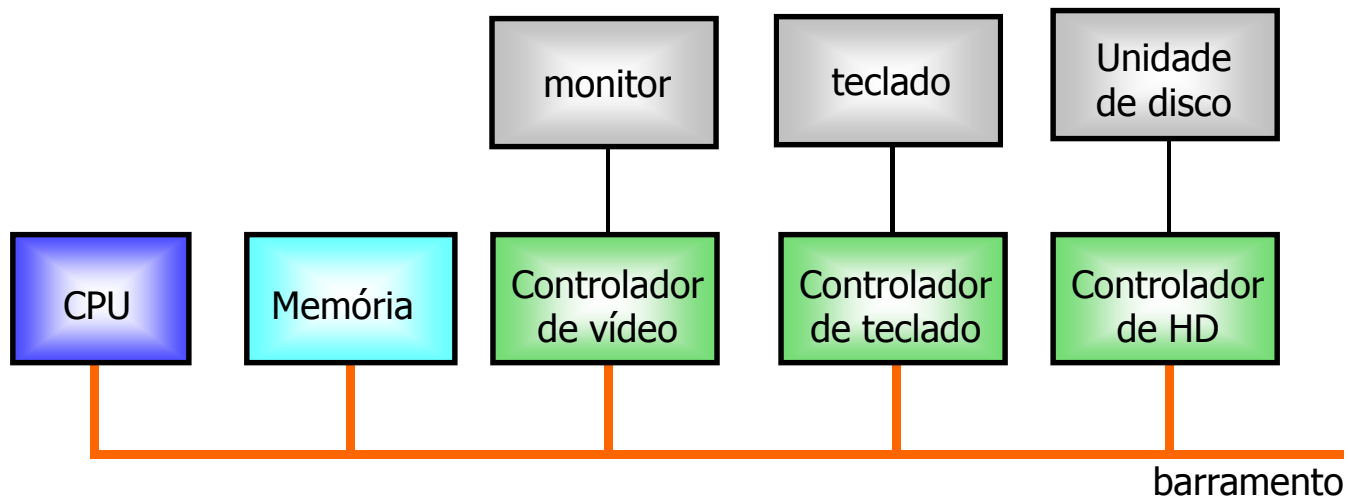
Sistemas Operacionais

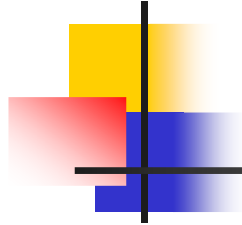
Unidade Quatro

Prof. Flávio Márcio de Moraes e Silva

Estrutura de Sistema de Computação

Um sistema de computação moderno





Periféricos

- Dispositivos externos que proporcionam a troca de informação com o computador.
- Não estão conectados diretamente ao barramento. Mas sim através de um dispositivo de hardware denominado interface.
- Interfaces empregam em seu projeto um dispositivo de hardware denominado controlador.
- O controlador implementa as operações de ler, escrever, reinicializar, ler status ou escrever comando. Transformando-as em suas respectivas ações eletrônicas, elétricas e mecânicas.



Controladores de Dispositivos

- As unidades de E/S constituem-se de um componente mecânico e um eletrônico
- O componente eletrônico é chamado de **controlador do dispositivo**
 - Nos PCs apresenta-se como uma placa controladora
- A placa controladora possui um conector no qual pode ser plugado um cabo que a conecta ao dispositivo
 - Empresas desenvolvem controladoras ou dispositivos para uma interface padrão



Controladores de Dispositivos

- O trabalho da controladora é converter o fluxo serial de bits em um bloco de bytes e executar a correção de erros necessária
- O bloco é montado bit a bit no *buffer* da controladora



Controladores de Dispositivos

- Dispositivos de E/S e a CPU podem executar de forma concorrente
- Cada controladora de dispositivo cuida de um tipo específico de dispositivo
- Cada controladora de dispositivo tem um buffer local
- A CPU transfere dados entre o buffer da controladora e a RAM
- A E/S se dá entre o dispositivo e o buffer local
- A controladora informa a CPU que terminou sua operação, através de uma *interrupção*

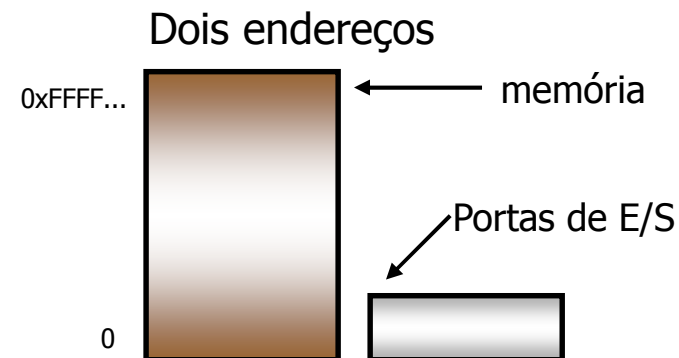


E/S Mapeada na Memória

- Alguns controladores possuem registradores de comunicação
 - Através desses registradores, o SO pode comandar o dispositivo
- Além dos registradores de controle, muitos dispositivos tem um buffer de dados em que o SO pode ler ou escrever
 - Ex: placa de vídeo

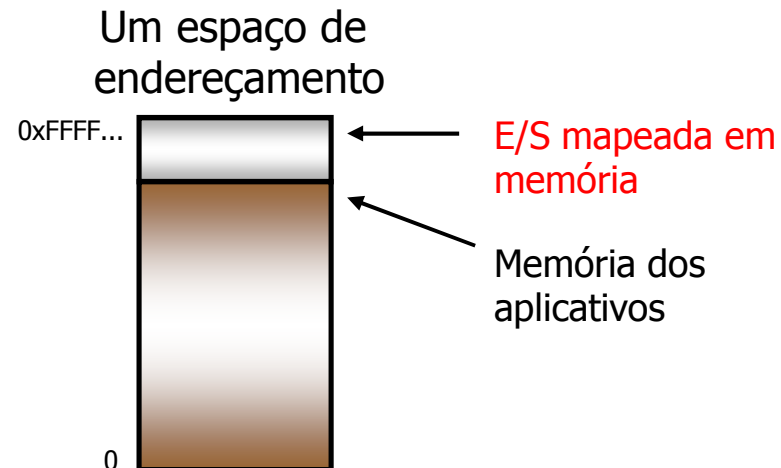
E/S Mapeada na Memória

- Há duas formas da CPU se comunicar com os registradores de controle e com o buffer da controladora
- E/S mapeada em espaço de E/S: No projeto do processador são especificados registradores e instruções específicas para o tratamento de E/S



E/S Mapeada na Memória

- E/S mapeada em memória: todos os registradores de controle são mapeados em endereços da memória
 - Cada registrador é associado a um endereço da memória, no qual nenhum outro dado pode ser armazenado





E/S Programada

- Método de E/S mais simples: toda a interação entre o processador e o controlador é de responsabilidade do programador. Princípio: envio de um comando ao controlador e aguardar a conclusão do mesmo.
- CPU espera até operação de E/S ser completada
 - Verificação contínua do registrador de saída OU
 - Consultas periódicas (polling) entremeadas com outras operações de cálculo.
- **Vantagem:** Implementação Simples
- **Desvantagem:** desperdício de processamento



E/S Orientada a Interrupções

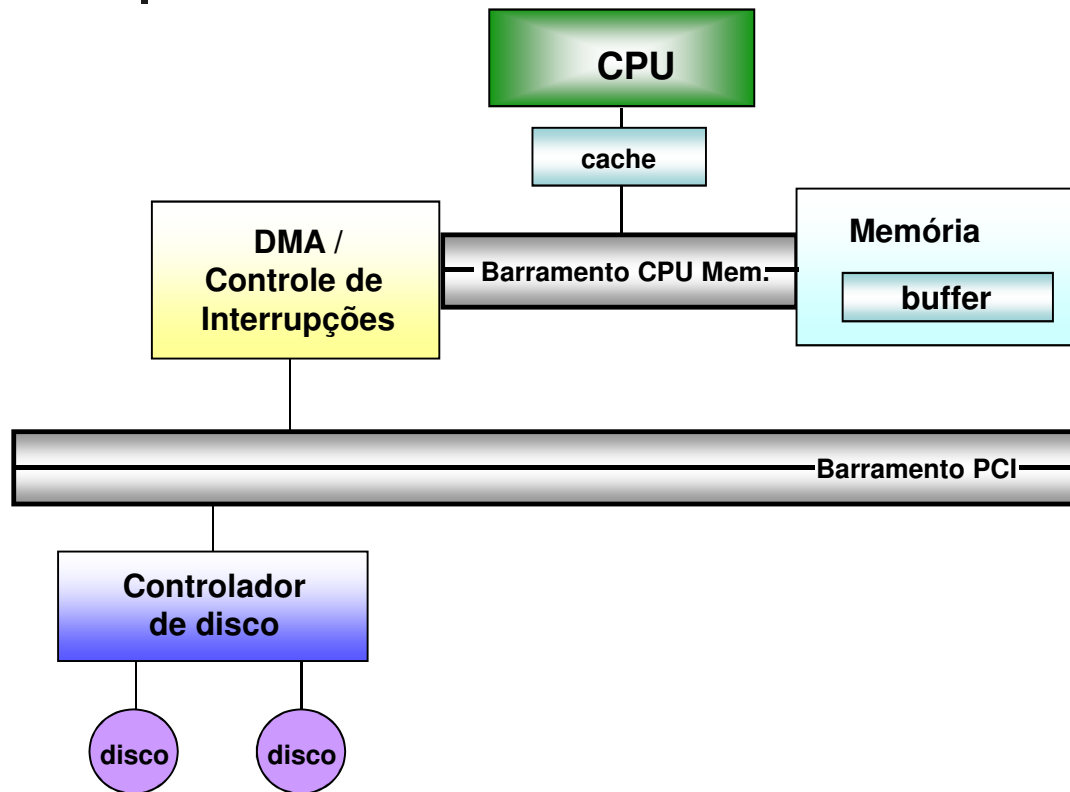
- Permite que a CPU execute outro processo enquanto o dispositivo de E/S realiza suas tarefas.
- O próprio dispositivo informa a CPU, via interrupção, quando a operação foi concluída
 - Não necessita checar periodicamente o dispositivo para verificar se a atividade já foi executada
- **Vantagem:** Contorna o problema de espera da CPU
- **Desvantagem:** pode gerar muitas interrupções. Interrupções consomem muito tempo.



Acesso Direto a Memória (DMA)

- Usado para permitir a dispositivos de E/S de alta velocidade transmitam informação em velocidade comparável à da memória.
- Princípio: CPU inicializa o controlador de DMA e fica liberada para fazer outras tarefas.
- Controladora do dispositivo transfere blocos de dados do buffer diretamente à memória principal, sem a intervenção da CPU.
 - Requer controladora de DMA
 - Requer controlador de interrupções para a CPU

Etapas de uma Transferência DMA



- CPU configura o controlador DMA:
 - Tipo de operação (leitura ou gravação)
 - Endereço do dispositivo
 - Endereço inicial do bloco de memória
 - Quantidade de dados a ser transferida
- Controlador DMA realiza as operações de transferência
- Controlador DMA envia um sinal de interrupção
- CPU recebe a interrupção e ativa a rotina de tratamento correspondente

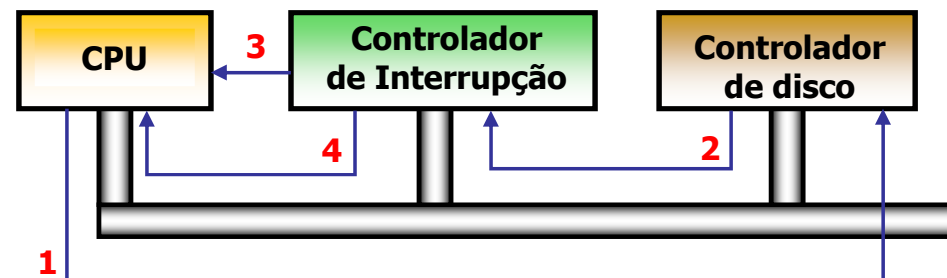


Interrupções

- Os SOs modernos são baseados em interrupções
- Interrupções transferem o controle para rotinas do SO de tratamento de interrupções
- Esta tarefa é gerenciada por um dispositivo de hardware denominado controlador de interrupções.
- O SO preserva o estado da CPU, armazenando os valores dos registradores e o contador de programa
- Outras interrupções ficam desabilitadas até que a interrupção seja processada

Passos de uma Interrupção

1. O *driver* informa ao controlador de E/S o que deve ser feito
2. A controladora do dispositivo de E/S informa ao chip controlador de interrupções o término da tarefa
 - Isto é feito através de linhas específicas do barramento
3. Se a controladora de interrupções estiver pronta (não ocupada com outra interrupção) avisa o término da tarefa de E/S à CPU
4. Colocando o número do dispositivo de E/S no barramento para que a CPU leia e saiba qual dispositivo de E/S terminou a tarefa





Interrupções

- **Vetor de interrupções:** Organiza múltiplas interrupções ao mesmo tempo
 - Contém os endereços dos procedimentos tratadores de interrupção para o dispositivo em questão
 - O número do dispositivo pode ser utilizado como índice
 - Baseadas em prioridade



O Sub-Sistema de E/S

- Complexo devido a diversidade de dispositivos de E/S
- Principal objetivo: padronizar as rotinas de acesso aos periféricos.
- Dividido em 4 camadas
 - Hardware (nível 1)
 - Drivers (nível 2)
 - E/S independente do dispositivo (nível 3)
 - E/S nível de usuário (nível 4)



Camadas do sistema de E/S

- Drivers: tem por objetivo abstrair detalhes do periférico para a camada superior, são aplicações que lidam com controladores, interrupções, DMA, etc
- E/S independente do dispositivo: implementa os principais procedimentos de gerência de E/S. Como: escalonamento, denominação, bufferização, cache de dados, alocação/liberação, permissões de acesso e tratamento de erros.
- E/S no nível de usuário: é a visão que o usuário (programador) tem dos dispositivos de E/S. Depende do fabricante/compilador utilizado. Ex: C++ (printf(),scanf())



Discos Rígidos – HD (hard disk)

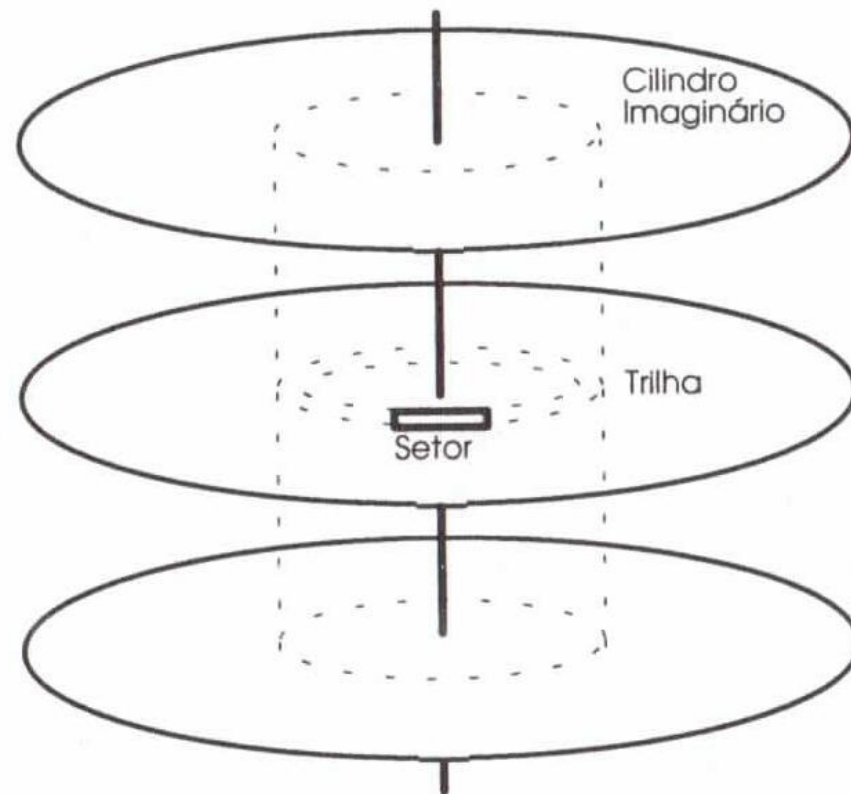
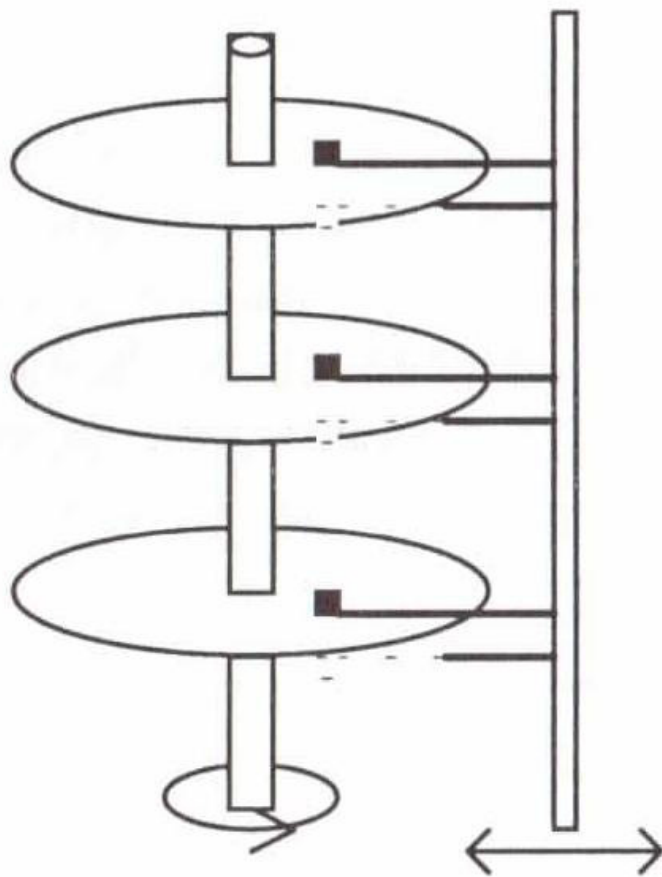
- Principal periférico para o SO
- discos metálicos empilhados (aço ou alumínio) + cabeçotes (1 para cada superfície de cada disco)
- As duas superfícies dos discos são revestidas com uma película magnética onde os dados são gravados
- Organização lógica
 - Trilhas: circunferências concêntricas
 - Setores: divisão das trilhas (normalmente blocos de 512bytes)
 - Cilindros: trilhas com a mesma distância do eixo



Discos Rígidos – HD (hard disk)

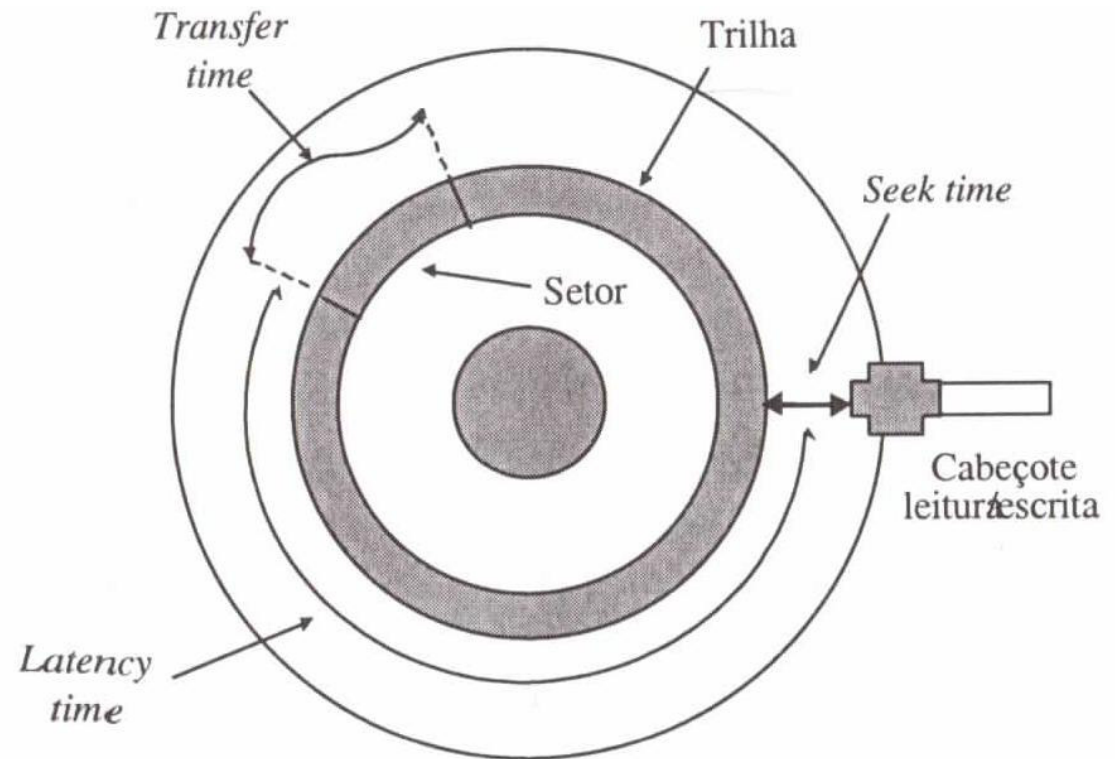
- Formatação física: definição de trilhas e setores (fabricante)
- Formatação lógica: gravar informações no disco de forma que os arquivos possam ser escritos, lidos e alterados pelo SO.
- Partição: dividir logicamente um disco em vários outros discos.

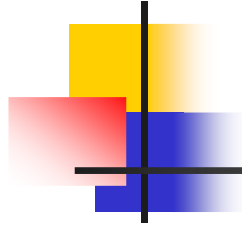
Discos Rígidos – HD (hard disk)



Discos Rígidos – HD (hard disk)

- Tempo de acesso = soma dos tempos:
 - Seek Time: deslocar cabeçote até a trilha
 - Latency Time: atraso rotacional
 - Transfer Time: transferência dos dados





Algoritmos de Escalonamento do HD

- Conforme o explicado, o tempo para ler ou escrever um bloco no disco é determinado por 3 fatores
 - Tempo de posicionamento
 - Tempo de rotação
 - Tempo de transferência real do dado
- O que pode mudar de um sub-sistema de E/S para outro é a política de posicionamento. A seguir algumas políticas/algoritmos.



Algoritmos de Escalonamento do HD

- **Algoritmo FIFO** (*First in, first out*) ou FCFS (*first-come, first-served*)
 - As requisições são atendidas na ordem em que elas chegam
 - Ex: se o braço está posicionado no cilindro 11 e chegam novas requisições para os cilindros 1, 36, 16, 34, 9, 12. O braço deverá percorrer 10, 35, 20, 18, 25, 3 cilindros respectivamente, totalizando 111 cilindros percorridos



Algoritmos de Escalonamento do HD

- **Algoritmo SSF** (*shortest seek first* – posicionamento mais curto primeiro)
- Manter uma tabela indexada pelo número do cilindro, com todas as requisições pendentes
 - A próxima requisição seria a mais próxima da posição atual
 - Ex.: Estando a cabeça de leitura no cilindro 11, as requisições para os cilindros 1, 36, 16, 34, 9, 12 seriam atendidas na seguinte ordem: 12, 9, 16, 1, 34, 36. Assim percorreria 1, 3, 7, 15, 33 e 2 cilindros respectivamente, totalizando 61 cilindros percorridos



Algoritmos de Escalonamento do HD

- Problema do algoritmo SSF
 - Se várias requisições de leitura chegam próxima ao cabeçote de leitura, as requisições mais distantes podem demorar um tempo excessivo para serem atendidas
- **Algoritmo do elevador**
 - Os dados são lidos na mesma direção até não haver mais requisições naquela direção, então o cabeçote muda de direção



Algoritmos de Escalonamento do HD

- **Elevador circular**

- Determina-se uma direção, por exemplo externa->interna, então os dados são lidos nesta direção até não haver mais requisições naquela direção, então o cabeçote volta à borda original (externa) e percorre a mesma direção.
- Desta forma evita-se atender trilhas que foram atendidas recentemente.



RAID (Redundant Array of Independent Disks)

- Estrutura que se propõe a solucionar problemas decorrentes do armazenamento de grande volume de dados.
- Grande volume = muitos setores , se ocorrer algum problema físico com 1 deles = perda de dados
- Backups nem sempre são a versão mais atual dos dados

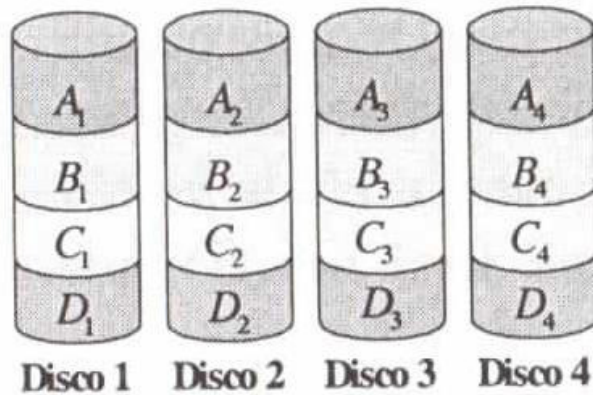


RAID (Redundant Array of Independent Disks)

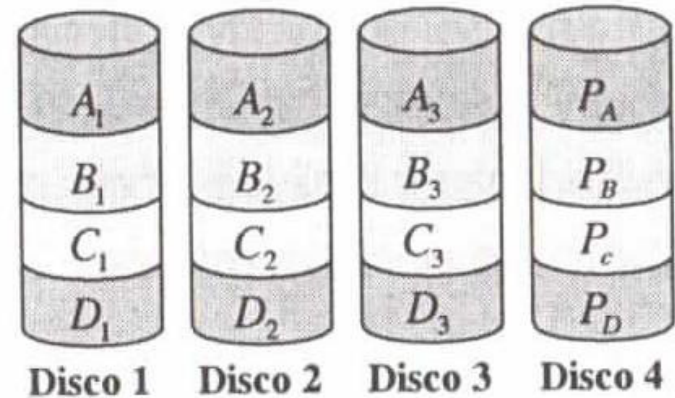
- RAID: combinar vários discos em uma estrutura lógica que aumente a confiabilidade. Armazenar os dados de forma redundante nos discos. Escrever e acessar os dados de forma paralela em diferentes discos.
 - RAID 0: dados divididos e escritos entre os discos (stripping)
 - RAID 1: mirroring – cópia fiel em disco secundário
 - RAID 2,3,4: stripping com informação de paridade para recuperação salva em 1 disco. (2 – bit, 3 – byte e 4 – bloco)
 - RAID 5: stripping com paridade dividida entre discos.

RAID (Redundant Array of Independent Disks)

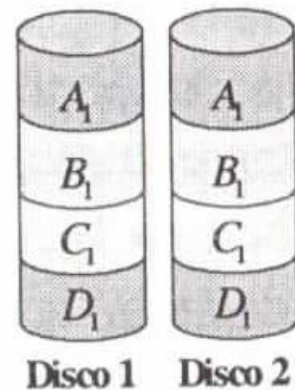
RAID 0



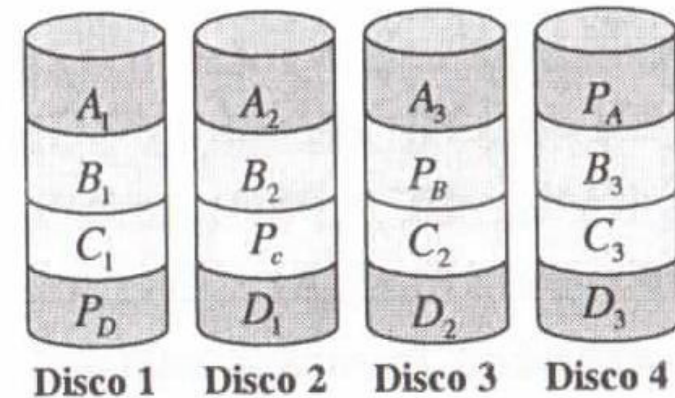
RAID 3

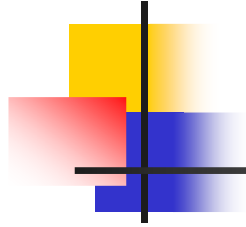


RAID 1



RAID 5





Exercício

- O braço está posicionado no cilindro 15 e chegam requisições para os cilindros 3, 40, 16, 29, 9, 12.
- Execute os algoritmos de escalonamento de HD: FIFO, SSF, Elevador (direção inicial crescente) e Elevador Circular (direção crescente).