



**Curso:** Tecnólogo - Superior em Sistemas Para Internet

**Disciplina:** Sistemas Operacionais

Nome: \_\_\_\_\_ Nota: \_\_\_\_\_

### Atividade 01

**Com relação ao mecanismo de memória virtual, utilize V para as afirmações VERDADEIRAS e F para as afirmações FALSAS.**

( ) Quando um programa acessa uma página mapeada no espaço de memória virtual, mas que não foi carregada na memória física do computador acontece uma interrupção (ou exceção) disparada pelo hardware: ocorre uma interrupção de falta de página (page-fault)

( ) Normalmente as memórias mais rápidas, como os registradores da CPU e os caches, têm menor capacidade de armazenamento, mais caras e consomem menos energia que memórias mais lentas, como a memória principal (RAM) e os discos.

( ) Se o processo solicita uma página, um dispositivo de hardware traduz endereços virtuais em endereços físicos, é geralmente implementada como parte da Unidade Central de Processamento ou CPU (Central Processing Unit) e é conhecido como MMU.

( ) São algoritmos de substituição de páginas: Pior Encaixe (Worst-fit), Buddy, Encaixe Anterior (Under-fit).

### Atividade 02

**Com relação a endereçamento Lógico e Físico, utilize V para as afirmações VERDADEIRAS e F para as afirmações FALSAS.**

( ) Endereços lógicos são aqueles utilizados nos processos.

( ) Os endereços físicos são utilizados no barramento do sistema.

( ) A conversão é feita de endereços físicos para endereços lógicos.

( ) A conversão entre endereços lógicos e físicos é feito por software específico conhecido como MMU.

### Atividade 03

**Explique as diferenças entre as três implementações de Memória Virtual vistos em aula (citar 0,3 pontos).**

a) \_\_\_\_\_

b) \_\_\_\_\_

c) \_\_\_\_\_



#### Atividade 04

Considerando a tabela de segmentos a seguir (com valores em decimal), calcule os endereços físicos (reais) correspondentes aos endereços lógicos **0:55, 1:50, 2:190, 3:900, 4:300, 5:790 e 6:99**.

Segmento	0	1	2	3	4	5	6
Base	200	44	0	2000	2500	900	1900
Limite	200	810	43	1000	500	1000	100

Endereço Real:

0: \_\_\_\_\_ 1: \_\_\_\_\_ 2: \_\_\_\_\_ 3: \_\_\_\_\_ 4: \_\_\_\_\_ 5: \_\_\_\_\_ 6: \_\_\_\_\_

#### Atividade 05

Considerando os algoritmos de leitura de disco **FCFS** (*First Come, First Served*) que consiste em atender as requisições na ordem da fila, ou seja, na ordem em foram pedidas pelos processos; **SSTF** (*Shortest Seek Time First – Menor Tempo de Busca Primeiro*): que consiste em sempre atender o pedido que está mais próximo da posição atual da cabeça de leitura; **SCAN**: cuja finalidade é “varrer” (*scan*) continuamente o disco, do início ao final, atendendo os pedidos que encontra pela frente; **ao atingir o final do disco**, ela inverte seu sentido de movimento e volta, atendendo os próximos pedidos.

Considere um disco hipotético com 1.000 blocos (enumerados de 0 ao 999), cuja cabeça de leitura se encontra inicialmente sobre o bloco 500.

A fila de pedidos de acesso pendentes contém pedidos de acesso aos seguintes blocos do disco, em sequência:

**278, 999, 447, 71, 161, 659, 335**

e utilizando o algoritmo SCAN, para atender os pedidos de leitura na ordem indicada,

a) Qual é a ordem de leitura utilizando o algoritmo SCAN? (0,5 pontos)

b) A cabeça de leitura teve de deslocar-se por quantos blocos? (0,5 pontos)

#### Atividade 06

Considerando os algoritmos de leitura de disco **FCFS** (*First Come, First Served*) que consiste em atender as requisições na ordem da fila, ou seja, na ordem em foram pedidas pelos processos; **SSTF** (*Shortest Seek Time First – Menor Tempo de Busca Primeiro*): que consiste em sempre atender o pedido que está mais próximo da posição atual da cabeça de leitura; **SCAN**: cuja finalidade é “varrer” (*scan*) continuamente o disco, do início ao final, atendendo os pedidos que encontra pela frente; **ao atingir o final do disco**, ela inverte seu sentido de movimento e volta, atendendo os próximos pedidos.

Considere um disco hipotético com 1.000 blocos (enumerados de 0 ao 999), cuja cabeça de leitura se encontra inicialmente sobre o bloco 500.

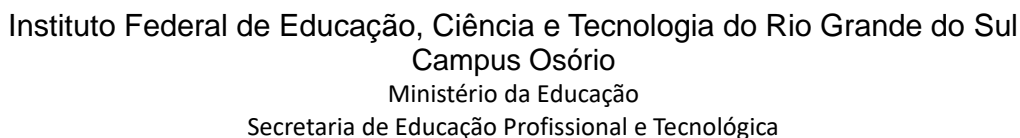
A fila de pedidos de acesso pendentes contém pedidos de acesso aos seguintes blocos do disco, em sequência:

**278, 999, 447, 71, 161, 659, 335**

e utilizando o algoritmo FCFS, para atender os pedidos de leitura na ordem indicada,

a) Qual é a ordem de leitura utilizando o algoritmo SCAN? (0,5 pontos)

b) A cabeça de leitura teve de deslocar-se por quantos blocos? (0,5 pontos)



Considerando os algoritmos de leitura de disco **FCFS** (*First Come, First Served*) que consiste em atender as requisições na ordem da fila, ou seja, na ordem em foram pedidas pelos processos; **STTF** (*Shortest Seek Time First – Menor Tempo de Busca Primeiro*): que consiste em sempre atender o pedido que está mais próximo da posição atual da cabeça de leitura; **SCAN**: cuja finalidade é “varrer” (*scan*) continuamente o disco, do início ao final, atendendo os pedidos que encontra pela frente; **ao atingir o final do disco**, ela inverte seu sentido de movimento e volta, atendendo os próximos pedidos.

e utilizando o algoritmo **SSTF**, para atender os pedidos de leitura na ordem indicada,

- a) Qual é a ordem de leitura utilizando o algoritmo SCAN? (0,5 pontos)
- b) A cabeça de leitura teve de deslocar-se por quantos blocos? (0,5 pontos)

Considere um alocador de memória do tipo **Worst-fit**. Dada uma área contínua de memória RAM com 1 GByte (1.024 MBytes) e considere que o alocador sempre usa “*blocos*” de 50MBytes.

Apresente a evolução da situação da memória para a sequência de alocações e liberações de memória indicadas a seguir (*Identificar a alocação: A1, A2, A3, A4, A5 e A6*).

- (a) Aloca A1 300 MB
- (b) Aloca A2 300 MB
- (c) Aloca A3 250 MB
- (d) Libera A2
- (e) Libera A1
- (f) Aloca A4 100 MB
- (g) Aloca A5 40 MB
- (h) Aloca A6 150 MB

[illegible]



### Atividade 09

Complete os espaços:

- a) \_\_\_\_\_: Esta abordagem armazena informações de paridade para tolerar falhas em blocos ou discos. Todavia, essas informações não ficam concentradas em um único disco físico, sendo distribuídas uniformemente entre eles.
- b) \_\_\_\_\_: Neste nível os discos físicos são divididos em áreas de tamanhos fixo chamadas *fatias* ou *faixas*. O maior espalhamento dos blocos sobre os discos físicos contribui para distribuir melhor a carga de acessos entre eles e assim ter um melhor desempenho.
- c) \_\_\_\_\_: Neste nível os dados são “fatiados” em bits individuais que são escritos nos discos físicos em sequência; discos adicionais são usados para armazenar códigos corretores de erros (*Hamming Codes*).
- d) \_\_\_\_\_: Neste nível, todo o conteúdo é replicado em dois ou mais discos.
- a) RAID 0 (striping); RAID 5; RAID 2; RAID 1 (Espelhamento)  
b) RAID 2; RAID 0 (striping); RAID 5; RAID 1 (Espelhamento)  
c) RAID 1; RAID 0 (striping); RAID 5; RAID 0 (Espelhamento)  
d) RAID 5; RAID 0 (striping); RAID 2; RAID 1 (Espelhamento)  
e) N.D.A.

### Atividade 10

Complete os espaços, indicando o nome dos diretórios de um sistema Linux:

- \_\_\_\_\_ : Programas binários básicos;
- \_\_\_\_\_ : Núcleo Linux e outros arquivos necessários para o processo de inicialização;
- \_\_\_\_\_ : Arquivos de dispositivo;
- \_\_\_\_\_ : Arquivos de configuração;
- \_\_\_\_\_ : Arquivos pessoais dos usuários;
- \_\_\_\_\_ : Aplicações extras fornecidas por terceiros;
- \_\_\_\_\_ : Arquivos pessoais do superusuário;
- \_\_\_\_\_ : Programas do sistema do superusuário;
- \_\_\_\_\_ : Arquivos temporários, este diretório é comumente limpo na inicialização;
- \_\_\_\_\_ : São específicos do núcleo Linux (processos, pipes, etc).

### Atividade 10

O que é DMA (significado 0,1 ponto) e como impacta no desempenho de um sistema computacional?