

Lista de Exercícios - Gerência de Memória

Henrique Oliveira da Cunha Franco

Questão 1. Explique o conceito de alocação contígua de memória.

A alocação contígua de memória é uma técnica em que cada processo é carregado em um único bloco contínuo de memória física. Isso facilita o acesso direto e simples aos dados, porém causa problemas de fragmentação externa e limita a flexibilidade no gerenciamento da memória.

Questão 2. Qual a função da MMU na gerência de memória?

A Unidade de Gerência de Memória (MMU) é responsável por traduzir endereços virtuais, gerados pelos programas, em endereços físicos, utilizados pela memória principal. Ela também pode realizar funções como proteção de memória e controle de acesso.

Questão 3. Qual a diferença entre endereço físico e virtual?

O endereço virtual é o endereço utilizado pelos programas em execução, enquanto o endereço físico é aquele onde os dados realmente estão localizados na memória. A MMU é responsável por mapear os endereços virtuais em físicos.

Questão 4. Se todos os processos que precisam executar não cabem na memória, o que pode ser feito? Explique a ideia de swapping.

Quando não há memória suficiente para todos os processos, o sistema pode utilizar a técnica de *swapping*, que consiste em transferir temporariamente processos da memória principal para o disco, liberando espaço para que outros possam ser executados. Quando necessário, o processo é trazido de volta à memória.

Questão 5. Explique o problema da fragmentação externa. Como solucionar o problema?

A fragmentação externa ocorre quando há espaços livres na memória, mas que estão distribuídos em blocos pequenos e não contíguos, impedindo que um processo maior seja alocado. Uma possível solução é a compactação da memória, que reorganiza os blocos para formar um espaço contínuo.

Questão 6. O que acontece quando um processo excede o tamanho alocado da sua área de crescimento?

Quando um processo excede sua área de crescimento (como a pilha ou o heap), pode ocorrer uma falha de segmentação (segmentation fault), levando à interrupção do processo,

a menos que o sistema permita expansão dinâmica de memória.

Questão 7. O sistema operacional mantém uma lista de espaços livres na memória física. Sempre que um novo processo é criado esta lista é percorrida e usada. Quais as formas de percorrer a lista, ou seja, de alocar um bloco na memória para o processo?

As formas de alocação são:

- **First-fit:** aloca o primeiro bloco de memória que for suficientemente grande.
- **Best-fit:** aloca o menor bloco que seja suficientemente grande, minimizando o espaço desperdiçado.
- **Worst-fit:** aloca o maior bloco disponível, esperando manter blocos menores disponíveis para futuras alocações.

Questão 8. Considere os blocos de memória: 10K, 4K, 20K, 18K, 7K, 9K, 12K, 13K. Processos a serem alocados: 5K, 10K, 15K, 8K, 3K, 7K, 6K. Abaixo estão as simulações dos algoritmos de alocação:

a. First-Fit - Alocação Primeiro Encaixe

Neste algoritmo, o sistema percorre a lista de blocos e aloca o primeiro que for suficientemente grande. A lista de blocos livres é: [10K, 4K, 20K, 18K, 7K, 9K, 12K, 13K].

- P1 (5K): Aloca no primeiro bloco suficiente → **10K** → sobra 5K
- P2 (10K): Bloco **4K** não serve, o próximo é **20K** → sobra 10K
- P3 (15K): Pula os anteriores, entra em **18K** → sobra 3K
- P4 (8K): Pula até encontrar **9K** → sobra 1K
- P5 (3K): Aloca em **4K** → sobra 1K
- P6 (7K): Entra exatamente em **7K** → sobra 0K
- P7 (6K): Próximo disponível suficiente é **12K** → sobra 6K

Blocos restantes: [5K, 1K, 10K, 3K, 1K, 6K, 13K]

b. Best-Fit - Melhor Encaixe

Neste algoritmo, a alocação ocorre no **menor bloco** disponível que acomode o processo. A cada passo, reordenamos os blocos por tamanho para escolher o melhor encaixe.

- P1 (5K): Melhor encaixe = **7K** → sobra 2K

- P2 (10K): Encaixa exatamente no bloco **10K** → sobra 0K
- P3 (15K): Melhor entre os grandes é **18K** → sobra 3K
- P4 (8K): Melhor encaixe é **9K** → sobra 1K
- P5 (3K): Melhor bloco = **4K** → sobra 1K
- P6 (7K): Melhor bloco = **12K** → sobra 5K
- P7 (6K): Melhor bloco = **13K** → sobra 7K

Blocos restantes: [2K, 0K, 3K, 1K, 1K, 5K, 7K]

c. Worst-Fit - Pior Encaixe

Neste caso, o processo é alocado no **maior bloco** disponível. Sempre escolhemos o bloco que deixará o maior espaço remanescente.

- P1 (5K): Maior bloco = **20K** → sobra 15K
- P2 (10K): Maior agora = **18K** → sobra 8K
- P3 (15K): Maior agora = **15K** → sobra 0K
- P4 (8K): Maior agora = **13K** → sobra 5K
- P5 (3K): Maior agora = **12K** → sobra 9K
- P6 (7K): Maior agora = **10K** → sobra 3K
- P7 (6K): Maior agora = **9K** → sobra 3K

Blocos restantes: [15K, 8K, 0K, 5K, 9K, 3K, 3K]