

L16_alocação de memória

Total de pontos 34/34 ?

Endereço de e-mail *

bruno.oliveira.duarte.bd@gmail.com

✓ 16.1 Explique o que é fragmentação externa. Quais formas de alocação de memória estão livres desse problema? Assinale as corretas *

3/3

- ☒ A estratégia worstfit é a melhor alternativa diminuir fragmentação externa ✓
- ☒ Fragmentação externa é quando, após sucessivas inserções e remoções, sobram espaços pequenos na memória onde, normalmente, novas requisições de alocação não podem ser atendidas devido ao diminuto espaço contíguo ✓
- ☐ Fragmentação externa é quando um processo não ocupa toda a página destinada a ele. Por exemplo, um processo de 900kb alocado em um espaço de 1024kb
- ☐ A estratégia bestfit não apresenta fragmentação externa
- ☐ Fragmentação externa é quando um processo está quebrado em seções, do tipo TEXT, DATA, etc
- ☐ A estratégia firstfit possui melhor desempenho pois evita fragmentação externa



✓ 16.2 Explique o que é fragmentação interna. Quais formas de alocação de memória estão livres desse problema? Assinale a correta *

- ☒ Uma forma de se minimizar fragmentação interna é utilizar memória segmentada, com tamanho variáveis ✓
- ☐ Usando memória paginada não haverá fragmentação interna
- ☐ Fragmentação interna é quando os dados do processo não cabem em apenas uma página
- ☐ Todas as formas de alocação de memória (worstfit, bestfit, etc) estão livres de fragmentação interna
- ☐ Outro:

✓ 16.3 Em que consistem as estratégias de alocação first-fit, best-fit, worst-fit e next-fit? Assinale a correta *

4/4

- ☐ First-fit é quando se escolhe a primeira página da memória para alocação do processo
- ☐ Best-fit é quando se escolhe a melhor página da memória. Melhor aqui se refere a página mais rápida
- ☐ Worst-fit é quando se escolhe a maior área livre. Como próprio nome diz não há nenhuma vantagem nessa estratégia
- ☐ Next-fit é quando a próxima página é usada, independente de estar livre ou não
- ☒ NDA ✓



16.4 Considere um sistema com processos alocados de forma contígua na memória. Em um dado instante, a memória RAM possui os seguintes “buracos”, em sequência e isolados entre si: 5K, 4K, 20K, 18K, 7K, 9K, 12K e 15K. Indique a situação final de cada buraco de memória após a seguinte sequência de alocações: 12K->10K->5K->8K->10K. Considere as estratégias de alocação first-fit, best-fit, worst-fit e next-fit. Associe as linhas as colunas, onde A, B, C... são nomes para os buracos *

	First-fit	Best-fit	Worst-fit	Next-fit	NDA	Pontuação	
A5 B4 C8 D8 E7 F9 G4 H0	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	2/2	✓
A5 B0 C8 D8 E7 F9 G4 H0	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	2/2	✓
A5 B4 C8 D8 E2 F1 G2 H15	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	2/2	✓
A5 B4 C4 D4 E7 F9 G4 H0	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	2/2	✓
A0 B4 C20 D8 E7 F1 G0 H5	<input type="radio"/>	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	2/2	✓
A0 B4 C0 D8 E7 F9 G2 H15	<input checked="" type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	2/2	✓
A0 B0 C20 D8 E7 F1 G0 H15	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	2/2	✓
A5 B4 C8 D8 E2 F1 G2 H5	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input type="radio"/>	<input checked="" type="radio"/>	2/2	✓



✓ 16.5 Considere um banco de memória com os “buracos” não-contíguos da questão 5 do livro. Nesse banco de memória devem ser alocadas áreas de 5MB, 10MB e 2MB, nesta ordem, usando os algoritmos de alocação First-fit, Best-fit ou Worst-fit. Indique a alternativa correta: *

3/3

- ☐ Se usarmos Best-fit, o tamanho final do buraco B4 será de 6 Mbytes
- ☒ Se usarmos Worst-fit, o tamanho final do buraco B4 será de 15 Mbytes
- ☐ Se usarmos First-fit, o tamanho final do buraco B4 será de 24 Mbytes
- ☐ Se usarmos Best-fit, o tamanho final do buraco B5 será de 7 Mbytes
- ☐ Se usarmos Worst-fit, o tamanho final do buraco B4 será de 9 Mbytes



✓ 16.6 Considere um alocador de memória do tipo Buddy binário. Dada uma área contínua de memória RAM com 1GByte (1.024 MBytes), apresente a evolução da situação da memória para a sequência de alocações e liberações de memória indicadas a seguir. (a) Aloca A1 200 MB, (b) Aloca A2 100 MB, (c) Aloca A3 150, (d) Libera A2, (e) Libera A1, (f) Aloca A4 100 MB, (g) Aloca A5 40 MB, (h) Aloca A6 300 MB. Marque a opção que reflete a situação final da memória *

5/5

- ☐ A6, A3, livre64 A4, A5, Livre64
- ☐ A6, livre256, A3, A4, A5, Livre64
- ☐ A3, A4, A5, Livre128, A6
- ☐ A6, livre 128, A3, A4, A5, Livre64
- ☐ A1, 256, 512
- ☐ A6, A3, A4, A5, Livre128
- ☐ A1, A2, A3, A5, A4, A6, livre128
- ☒ A6, A3, A4, A5, Livre64



Crie um breve resumo do capítulo com suas próprias palavras. Procure destacar os principais conceitos aprendidos. Mínimo de 100 e máximo de 200 palavras, o que equivale entre 10 a 20 linhas aproximadamente. *

Para que um sistema computacional possa ser executado, precisa de memória. Alocar memória é reservar áreas da memória RAM para um determinado fim. O alocador de memória é responsável por alocar e liberar a memória. Um problema de alocação refere-se ao tamanho de memória a ser alocado. Muitas vezes, liberar memória cria áreas livres que não podem ser usadas devido ao seu tamanho, num processo chamado de fragmentação externa.

Para minimizar esse problema, existem mecanismos de análise que determinam as melhores posições para alocar memória: first-fit, best-fit, worst-fit e next-fit. Outra maneira de resolver o problema é reorganizar áreas de memória física ocupadas para outros locais, num processo chamado de desfragmentação. Ainda, é possível arredondar os pedidos de alocação minimizando pequenas fragmentações externas.

O alocador Buddy é uma estratégia que consiste em alocar um espaço em que caiba a memória requisitada, particionando espaços maiores em pares. Ao liberar, busca liberar os pares correspondentes.

O alocador Slab é especializado em alocações e liberações para o núcleo do SO, e baseia-se no caching de objetos.

Aplicações de usuário também podem precisar alocar memória. Para isso, normalmente utilizam-se de bibliotecas de sistemas, como a LibC. Normalmente seguem as estratégias best-fit ou memory pool.

Este conteúdo não foi criado nem aprovado pelo Google. - [Termos de Serviço](#) - [Política de Privacidade](#)

Google Formulários

