

COM120 – SISTEMAS OPERACIONAIS – EP08

MEMÓRIA VIRTUAL, GERENCIAMENTO DE MEMÓRIA E PAGINAÇÃO

Matheus Martins Batista¹
Carlos Minoru Tamaki²

1 -

Na gestão da memória comum dos computadores, a informação é carregada em pedaços de tamanho específico. Cada pedaço é referido como uma página. A memória principal pode conter apenas um número limitado de páginas de cada vez. Requer um quadro para cada página que pode carregar. Uma falha de página ocorre quando uma página não é encontrada, e pode precisar de ser carregada do disco para a memória.

No armazenamento de computadores, a anomalia de Bélády é o fenómeno em que o aumento do número de molduras de página pode resultar num aumento do número de falhas de página para certos padrões de acesso à memória. Este fenómeno é geralmente experimentado quando se utiliza o algoritmo de substituição de páginas first-in first-out (FIFO). No FIFO, a falha de página pode ou não aumentar à medida que os frames de página aumentam, mas nos algoritmos Optimal (ideal) e baseados em pilha como o LRU, à medida que os frames de página aumentam, a falha de página diminui. László Bélády demonstrou isto em 1969.

Quando ocorre uma falha de página e todas os frames estão em uso, é necessário limpar uma para criar espaço para a nova página. Um algoritmo simples é FIFO: qualquer página que esteja nas molduras há mais tempo é a que é apagada. Até ser demonstrada a anomalia de Bélády, acreditava-se que um aumento no número de molduras de página resultaria sempre no mesmo número de falhas de página ou em menos páginas.

Para ilustrar, a tabela abaixo retrata um caso em que isso acontece usando FIFO, quando subimos de 3 molduras para 4.

		0	1	2	3	0	1	4	0	1	2	3	4	
Página mais nova		0	1	2	3	0	1	4	4	4	2	3	3	
Página mais velha			0	1	2	3	0	1	1	1	4	2	2	
		P	P	P	P	P	P	P			P	P		9 Page Faults
		0	1	2	3	0	1	4	0	1	2	3	4	
Página mais nova		0	1	2	3	3	3	4	0	1	2	3	4	
Página mais velha			0	1	2	2	2	3	4	0	1	2	3	
				0	1	1	1	2	3	4	0	1	2	
		P	P	P	P			P	P	P	P	P	P	10 Page Faults

¹ Graduando em Ciências da Computação pela Universidade Federal de Itajubá – E-mail: matmb@unifei.edu.br

² Professor orientador. Mestre em Ciência e Tecnologia da Computação. Docente na Universidade Federal de Itajubá – E-mail: minoru@unifei.edu.br

2 -

- a) Em um erro de página, o estado do thread é configurado para bloqueado (1) como uma operação de E/S que é necessário para levar a nova página à memória.
- b) A thread continuará a ser executada (2) se o endereço for resolvido na tabela de páginas.

3 -

Para um sistema utilizando **FIFO** (*First-in, First-out*): deve-se encontrar o primeiro segmento grande o suficiente para receber o segmento que entra. Caso a relocação não for possível e nenhum segmento for grande o suficiente, é necessário selecionar uma combinação de segmentos cujas memórias sejam contíguas, que sejam “os mais próximos ao primeiro da lista” e que possam acomodar o novo segmento. Se a relocação for possível, reorganize a memória de modo que os primeiros N segmentos suficientemente grandes para o segmento que entra fiquem contíguos na memória. Adicione qualquer espaço restante à lista de espaços livres em ambos os casos.

Para um sistema utilizando **LRU** (*Least Recently Used*): selecione o segmento que não tenha sido utilizado pelo período de tempo mais longo e que seja grande o suficiente, adicionando qualquer espaço restante à lista de espaços livres. Caso nenhum segmento for suficientemente grande, selecione uma combinação dos segmentos “mais antigos” que estejam contíguos na memória (se a relocação não estiver disponível) e que sejam suficientemente grandes. Se a relocação estiver disponível, reorganize os N segmentos mais antigos para que fiquem contíguos na memória e substitua-os pelo novo segmento.

REFERÊNCIAS

Todas as questões:

SILBERSCHATZ, Abraham; GALVIN, Peter Baer; GAGNE, Greg. **Fundamentos de sistemas operacionais**. 9. ed. [S. l.]: Livros Técnicos e Científicos Editora, 2015. Cap. 8-9, p. 398-521, 1012 p. ISBN 978-1-1180-6333-0.

MACHADO, Francis Berenger; MAIA, Luiz Paulo. **Arquitetura de Sistemas Operacionais**. 5. ed. [S. l.]: Livros Técnicos e Científicos Editora, 2013. 266 p. ISBN 978-8-5216-2210-9.

Questão 1 e 3:

CRUZ, Pedro. **Gerenciamento de memória**. [S. l.], 28 ago. 2017. Disponível em: https://www.gta.ufrj.br/~cruz/courses/eel770/slides/9_memoria.pdf. Acesso em: 27 out. 2021.

KRUEGEL, Christopher. *Operating Systems: Virtual Memory and Paging*. UC Santa Barbara, 15 maio 2012. Disponível em: <https://web.archive.org/web/20160810021642/https://www.cs.ucsb.edu/~chris/teaching/cs170/doc/cs170-08.pdf>. Acesso em: 5 nov. 2021.

GOMES, Roberta Lima. **Gerência de Memória: Paginação**. [S. l.], 21 ago. 2013. Disponível em: [http://www.inf.ufes.br/~zegonc/material/Sistemas_Operacionais/Gerencia%20de%20Memoria%20\(2\)%20-%20Paginacao-BETA.pdf](http://www.inf.ufes.br/~zegonc/material/Sistemas_Operacionais/Gerencia%20de%20Memoria%20(2)%20-%20Paginacao-BETA.pdf). Acesso em: 5 nov. 2021.

GOMES, Roberta Lima. **Gerência de Memória: Aspectos de Projeto**. [S. l.], 21 jun. 2010. Disponível em: [http://www.inf.ufes.br/~zegonc/material/Sistemas_Operacionais/Gerencia%20de%20Memoria%20\(2\)%20-%20Paginacao-BETA.pdf](http://www.inf.ufes.br/~zegonc/material/Sistemas_Operacionais/Gerencia%20de%20Memoria%20(2)%20-%20Paginacao-BETA.pdf). Acesso em: 5 nov. 2021.

Questão 2:

CUNHA, Roberto Bernandinho; PREUSS, Evandro; MACEDO, Ricardo Tombesi. **Sistemas Operacionais**. 1. ed. [S. l.]: Núcleo de Tecnologia e Informação – NTE. Licenciatura Universidade Federal de Santa Maria, 2017. Cap. 3, p. 67-69, 147 p. ISBN 978-85-8341-218-2