

## P3 - Sistemas Operacionais

Matheus S. Redecker

<sup>1</sup>Pontifícia Universidade Católica do Rio Grande do Sul - PUCRS

1.

Endereços lógicos iguais podem ter diferentes endereços físicos, endereço físico são endereços em memória real, e endereços lógicos são endereços 'criados' para a execução de processos.

2.

Fragmentação externa é quando processos vão deixando de existir e com isso vão deixando espaços cada vez menores e assim não deixando novos programas entrarem, mesmo tendo espaço total mas não tem espaço contínuo para alocar, já fragmentação interna é a perda de memória em um tamanho fixo, processos pequenos e espaços de memória reservados muito grande com isso o processo não abrange toda a área reservada e essa área não é usada.

3.

$1024 * 2 = 2k$  por página então  $2 * 8$  tamanho da tabela = 16K que precisa de 14bits para endereço lógicos.

$2048 * 32 = 64K$  que é o tamanho total da tabela então precisa de 16 bits para endereçar a memória física.

4.

O sistema consulta na tabela de páginas o endereço correspondente na memória física, assim sabemos qual é a moldura para aquele endereço virtual (SOFTWARE), se não encontramos ali o endereço temos um page fold e temos que carregar a página para a memória física (HARDWARE).

5.

O processo "a" terá mais page folds e não irá usar o que buscar da memória, já o processo "b" será beneficiado pelo page fold, cada um que tiver ele carregará 199 elementos que serão acessados no futuro, o que não acontece no processo "a" pois ele carregará eventualmente algum endereço que será realmente usado.

6.

a) Como temos páginas de 4K, então do endereço 0 4096 é a primeira página, ou seja, página virtual zero com isso o endereço 4096 será traduzido para a moldura 2 de acordo com a tabela, e o endereço 8292 estará na página virtual 2 que vai de 8193 16384 e será traduzido para a moldura 6.

b) Na tabela invertida cada entrada é representada por uma hash composta por a página virtual e a moldura, então temos:

**Table 1. Tabela Invertida**

-	pid	endvirtual
0	0	3
1	0	1
2	0	0
3	0	5
4	0	4
5	0	9
6	0	2
7	0	11

Com isso os endereços serão endereçados em:

1- 4094: será preciso do par (0,0) então moldura 2

2- 8292: será preciso do par (0,2) então moldura 6

c)

**Table 2. Primeiro nivel**

-	pt1
0	pt2
1	-
2	-
3	-

**Table 3. Segundo nivel**

-	pt2
0	1
1	0
2	-
3	-

7.

a,b) Veja a tabela abaixo:

**Table 4. Tabela Convencional**

pagina	bitvalidade	moldura	cont LRU
0	0	-	-
1	1	1	1
2	1	3	0
3	1	2	0
4	0	-	-
5	0	-	-
6	0	-	-
7	0	-	-
8	0	-	-
9	0	-	-
10	0	-	-
11	0	-	-
12	0	-	-
13	0	-	-
14	0	-	-
15	1	0	0

8.

Mapa de bits: O sistema operacional mantém um bit para cada bloco da memória para indicar se o mesmo está livre ou ocupado, as vantagens é a simplicidade de implementação já as desvantagem seria quanto mais um processo necessita de unidades de alocação o gerenciador de memória necessita encontrar uma sequência equivalente de bits 0 e isso é um processo lento.

Lista encadeadas: É mantido uma lista encadeada de segmentos de memórias que podem conter os espaços de memória livres e também alocadas, as vantagens é que fica fácil de encontrar espaços vagos para novas entradas e colocar essa nova entrada é muito simples, mas como desvantagens é a fragmentação externa, pois dependendo vamos ter 'buracos' tão pequenos que não caberão nenhum processo.

9.

a) Verdadeiro, pois quanto mais molduras maior será a quantidade de páginas que conseguiremos endereçar guardar.

b) Falso, o processo de segmentação pura causa apenas fragmentação externa.

c) Verdadeiro.

d) Verdadeiro, dependendo da aplicação, como por exemplo acesso de matrizes.

e) Falso, a MMU força uma interrupção e o Sistema Operacional faz o tratamento do page fold.