

4ª. Lista de Exercícios (cedida pela Profa. Roberta Lima Gomes)
Período: 2008/1

Prof^a. Patrícia Dockhorn Costa
email: pdcosta@inf.ufes.br

Data de Entrega: não há. O objetivo da lista é ajudar no estudo individual dos alunos. Soluções de questões específicas poderão ser discutidas em sala de aula, conforme interesse dos alunos.

- 1) Utilizando as primitivas de *semáforos*, explique como implementar um *monitor*.
- 2) Considere a seguinte modelagem, por monitor, para o problema do produtor-consumidor com n produtores e m consumidores.

```
monitor buffer {
  int itens=0; cond temItens, temEspacos;
  ...
  int pega() {
    while (1) {
      if (!itens) wait(temItens);
      pega item no buffer
      itens--;
      signal(temEspacos);
      return (item);
    }
  }

  void coloca() {
    while (1) {
      if (itens==MAX) wait(temEspacos);
      coloca item no buffer
      itens++;
      signal(temItens);
    }
  }
}
```

Suponha que esse monitor funciona com a disciplina “sinaliza e continua”, dada em sala (mas observe que ele admite filas separadas por variáveis de condição). Explique por que essa solução não funciona corretamente.

- 3) Cite duas diferenças entre endereços lógicos e físicos.
- 4) Explique a diferença entre fragmentação interna e externa.

- 5) Considere um espaço de endereçamento lógico de oito páginas de 1024 palavras cada, mapeado em uma memória física de 32 quadros. Quantos bits existem no endereço lógico? E no endereço físico?
- 6) Determinado computador fornece a seus usuários um espaço de memória virtual de 232 Kbytes. O computador tem 218 Kbytes de memória física. A memória virtual é implementada por paginação e o tamanho de página é 4096 bytes. Um processo de usuário gera o endereço virtual 123456. Explique como o sistema estabelece a posição física correspondente. Faça a distinção entre as operações de software e hardware.

- 7) Considere o vetor bidimensional A:

```
int A[][] = new int[100][100];
```

onde A[0][0] está na posição 200, em um sistema paginado com páginas de tamanho 200 palavras. Observe que o vetor é armazenado em linha. Ou seja, o vetor é armazenado como A[0][0], A[0][1] ,..., A[0][100], A[1][100], ..., A[100][100]. No sistema paginado em questão, cada página pode armazenar até 200 elementos de A. Um processo pequeno esta na página 0 (posições de 0 a 199, travado em memória) para manipular a matriz; assim, toda a busca de instruções será a partir da página 0.

Para uma memória com apenas três quadros de página, indique em que situação, código a) ou código b), ocorre mais faltas de páginas nos seguintes laços de inicialização do vetor, usando o algoritmo de substituição de página LRU em Software (Least Recently Used) e assumindo que a moldura 0 tem a página 0 do processo e que os outros dois estão inicialmente vazios. Justifique sua resposta.

```
(a) for (int j=0; j<100; j++)
    for (int i=0; i<100; i++)
        A[i][j]=0;
```

```
(b) for (int i=0; i<100; i++)
    for (int j=0; j<100; j++)
        A[i][j]=0;
```

- 8) Considere o seguinte padrão de referência de página:

1; 2; 3; 4; 2; 1; 5; 6; 2; 1; 2; 3; 7; 6; 3; 2; 1; 2; 3; 6

Qual o menor número de quadros que evita faltas de páginas em, pelo menos, 50% destas referências, considerando a substituição LRU ? Considere que todos os quadros de páginas estão inicialmente vazios.

- 9) Um pequeno computador tem quatro molduras de página. Na primeira interrupção de tempo, os bits R são 0111 (o da página 0 em 0 e os demais em 1). Nas interrupções seguintes tais valores são 1011, 1010, 1101, 0010, 1010, 1100, 0001. Se o algoritmo do envelhecimento (aging) for usado, com um contador de oito bits, dê o valor dos quatro contadores após a última interrupção.

- 10) Considere um sistema com páginas de 4K, endereçamento lógico de 16 páginas, e endereçamento físico de 8 frames. Considere a seguinte tabela de página do processo em execução:

	bit validade	moldura
0	1	2
1	1	1
2	1	6
3	1	0
4	1	4
5	1	3
6	0	-
7	0	-
8	0	-
9	1	5
10	0	-
11	1	7
12	0	-
13	0	-
14	0	-
15	0	-

- a) Mostre em quais endereços físicos a MMU traduz cada uma das seguintes referências à memória feitas pelo processo corrente: 8292 e 4094.
- b) Mostre como fica uma Tabela Invertida para este sistema, com o mesmo mapeamento, e como os endereços do item anterior são traduzidos com esta tabela.
- c) Mostre como fica uma Tabela em dois níveis para este sistema com o mesmo mapeamento, se o primeiro e segundo níveis suportarem 4 entradas cada. Além disso, mostre como os endereços do item a) são traduzidos com esta tabela.

- 11) Suponha que um processo possa ter no máximo 4 molduras de páginas. Dê uma seqüência de requisições de páginas que faça com que ambas as estratégias, tanto FIFO quanto LRU, substituam a mesma página na primeira troca e dê outra seqüência de requisições que faça com que ambas substituam páginas diferentes.
- 12) Considere um sistema com páginas de 4K, endereçamento de 16 bits e memória física de 16K, cujo Sistema Operacional trabalhe com Memória Virtual Paginada. Considere que nenhum programa esteja em execução e que todas as molduras estejam inicialmente livres. Por simplicidade, suponha que todas as molduras possam ser ocupadas pelo programa (embora isto seja irreal, já que pelo menos o Sistema Operacional deve estar em execução e ocupando espaço de memória). Suponha que um programa entre em execução e faça referência aos seguintes endereços virtuais, nesta ordem: 200, 8191, 4196, 16383, 8192, 65535. Se é usado o algoritmo de troca de página LRU, responda:
 - a) Construa uma Tabela de Páginas (Convencional, ou Invertida ou Multinível) para este sistema, indicando os campos existente na Tabela de Páginas.
 - b) Mostre para cada endereço de memória virtual referenciado pelo programa, em qual endereço físico ele é traduzido, e como este processo é feito pela MMU e pelo Sistema Operacional. Atualize a Tabela de Páginas, quando necessário.
- 13) Compare as 2 estruturas de dados para controlar os blocos de memória livre, mostrando vantagens e desvantagens de cada uma delas.
- 14) Analise a veracidade das afirmativas abaixo, justificando sua resposta:
 - a) Quando se aumenta a quantidade de memória diminui-se a taxa de falta de página.
 - b) Um esquema de segmentação pura causa fragmentação interna e externa.
 - c) Os motivos que levam o processador a ficar muito ocioso num sistema multitarefa, podem ser pouca quantidade de memória quanto tamanho de página muito pequeno.

d) Programas mal projetados podem causar muitas faltas de páginas

e) MMU é um dispositivo responsável pelo tratamento de falta de página.

15) Considere um sistema de memória composto de 4 molduras de páginas de 2k cada em um espaço de endereçamento total de 16 bits. Se no instante 233 os seguintes dados estão disponíveis:

Moldura	Página Virtual	Carga	Última Referência	R	M
1	7	100	231	0	1
2	8	112	233	1	1
3	5	187	230	0	1
4	10	200	232	0	0

Mostre qual página será removida da memória para se atender a requisição de acesso ao endereço virtual 4095, para cada um dos seguintes algoritmos: FIFO, NRU, LRU e 2a. chance.