

UFES - DEPARTAMENTO DE INFORMÁTICA

3ª. Prova de Sistemas Operacionais - Período: 2015/1 - Profª. Roberta Lima Gomes

- 1) **(1,5)** No segundo trabalho prático da disciplina, os grupos deveriam implementar uma solução para um problema de concorrência e sincronização clássico entre threads. O grupo tinha opção de desenvolver o trabalho em C, usando a API pthreads, ou em Java, usando o monitor da classe Objeto. Indique qual foi a opção feita pelo seu grupo (e por quê?) e descreva como funciona a sincronização de threads no caso da escolha do seu grupo (se seu grupo implementou nas duas linguagens, basta descrever uma delas).
- 2) **(2,0)** Um colega te contou que ele implementou um algoritmo de substituição de página “NRU (Not Recently Used) melhorado”. Qual a diferença entre NRU (o tradicional, sem melhorias) e o LRU (Least Recently Used)? Por que o “verdadeiro” LRU é difícil de ser implementado? Seu colega te mostrou uma série de experimentos rodando o mesmo programa utilizando o tal “NRU melhorado” com diferentes quantidades de molduras alocadas. Você percebeu que, estranhamente, o algoritmo apresentou mais Page Faults quando havia 6 molduras do que quando havia 4 molduras alocadas. Como você explicaria esse “fenômeno” para seu colega?
- 3) **a. (1,0)** Explique o que é o *Working Set* (Conjunto de Trabalho) de um processo. Explique por que o *Working Set* de um processo apresenta uma boa estimativa do número de frames (molduras) que o sistema operacional deveria alocar para o processo.
b. (1,0) O que acontece se a soma dos tamanhos dos *Working Sets* de todos os processos em execução no sistema for maior do que o número total de frames da memória principal. Explique uma estratégia que o sistema operacional poderia usar para melhorar seu desempenho neste caso.

	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	-

- 4) **(2,0)** Considere um sistema com páginas de 8K, endereçamento lógico de 16 páginas, e endereçamento físico de 8 frames. Considere (à esquerda) a tabela de página do processo em execução. Mostre em quais endereços físicos a MMU traduz cada uma das seguintes referências à memória feitas pelo processo corrente: 0x0FFFF e 0x056A1. Lembre-se de indicar o número de bits tanto do endereço virtual quanto do endereço físico.

- 5) **(2,5)** No contexto do sistema de arquivos do UNIX, explique os passos necessários para se carregar em memória todo o conteúdo do arquivo/tmp/notas.txt de 270kbytes. O que mudaria se o arquivo tivesse 11 kbytes? Assuma blocos de 1 kbytes e inodes node com 10 ponteiros diretos para blocos, um indireto simples, um indireto duplo e um indireto triplo. Assuma também que cada ponteiro para bloco tem 32 bits.