

## UFES - DEPARTAMENTO DE INFORMÁTICA

### 2ª. Prova de Sistemas Operacionais - Período: 2015/2 - Profª. Roberta Lima Gomes

- 1) **(1,5)** Sobre o primeiro trabalho prático da disciplina, explique como você (sua dupla) implementou o comando interno `wait`, o qual deveria fazer com que a *shell* libere todos os processos filhos que estejam no estado “Zombie” antes de exibir um novo *prompt*.
- 2) **(2,5)** Considere o problema dos leitores e escritores, em que existem diversos processos que eventualmente fazem acessos de leitura a uma base de dados e diversos processos que eventualmente fazem acessos de escrita à mesma base. Vários acessos de leitura podem ocorrer simultaneamente, mas um acesso de escrita não pode ocorrer simultaneamente com acessos de nenhum tipo. Considere o código a seguir para os processos de leitura e de escrita. Suponha que todos os semáforos são iniciados com valor 1:

leitor:	escritor:
...	...
1 while(1) {	21 while(1) {
2 P(R);	22 ...produz
3 P(M);	23 P(R);
4 rc++;	24 P(W);
5 if (rc==1) P(W);	25 ESCREVE;
6 V(M);	26 V(W);
7 V(R);	27 V(R);
8 LÊ;	28 }
9 P(M);	
10 rc--;	
11 if (rc==0) V(W);	
12 V(M);	
13 ... consome	

- (a) Essa solução pode levar a *starvation* de algum dos processos? Explique sua resposta, usando os números das linhas de código para se referir aos passos do programa.
  - (b) Explique o papel do semáforo M. Dê um exemplo de problema que poderia ocorrer caso as operações sobre ele fossem retiradas.
- 3) **(2,0)** Emule um semáforo S usando a construção de um monitor, considerando as operações *Semaforo\_Wait* e *Semaforo\_Signal*. Use pseudo-código, seguindo a estrutura a seguir:

```
Monitor Emulação_Semaforo
//Declaração das variáveis compartilhadas (incluindo variáveis de condição) do monitor
...
Operation Semaforo_Wait
    //Código da operação de entrada “Semaforo_Wait”
    ...
end Semaforo_Wait;
Operation Semaforo_Signal
    //Código da operação de entrada “Semaforo_Wait”
    ...
end Semaforo_Signal;
end Monitor.
```

- 4) **(2,5)** Em um sistema que suporta programação concorrente apenas por meio da troca de mensagens, será criado um Servidor para controlar o uso das portas seriais. Quando um processo Cliente deseja usar uma porta serial, ele envia uma mensagem “Aloca” para o Servidor. Existem N portas seriais, todas equivalentes, mas cada uma pode ser usada somente por um Cliente de cada vez. O Servidor informa ao Cliente a porta que ele vai usar por meio da mensagem “Porta p”. Ao concluir o uso, o Cliente envia para o Servidor a mensagem “Libera p”. Suponha que existam mais do que N processos Clientes. Mostre o algoritmo do Servidor, em português estruturado. Supor “receive” bloqueante.
- 5) **(1,5)** Por que as bibliotecas que implementam threads no nível do usuário oferecem funções de semáforo, dado que tais funções já são oferecidas pelo sistema operacional?

Boa Prova !!!