

## Exercícios Sistemas de Computação – Lista 2

- 1) Escreva um programa que implemente uma comunicação entre processos pai e filho de forma que o pai determine a preempção do filho, ou seja, o pai sistematicamente (a cada 2 segundos) interrompe a execução do filho por um determinado tempo (outros 2 segundos) e em seguida libera a continuidade de sua execução.

Dica: Usar sinais (para interrupção da execução de um processo – sinal SIGSTOP - e para a continuidade da sua execução – sinal SIGCONT).

`int kill ( pid_t pid, int sig )`

envia um sinal para o processo pid

- Se `pid > 0`, o sinal é enviado ao processo pid
- Se `pid = 0`, o sinal é enviado a todos os processos do grupo ao qual pertence o processo corrente
- Se `pid = -1`, o sinal é enviado a todos os processos para os quais o processo atual tem permissão de fazer (exceto para o processo 1)
- Se `pid < -1`, o sinal é enviado a todos os processos do grupo `-pid`

- 2) Quais das instruções a seguir devem ser executadas apenas em modo kernel? Explique suas respostas.

- a. Desabilitar todas as interrupções,
- b. consultar a data e hora do sistema,
- c. alterar a data e hora do sistema,
- d. alterar informações residentes no núcleo do sistema,
- e. somar duas variáveis declaradas dentro do programa,
- f. realizar um desvio para uma instrução dentro do próprio programa,
- g. acessar diretamente posições no disco.

- 3) Explique a diferença entre processos foreground e background.

- 4) Qual a diferença entre preempção por tempo e preempção por prioridade?

- 5) Considere um sistema multitarefa com um único processador. Os processos são criados nesse sistema segundo as informações na tabela abaixo:

Desenhe o diagrama de tempo mostrando a alocação da UCP para cada um dos três processos segundo as condições especificadas, calcule e apresente seus respectivos tempos de **turnaround**:

Considere um sistema utilizando escalonamento circular com fatia de tempo (*time slice*) igual a 2 u.t. Desconsidere o tempo de mudança de contexto (troca de contexto) entre os processos. Considere, para sua resposta, que quando dois processos chegam **juntos** à fila de “prontos”, a ordem de

preferência será para aquele que vier da fila de “novos”, “bloqueados” e “em execução.”

<b>Processo</b>	<b>Instante de Ativação (ut)</b>	<b>Tempo total de CPU (ut)</b>	<b>Instantes de tempo de ocorrência das operações de E/S (com relação ao seu tempo de CPU)</b>	<b>Tempo gasto pela operação de E/S</b>
P1	0	5	---	---
P2	2	6	2	3
P3	3	3	1 I/O a cada 1 ut	2
P4	5	2	---	---
P5	6	4	3	1