

AULA 11 - Exercícios

01. Em um sistema operacional , o escalonador utiliza duas filas. A fila **A** contém os processos do pessoal do CPD e a fila **B** contém os processos dos alunos. O algoritmo entre filas é fatia de tempo (Round Robin). De cada 11 unidades de tempo de processador, 7 são fornecidas para os processos da fila **A**, e 4 para os processos da fila **B**. O tempo de cada fila é dividido entre os processos também por fatias de tempo, com fatias de 2 unidades para todos. A tabela abaixo mostra o conteúdo das duas filas no instante zero. Considere que está iniciando um ciclo de 11 unidades, e agora a fila **A** vai receber as suas 7 unidades de tempo. Mostre a seqüência de execução dos processos, com os momentos em que é feita a troca.

Obs. Se terminar a fatia de tempo da fila **X** no meio da fatia de tempo de um dos processos, o processador passa para a outra fila. Entretanto, esse processo permanece como primeiro da fila **X**.

Fila	Processo	Duração
A	P1	6
A	P2	5
A	P3	7
B	P4	3
B	P5	8
B	P6	4

02. Considere os 4 processos listados a seguir e simule a execução dos algoritmos: FIFO, SJF e Prioridade.

Processo	Instante de Chegada	Tempo de execução	Prioridade
P1	0	8	10
P2	1	4	5
P3	2	9	8
P4	3	5	3
P5	4	10	12
P6	5	14	2

03. Um algoritmo de escalonamento de CPU determina uma ordem de execução de seus processos escalonados. Dados n processos a serem escalonados em um processador, quantas maneiras diferentes de escalonar estes processos são possíveis?

04. Defina a diferença entre escalonamento preemptivo e não-preemptivo. Explique por que o escalonamento não-preemptivo é improvável de ser utilizado em um centro de computação.

05. Cinco processos, de A até E, chegam ao computador ao mesmo tempo. Eles têm seus tempos de processamento estimados em 10, 6, 2, 4 e 8 minutos respectivamente. Suas prioridades (atribuídas externamente) são 3, 5, 2, 1 e 4, respectivamente, sendo 5 o representante da prioridade mais alta. Nenhum dos processos faz I/O. Para cada um dos algoritmos de escalonamento abaixo, determine o tempo médio de turnaround dos processos. Ignore o overhead causado pela troca de contexto.

- (a) Round Robin
- (b) Escalonamento com prioridade
- (c) FIFO (ordem de execução: 10, 6, 2, 4, 8)
- (d) SJF

Para (a), assumo que a fila começa em A, indo em ordem até E (quantum = 4).

06. O sistema operacional Linux utiliza dois algoritmos de escalonamentos, um de tempo compartilhado e outro baseado em prioridades, para tratar tarefas de tempo real. Cada processo é associado a uma classe de escalonamento. No algoritmo de escalonamento para processos de tempo compartilhado, o Linux usa um algoritmo por prioridades, baseado em créditos. Cada processo possui um determinado número de créditos (inicialmente, o número de créditos é igual à prioridade do processo); o processo com o maior número de créditos na fila de prontos é selecionado pelo escalonador. A cada interrupção do temporizador, o processo em execução perde um crédito; quando seu crédito chega a zero, o escalonador é ativado para selecionar outro processo para ganhar o processador. Se nenhum processo na fila de prontos tiver créditos, o algoritmo faz nova atribuição de créditos a todos os processos (inclusive aos processos bloqueados), de acordo com a seguinte regra:

$$\text{créditos} := \text{créditos} / 2 + \text{prioridade}.$$

Avalie o algoritmo de escalonamento por prioridade usado pelo Linux e o Algoritmo por Revezamento (quantum = 3ms) quanto à utilização da CPU e tempo médio de resposta. Para tanto, considere o seguinte volume de trabalho:

ordem	processo	surto de CPU*	duração de I/O	tempo total de CPU	prioridade
1	A	2 ms	5 ms	6 ms	3
2	B	3 ms	10 ms	6 ms	3
3	C	---	---	14 ms	3
4	D	---	---	10 ms	3

(*) tempo de CPU necessário antes de cada solicitação de I/O (processos A e B ficam alternando entre surtos de CPU e em operações de I/O).