

UFES – Departamento de Informática

1ª. Prova de Sistemas Operacionais – 2015/2 – Profª. Roberta L. Gomes

Nome: _____

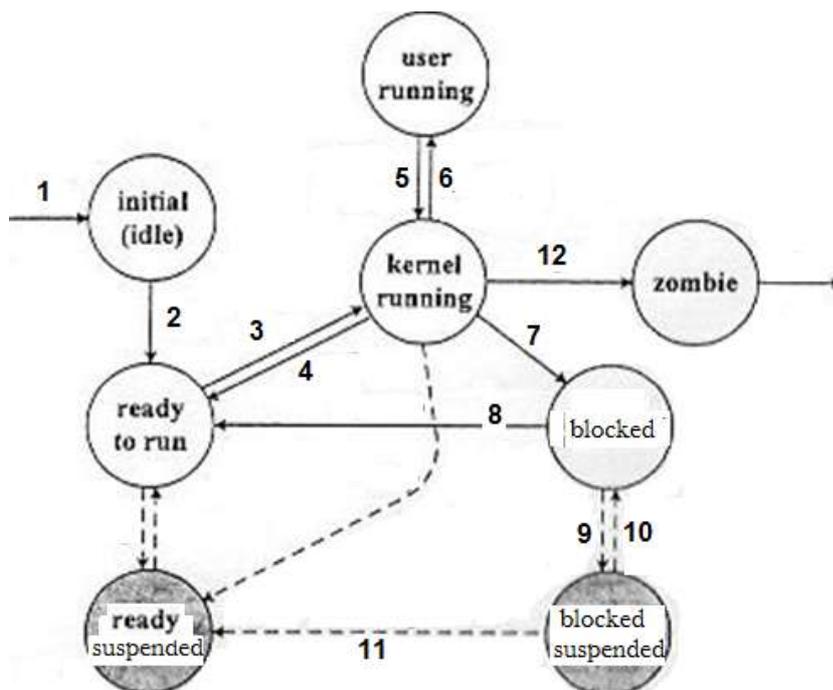
- 1) (2,5) Considere as seguintes estratégias de escalonamento: FIFO, Round-Robin, Prioridades, SJF (*Shortest Job First*), Multinível. O sistema executa muitos tipos de processos (curtos, longos, interativos, em lotes, CPU bound, I/O bound, real-time, etc.). Para cada item da lista de requisitos abaixo, de (a) a (e), :

- Diga **qual a estratégia que melhor** satisfaz considerando todos eles (ou a maior parte deles) simultaneamente;
- Para **cada uma das demais estratégias de escalonamento**, explique porque ela não consegue satisfazer pelo menos um dos requisitos abaixo.

REQUISITOS:

- O número de processos interativos é relativamente pequeno. Eles deverão ter alta prioridade, mas são aceitáveis pequenos atrasos.
- O número de processos de tempo-real também é pequeno, mas além de ter alta prioridade, precisam de garantias de tempo de resposta.
- Processos I/O bound devem ter prioridade alta para manter ativos os processadores de I/O.
- Processos CPU bound podem retardar processos I/O bound, mesmo que esses já estejam esperando há muito tempo.
- Quando os processos CPU bound são os únicos que estão prontos, a sobrecarga do sistema operacional deve ser minimizada.

- 2) (2,0) Dado o diagrama abaixo, descreva e explique os eventos que provocam cada uma das transições entre estados (lembre-se de explicar por que/em que condições cada transição pode ocorrer).



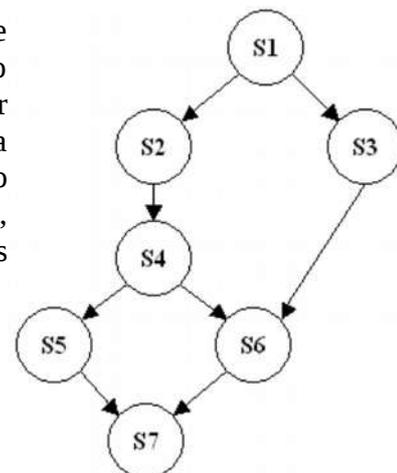
3) (2,0) Cinco processos A, B, C, D and E chegam nesta ordem ao mesmo tempo com os seguintes CPU burst e prioridades (menor valor significa maior prioridade):

	CPU Burst	Prioridade	Chegada
A	3	3	1
B	7	5	0
C	4	1	5
D	1	4	2
E	2	2	6

Preencha o quadro abaixo com o tempo de espera (*Waiting Time*) de cada processo, e o tempo de espera médio (*Average Waiting Time*) para cada algoritmo de escalonamento indicado (ignore o overhead devido a trocas de contexto).

<i>Scheduling Policy</i>	<i>Waiting Time</i>					<i>Average Waiting Time</i>
	<i>A</i>	<i>B</i>	<i>C</i>	<i>D</i>	<i>E</i>	
First-Come-First-Served						
Preemptive Shortest-Job First						
Priority						
Round-Robin (time quantum=2)						

4) (2,0) Um grafo de precedência é um grafo direcionado em que a relação (a) → (b) indica que 'a' precede 'b'. Na figura ao lado, os nós do grafo representam funções que devem ser executadas por processos. Dessa forma, o gráfico de precedência ilustrado representara a ordem em que as funções serão executadas. Usando as chamadas fork() e wait() no UNIX, implemente um programa em C que seja capaz de executar as funções "Sx()" na ordem indicada pelo grafo.



5) (1,5) No UNIX, o que são e para que servem Sinais? Explique o que significa *Capturar*, *Bloquear* e *Ignorar* um Sinal.

Boa Prova!