

ALUNO: AXELL BRENDOW

1	2	3	4	5	6	7	8	9
B	D	D	C	B	A	E	D	C

**Questão 1** - Em um sistema operacional, um processo pode, em um dado instante de tempo, estar em um de três estados: em execução, pronto ou bloqueado. Considere as afirmativas abaixo sobre as possíveis transições entre estes estados que um processo pode realizar.

- I. Do estado em execução para o estado bloqueado
- II. Do estado em execução para o estado pronto
- III. Do estado pronto para o estado em execução
- IV. Do estado pronto para o estado bloqueado
- V. Do estado bloqueado para o estado em execução
- VI. Do estado bloqueado para o estado pronto

Quais são as afirmativas verdadeiras?

- (a) Somente as afirmativas I, II e VI são verdadeiras.
- (b) Somente as afirmativas I, II, III e VI são verdadeiras.
- (c) Somente as afirmativas I, III, IV e VI são verdadeiras.
- (d) Somente as afirmativas I, III, IV e V são verdadeiras.
- (e) Todas as afirmativas são verdadeiras.

**Questão 2** - Os sistemas operacionais modernos utilizam o conceito de fila circular no escalonamento de processos. O processo que está no início da fila de processos prontos é selecionado, executado por algum tempo e, ao término da fatia de tempo, retorna para o final da fila. O mecanismo apresentado permite que as aplicações sejam:

- a) executadas de forma aleatória dentro da fila de processos prontos.
- b) executadas apenas uma vez, pois o esquema de filas não permite que processos já selecionados possam retornar para a mesma fila.
- c) executadas conforme são criadas ou esgotem as suas fatias de tempo.
- d) selecionadas conforme a sua prioridade dentro do sistema.
- e) selecionadas no meio da fila, por terem mais prioridade que os demais processos.

**Questão 3** - Alguns dos objetivos dos algoritmos de escalonamento de processos são comuns a todos os tipos de sistemas operacionais. Outros, entretanto, variam de acordo com o tipo de sistema. Qual dos objetivos abaixo **NÃO** se aplica a algoritmos de escalonamento de processos?

- a) Atender às requisições dos usuários o mais rápido possível.
- b) Manter a CPU ocupada o tempo todo.
- c) Maximizar o número de tarefas processadas por unidade de tempo.
- d) Manter os dispositivos de E/S ocupados o máximo de tempo possível.
- e) Minimizar o tempo entre a submissão e o término de um job.

**Questão 4** - Considere um sistema operacional com escalonamento por prioridades, no qual a avaliação do escalonamento é realizada em um intervalo mínimo de 5 ut. Neste sistema, os processos A e B competem por uma única UCP. Desprezando os tempos de processamento relativo às funções do sistema operacional, a tabela a seguir fornece os estados dos processos A e B ao longo do tempo, medido em intervalos de 5 ut (E = execução, P = pronto e W = espera). O processo A tem menor prioridade que o processo B.

Em que tempos A sofre preempção?

	00-04	05-09	10-14	15-19	20-24	25-29	30-34	35-39	40-44	45-49
Processo A	P	P	E	E	E	P	P	P	E	W
Processo B	E	E	W	W	P	E	E	E	W	W

	50-54	55-59	60-64	65-69	70-74	75-79	80-84	85-89	90-94	95-99	100-105
Processo A	P	E	P	P	E	E	W	W	P	E	E
Processo B	W	P	E	E	W	W	P	E	E	–	–

- a) Entre os instantes 9-10, 39-40, 69-70.
- b) Entre os instantes 19-20, 54-55, 79-80.
- c) Entre os instantes 24-25 e 59-60.
- d) Entre os instantes 79-80.
- e) Nunca, pois o processo B tem maior prioridade do que o processo A.

**Questão 5** - *Starvation* ocorre quando:

- a) A prioridade de um processo é ajustada de acordo com o tempo total de execução do mesmo.
- b) Pelo menos um processo é continuamente postergado e não executa.
- c) Pelo menos um evento espera por outro evento que não vai ocorrer.
- d) Dois ou mais processos são forçados a acessar dados críticos alternando estritamente entre eles.
- e) O processo tenta mas não consegue acessar uma variável compartilhada.

**Questão 6** - Qual das seguintes condições **não** é necessária para a ocorrência de um *deadlock*?

- a) Escalonamento preemptivo de recursos.
- b) Alocação parcial de recursos a processos.
- c) Uso mutuamente exclusivo de recursos por processos.
- d) Processos em espera circular.
- e) Necessidade de compartilhamento de recursos por processos.

**Questão 7** - Considere o seguinte código que implementa exclusão mútua entre dois processo i e j:

```

Processo Pi
    while (1) {
        while (turn != i) ;    // entrada seção crítica
                               seção crítica
        turn = j;              // saída da seção crítica
                               código restante
    }

```

De acordo com o código acima pode-se afirmar, exceto:

- a) A solução garante exclusão mútua.
- b) Um processo bloqueia o outro mesmo não estando na seção crítica.
- c) Os processos fazem espera ativa.
- d) Existe alternância estrita.
- e) A solução garante progresso.

**Questão 8** - Considere o seguinte programa com dois processos concorrentes. O escalonador poderá alternar entre um e outro, isto é, eles poderão ser intercalados durante sua execução. As variáveis x e y são compartilhadas pelos dois processos e inicializadas antes de sua execução.

```

programa P
    int x = 0;
    int y = 0;

    processo A {
        while (x == 0);
        print("a");
        y = 1;
        y = 0;
        print("d");
        y = 1;
    }

    processo B {
        print("b");
        x = 1;
        while (y == 0);
        print("c");
    }

```

As possíveis saídas são:

- a) adbc ou bcad    b) dbca ou dcab    c) abdc ou abcd    d) badc ou bacd    e) Nenhuma das opções

**Questão 9** - O escalonamento de CPU trata do problema de decidir qual dos processos na fila de prontos deve ser entregue à CPU. Considere que o algoritmo de escalonamento Round-Robin esteja sendo utilizado e que o conjunto de processos abaixo chegue no momento 0, com a extensão do tempo de burst de CPU indicada em milissegundos. Dado: P1 é o primeiro processo na fila de prontos, P2 é o segundo e P3 é o terceiro.

Processo	Tempo de burst
P1	10
P2	3
P3	4

Se for utilizado um quantum de 4 milissegundos, o tempo de RETORNO de P1 será de:

- a) 21    b) 13    c) 17    d) 24    e) 18

Transições Processo \ Instante	P1	P1/2	P2/3	P3/1	Ø
	0	4	7	11	17
P1	10	6	6	6	0
P2	3	3	0	0	0
P3	4	4	4	0	0

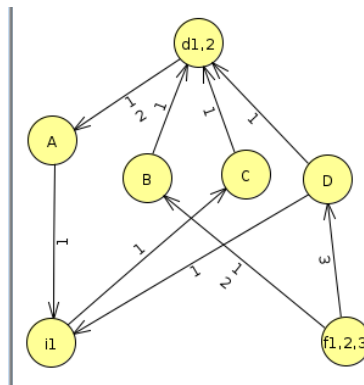
A parte laranja contém o tempo restante de cada processo para cada instante.

**Questão 10** - Considere a situação em que 4 processos A, B, C, D concorrem por recursos da máquina. Há 2 unidades de disco, 3 unidades de fita e 1 unidade de impressão. Os processos se encontram na seguinte situação:

- O processo A está de posse de 2 unidades de disco;
- O processo B está de posse de 2 fitas e requisita 1 unidade de disco;
- O processo C está de posse de 1 unidade de impressão e requisita uma unidade de disco;
- O processo D está de posse da outra fita;

O processo D requisita a unidade de impressão e, logo após, o processo A faz a mesma requisição. A situação acima leva a leva a deadlock?

**Obs: mostre seu raciocínio com o grafo de alocação de recurso!**



**Há deadlock. O processo A está na espera de uma unidade de impressão, porém o processo C está de posse dela. Já o processo C requisita uma unidade de disco, porém o processo A está de posse de todas as unidades de disco.**