

14.4

0.6 AVALIAÇÃO – UNIDADE 2 – PROCESSOS E THREADS – 09/05/2016

1) Considere que cinco processos sejam criados no instante de tempo 0 (P1, P2, P3, P4 e P5) e possuam as características descritas na tabela a seguir:

4

	Process o	Tempo de UCP	Prioridade
3º	P1	10	3
4º	P2	14	4
1º	P3	5	1
2º	P4	7	2
5º	P5	20	5

p < = ()

Desenhe um diagrama ilustrando o escalonamento dos processos e seus respectivos tempos de turnaround, segundo as políticas especificadas a seguir. O tempo de troca de contexto deve ser desconsiderado.

- a) FIFO
- b) SJF
- c) Prioridade (número menor implica prioridade maior) Circular com fatia de tempo igual a 2 u.t.

2) Preencha com nome do algoritmo de escalonamento correto:

- 1.0
- a) Round Robin: Especialmente útil para sistemas de tempo compartilhado. Cada processo ganha um tempo limite para sua execução. Após esse tempo ele é interrompido e colocado no fim da fila de prontos.
 - b) prioridade: A cada processo é associada uma prioridade e a CPU é alocada para o processo com a mais alta prioridade.
 - c) SJF: Algoritmo projetado especialmente para jobs em lote, onde o tempo de execução é conhecido previamente, funciona, por exemplo, quando processos de mesma importância estão na fila para serem iniciados pela CPU, o agendador utiliza primeiro o processo menor, tendo um tempo de resposta mais rápido.

0.8 3) Preencha com:

TMU – Thread Modo Usuário,
TMK – Thread Modo Kernel -
TMH – Thread Modo Híbrida

- a) TMK: O sistema operacional sabe da existência de cada thread e pode escaloná-los individualmente.
- b) TMH: Um TMU pode ser executado em um TMK, em um determinado momento, e no instante seguinte ser executado outro.
- c) TMU: Possuem uma grande limitação, pois o sistema operacional gerencia cada processo como se existisse apenas um único thread.
- d) TMH: Um TMU pode ser executado em um TMK, em um determinado momento, e no instante seguinte ser executado outro.
- e) TMH: A vantagem deste modelo é a possibilidade de implementar aplicações multithreads mesmo em sistemas operacionais que não suportam threads.

4) No escalonamento TMK quando um processo está em execução nenhum evento externo pode ocasionar a perda do uso da CPU, já no escalonamento TMH o sistema operacional pode interromper um processo em execução e passá-lo para o estado de pronto.

5) Cite 3 critérios de uma política de escalonamento: Não preempção, espera por recurso, exclusão mútua.

6) Qual nome se dá a situação em que um processo aguarda por um recurso que nunca estará disponível ou um evento que nunca ocorrerá? Espera por recurso.

7) Dois PROCESSOS atualizam SALDO de um mesmo CLIENTE.

O processo 1 (do primeiro funcionário) lê o registro do cliente e soma ao campo Saldo o valor do lançamento do débito; Antes de gravar o novo saldo do arquivo, o processo 2 (do segundo funcionário) lê o registro do mesmo cliente e soma ao campo Saldo o valor creditado;

Em uma aplicação concorrente que controla saldo bancário em contas correntes, dois processos compartilham uma região de memória onde estão armazenados os saldos dos clientes A e B. Os processos executam, concorrentemente os seguintes passos:
Processo 1 (Cliente A) Processo 2 (Cliente B)

/* saque em A */

1a. $x := \text{saldo_do_cliente_A}; = 50$
1b. $x := x - 200; = 50 - 200 = -150$
1c. $\text{saldo_do_cliente_A} := x; /*saque em A*/ = -150$
2a. $y := \text{saldo_do_cliente_A}; = -150$
2b. $y := y - 100; = -150 - 100 = -250$
2c. $\text{saldo_do_cliente_A} := y; = -250$

/* deposito em B */

1d. $x := \text{saldo_do_cliente_B}; = 100$
1e. $x := x + 100; = 100 + 100 = 200$
1f. $\text{saldo_do_cliente_B} := x; /* deposito em B */ = 200$
2d. $y := \text{saldo_do_cliente_B}; = 200$
2e. $y := y + 200; 200 + 200 = 400$
2f. $\text{saldo_do_cliente_B} := y; = 400$

Supondo que os valores dos saldos de A e B sejam, respectivamente, 50 e 100, antes de os processos executarem, pede-se:

a) Quais os valores corretos esperados para os saldos dos clientes A e B após o término da execução dos processos? -

a) Quais comando adicionaria ao algoritmo para garantir as especificações corretas de concorrência do programa para que o mesmo não apresente inconsistência?

8) Cite os principais estados de um processo no seu ciclo de vida normal?

9) A área que impede que mais de um processo acesse o dado compartilhado ao mesmo tempo cuja implementação é essencial para sistemas operacionais multiprogramados é chamada de? exclusão mútua

Região crítica que faz parte da Exclusão Múltipla.

10) Qual dos contextos não é compartilhado entre processos e threads?

Exclusão mútua → cada recurso só pode estar alocado a um único processo em determinado instante.

1)

P1		10																	
P2				14															
P3						5													
P4								7											
P5										20									
P1						10													
P2								14											
P3		6																	
P4				7															
P5										15									
P1																			
P2																			
P3																			
P4																			
P5																			

2) a) Round-Robin (circular) b) Prioridade c) SJF

3) a) TMK b) TMH c) TMU d) TMH e) TMU

4) Não Preemptivo – Preemptivo

5) Prioridade, Tempo de processamento, ordem de chegada

6) Startivation

7) a) -250 – 400 b) PARBEGIN – PAREND

8) Pronto, Em Execução , Em espera - Stopped – Running - Waiting

9) Área Crítica

10) Contexto de Hardware