

Avaliação:	<b>Avaliação de Verificação Escolar - 2º bimestre</b>	Data: <b>25/11/2020</b>	<b>BCC</b>
		Termo: 4º	Turma: A
Disciplina:	<b>Sistemas Operacionais I</b>	Professor: <b>Robson A. Siscoutto</b>	

### Instruções para a Avaliação:

- 1 – A prova é pessoal e individualmente e respostas idênticas não serão aceitas;
- 2 – Antes de começar a responder as questões leia a prova atentamente e essas instruções;
- 3 – Não esqueça de colocar seu NOME e sua RA – Arquivo de Respostas;
- 4 – Responda as questões na ordem;
- 5 – Não é obrigatório colocar as perguntas da prova, apenas as respostas identificadas;
- 6 – Abaixo está o quadro de notas com os pesos de cada atividade de SO1 do 2 Bimestre.

### BOA PROVA!!!

#### \*Atividade(s) do quadro de notas

Atividade(s)	Peso (0-100)%
1 - Entrega do Algoritmo por vetor de Interrupção e do Algoritmo de Dekker (TR = Atividades e Avaliações Parciais Desenvolvidas durante o Bimestre)	4
2 - Entrega da Lista de Exercícios de Concorrência de Processos (TR = Atividades e Avaliações Parciais Desenvolvidas durante o Bimestre)	4
3 - Entrega dos Problemas Clássicos de SO (TR = Atividades e Avaliações Parciais Desenvolvidas durante o Bimestre)	6
4 - Avaliação Parcial sobre Concorrência de Processos (TR = Atividades e Avaliações Parciais Desenvolvidas durante o Bimestre)	8
5 - Avaliação Parcial sobre Deadlock (TR = Atividades e Avaliações Parciais Desenvolvidas durante o Bimestre)	8
6 - Entrega da Verificação de Aproveitamento Escolar - 2º Bimestre (25/11/2020) (AB = Avaliação bimestral - Online)	70

### QUESTÃO 1) (1,0)

Quais são as 4 condições que devem ser garantidas para se evitar que dois ou mais processos estejam na sua região crítica ao mesmo tempo, evitando condições de corrida? Explique cada uma delas.

### QUESTÃO 2)

Sobre o algoritmo do Peterson estudados no material da disciplina faça:

- (1,0) Apresente o algoritmo para dois processos concorrentes e faça um teste de mesa demonstrando seu funcionamento.
- (1,0) Rescreva o algoritmo para executar para 4 processos e faça um teste de mesa demonstrando seu funcionamento.

### QUESTÃO 3)

Sobre deadlock pergunta-se:

- (0,5) Explique as condições necessárias para a existência de deadlock em um sistema?
- (0,5) Explique as estratégias utilizadas para tratamento do deadlock e que condições de existência de deadlock estas estratégias anulam?

### QUESTÃO 4) (1,5)

Observe o grafo de alocação de recursos do sistema abaixo e responda as perguntas:

- 3 Conjuntos: P, R e A
  - $P = \{P_1, P_2, P_3\}$ ;  $R = \{R, S, T\}$
- Instâncias de Recurso:
  - 1 instancia de R
  - 1 instância de S
  - 1 instancia de T

A situação atual do grafo de alocação é:  $P_1 \rightarrow R$ ;  $S \rightarrow P_1$ ,  $T \rightarrow P_3$ . Com base nesse grafo de alocação atual, faça:

A) Suponha que  $P_1$  tenha pedido T em vez de S. Refaça o grafo de alocação e explique se isto levaria a uma situação de deadlock?

B) Suponha que o processo  $P_2$  pedido os recursos R e T. Refaça o grafo de alocação e explique se isto levaria a uma situação de deadlock?

### QUESTÃO 5)

Pesquisa e estude o algoritmo do banqueiro para Deadlocks e resolva o exercício abaixo. Considere o seguinte instantâneo de um sistema:

	Alocação				Máximo				Disponível			
	A	B	C	D	A	B	C	D	A	B	C	D
$P_0$	0	0	1	2	0	0	1	2	1	5	2	0
$P_1$	1	0	0	0	1	7	5	0				
$P_2$	1	3	5	4	2	3	5	6				
$P_3$	0	6	3	2	0	6	5	2				
$P_4$	0	0	1	4	0	6	5	6				

Responda as seguintes perguntas utilizando o algoritmo do banqueiro:

- a) (0,5) Qual é o conteúdo da matriz necessário (Need)?
- b) (0,5) O sistema está em estado seguro? Justifique porquê?
- c) (0,5) Se uma requisição do processo P1 chegar para (0,4,2,0), a requisição poderá ser concedida imediatamente? Por quê?
- d) (0,5) Se uma requisição do processo P2 chegar para (1,0,0,2), a requisição poderá ser concedida imediatamente? Por quê? Após a execução de quais processos que P2 poderá ser atendido?

### QUESTÃO 6) (2,5)

Baseada na ideia do problema do produtor-consumidor, considere uma extensão definida da seguinte forma:

- a) Há dois processos consumidores e um processo produtor. Os processos executam concorrente e assincronamente.
- b) O produtor não pode escrever no buffer até que ambos os consumidores tenham lido dele. É aberta uma exceção para a primeira escrita.
- c) Nenhum consumidor pode ler o conteúdo de um buffer indefinido.
- d) Nenhum consumidor pode ler o mesmo conteúdo duas vezes sucessivamente.
- e) Ambos os consumidores podem executar seus eventos simultaneamente.

Defina as variáveis semáforos e use primitivas Down e UP para impor a exclusão mútua e sincronizar os processos.

Processo Produtor	Processo Consumidor1	Processo Consumidor2
...	...	...
repeat	repeat	repeat
...	...	...
coloca dados no buffer	lê dados no buffer	lê dados no buffer
...	...	...
until forever	until forever	until forever