

2ª Avaliação STR (2021.2) (Teoria)

1 mensagem

Formulários Google <forms-receipts-noreply@google.com>
Para: vitor18cesar@alu.ufc.br

14 de dezembro de 2021 19:56

Agradecemos o preenchimento de [2ª Avaliação STR \(2021.2\) \(Teoria\)](#)

Veja as respostas enviadas.

2ª Avaliação STR (2021.2) (Teoria)

Pessoal essa é nossa primeira avaliação de STR 2020.2. Segue algumas instruções a serem seguidas durante a realização dessa avaliação.

INSTRUÇÕES DA PROVA:

1. O aluno não poderá consultar materiais durante a avaliação. Apesar dessa avaliação ser remota e o professor não ter o controle do acesso a outros materiais, peço apenas a consciência de todos.
2. As repostas que necessitarem o upload de imagens podem ser feitas a mão ou digitalmente.
3. Para o envio da prova o aluno tem que fazer o login na conta google gmail ou institucional.
4. O formulário da avaliação só pode ser enviado uma única vez, portanto, confirmem as respostas antes de enviar.

E-mail *

vitor18cesar@alu.ufc.br

Nome Completo do Aluno: *

Vitor Cesar

Matrícula: *

Questão 1 - Cite e explique os métodos que são utilizados em programas concorrentes para sincronização e comunicação entre tarefas *

Na maioria dos programas concorrentes, a sincronização e comunicação entre tarefas pode ser feita através de dois métodos: a troca de mensagens ou message passing, ou então o acesso a variáveis compartilhadas ou shared memory.

Quando temos uma troca de mensagens, as tarefas enviam mensagens umas para as outras. Dessa forma o envio de mensagens serve a dois propósitos, os dados então podem ser trocados por meio do conteúdo das mensagens. E o fato de enviar ou receber uma mensagem também permite que as tarefas sincronizem suas ações.

E quando temos variáveis compartilhadas sendo empregadas, irá existir variáveis que podem ser acessadas de várias tarefas. Essas variáveis vão depender na forma em que esta sendo aplicada na questão, pode ser inteiro, string, ou qualquer tipo existente na linguagem de programação. Em muitos casos são compartilhadas estruturas de dados mais complexas tais como listas encadeadas ou tabelas. Essas variáveis que são compartilhadas vão permitir troca de dados de uma forma direta, e para a sincronização, vai ser disponibilizado pelo sistema operacional, pelo kernel ou pelo microkernel.

Questão 2 - Explique o que é inversão de prioridade e quais mecanismos para sistemas de tempo real são usados para tratar inversões de prioridade. *

A ocorrência de uma inversão de prioridade se dá quando a tarefa que tem a maior prioridade irá ficar esperando a execução da tarefa de menor prioridade.

Temos por exemplos, suponhamos que um sistema de tempo real composto por três tarefas t_1 , t_2 e t_3 , t_1 sendo a prioridade mais alta e t_3 a mais baixa. Essas três tarefas compartilham um recurso x , que esta sendo protegido pelo mutex X , quando a tarefa t_3 chega antes e executa $LOCK(X)$ e após um tempo depois chega a tarefa t_2 , que possui prioridade mais alta que t_3 e, assim, preempta o processador de t_3 , logo em seguida t_2 executa $LOCK(X)$, mas como o mutex está ocupado, ela vai bloquear, e t_3 volta a executar. O mesmo vai acontecer com t_1 , a tarefa t_1 chega, preempta t_3 pois sua prioridade é mais alta, e vai utilizar o processador até executar $LOCK(X)$, quando fica bloqueada. Assim t_1 e t_2 estarão bloqueadas a vão esperar o mutex, enquanto t_3 executa. E isso é uma inversão de prioridade.

Existem algumas soluções que irão gerar bloqueio como wait-free e lock-free. Também teremos os mutexes para tempo real, eles possuem formas para se resolver esses problemas como, desligamento da preempção, herança de prioridade e o teto de prioridade.

Questão 3 *

Considere um sistema composto por três tarefas periódicas, independentes:

τ_1 :	$P_1=4$	$C_1=2$	$D_1=4$
τ_2 :	$P_2=7$	$C_2=3$	$D_2=7$
τ_3 :	$P_3=28$	$C_3=2$	$D_3=28$

Podemos escalonar este sistema com prioridades preemptivas fixas? E com prioridades preemptivas variáveis? Justifique sua resposta. Se possível desenhe a escala de execução até o instante que achar necessário.

Arquivos enviados

 Q3 - Vitor Cesar.pdf

Questão 4 *

Um sistema contém os cinco jobs a seguir, em ordem decrescente de prioridade: J1, J2, J3, J4 e J5. Existe neste sistema dois recursos: X e Y. As necessidades de recursos estão listadas abaixo:

J1:	$C_1 = 5$	composto por	$1 [X; 3] 1$
J2:	$C_2 = 5$	composto por	$1 [Y; 3] 1$
J3:	$C_3 = 4$	composto por	nenhum
J4:	$C_4 = 6$	composto por	$1 [Y; 4] 1$
J5:	$C_5 = 7$	composto por	$1 [X; 2 [Y; 2] 1] 1$

- Determine o tempo máximo de bloqueio de cada um dos jobs, assumindo como política de alocação de recursos Desabilita Preempção.
- Determine o tempo máximo de bloqueio de cada um dos jobs, assumindo como política de alocação de recursos Teto de prioridade (Priority Ceiling).

Arquivos enviados

 Q4 - Vitor Cesar.pdf

Questão 5 *

Um sistema contém os quatro jobs a seguir, em ordem decrescente de prioridade: J1, J2, J3, J4. Existe neste sistema três recursos: X, Y e Z. Os instantes de liberação (release) de cada job e as suas necessidades de recursos estão listadas abaixo:

J1:	$r_1=6$	$C_1=3$	$1[Z,1]1$
J2:	$r_1=4$	$C_1=5$	$1[Z,1[Y,1],1]1$
J3:	$r_1=2$	$C_1=5$	$1[Y,1[X,1]1]1$
J4:	$r_1=0$	$C_1=5$	$1[X,3]1$

Desenhe a escala de tempo deste sistema, de zero até a conclusão do último job, considerando que recursos são gerenciados usando teto de prioridade.

Arquivos enviados

 Q5 - Vitor Cesar.pdf

[Crie seu próprio formulário do Google.](#)

[Denunciar abuso](#)