

Conectividade em Sistemas Ciberfísicos – Avaliação I  
RA1 (2 PONTOS)

ESTUDANTE:

**QUESTÃO 1 (0.5).** Em relação as funções de um sistema operacional indique as afirmativas corretas. Caso considere que uma afirmativa é FALSA, apresente uma breve explicação sobre o que está errado na afirmativa.

- ( ) A gerencia de processos é responsável por controlar o uso da memória RAM (Random Access Memory) pelos programas em execução, isto é, processos.
- ( ) A gerência de armazenamento controla o acesso a dispositivos como discos de estado sólido (SDD) e discos rígidos (HDD).
- ( ) A gerência de memória é responsável por determinar qual programa se tornará ativo no computador, e terá direito de usar os recursos da CPU.
- ( ) A gerência de E/S é responsável por controlar dispositivos como mouse e teclado .

**QUESTÃO 2 (0.5).** Sobre o gerenciamento de processos em sistemas operacionais indique as alternativas verdadeiras. Não é necessário justificar as alternativas. Ao invés disso responda a pergunta formulada.

- ( ) Processos que realizam muitas operações de E/S consomem pouco tempo de CPU, pois são colocados em estado de espera até que a operação seja completada.
- ( ) Quando um processo é colocado em estado de espera, o sistema operacional escolhe outro processo para ocupar o tempo de CPU utilizando um algoritmo de escalonamento.
- ( ) Uma operação de leitura do teclado (como **input()** no Python) é do tipo bloqueante, e coloca o processo que fez a chamada em estado de espera até que ela seja completada.
- ( ) Em um sistema com escalonamento preemptivo, um processo pode ocupar indefinidamente o tempo de CPU até que ele seja bloqueado por uma operação de E/S ou ceda à sua vez através de uma chamada que o coloque estado de espera (como **time.sleep()** no Python, por exemplo).

Qual a diferença entre escalonamento preemptivo e cooperativo?

**QUESTÃO 3 (0.5).** Indique a diferença entre processos e threads e responda a pergunta formulada.

<input type="checkbox"/> Compartilham o mesmo espaço de memória.	1. Threads
<input type="checkbox"/> Podem compartilhar variáveis em um mesmo programa.	2. Processos
<input type="checkbox"/> Pode ser executado em núcleos diferentes de um processador.	3. Ambos
<input type="checkbox"/> Permite que a troca de contexto seja executada de forma relativamente rápida.	4. Nenhum dos dois
<input type="checkbox"/> Permite que funções de um mesmo programa sejam executadas de forma paralela.	

Por que não é possível compartilhar variáveis entre processos diferentes?

**QUESTÃO 4 (0.5).** Em relação a processos e threads indique as alternativas corretas. Leia as alternativas em sequência, pois elas se referem a um mesmo projeto. Não é necessário justificar as alternativas. Ao invés disso responda a pergunta formulada.

- ☐ É possível desenvolver um servidor que atende a múltiplos clientes de forma concorrente usando uma thread diferente para tratar cada cliente ou usando um processo diferente para tratar cada cliente.
- ☐ O servidor desenvolvido com threads será mais eficiente, pois o tempo para criar e chavear threads (trocar de contexto) é menor que o tempo para criar e chavear processos.
- ☐ A abordagem baseada em processos não permitirá que as informações sejam compartilhadas entre clientes diferentes usando variáveis, pois o sistema operacional coloca os processos em espaços de memória diferentes.
- ☐ Para compartilhar informações entre processos diferentes é necessário utilizar uma forma de comunicação entre processos, como sockets, mesmo que os processos estejam no mesmo computador.
- ☐ Na abordagem baseada em processos, uma violação do espaço de memória causada por um processo pode fazer com que o sistema operacional encerre o programa servidor e derrube a conexão com todos os clientes.

Um desenvolvedor decidiu criar um navegador Web onde cada aba é um processo. Cite pelo menos uma vantagem dessa abordagem em relação a criar as abas com threads.