1) Qual a diferença entre os SOs Fracamente Acoplados e os Fortemente Acoplados?

- Sistemas fortemente acoplados, são sistemas que vários processadores compartilhando a mesma memória física e dispositivos de I/O e são gerenciados por apenas um sistema operacional.

- Sistema fracamente acoplados, são sistemas que possuem dois ou mais sistemas computacionais conectados por uma linha de comunicação, tendo com isso, cada sistema trabalhando independentemente e possuindo seu próprio sistema operacional.

2) Os métodos de E/S podem ser síncronos ou assíncronos. Qual a principal diferença entre eles?

3) No modo Kernel, a CPU executa o SO! Quando ocorre a mudança de modo usuário para modo kernel?

A mudança ocorre quando um processo em modo usuário precisa fazer uma chamada de sistema ou quando recebe um sinal de interrupção (periféricos de I/O).

4) Explique a função das chamadas fork(), wait() e exec()!

Fork - criar um novo processo através da duplicação do processo chamado. O novo processo, chamado *child*, é uma cópia exata do processo chamado, conhecido como o *paren.*

Wait – suspende a execusão do do processo *parent* até que o processo *child* termine (ou mude de estado).

Exec - inicia a execusão de um programa (binário)

5) Quais os elementos de um processo?

Um processo inclui um condador de programa, pilha e seção de dados.

A estrutura de um processo é dividida em 3 partes: contexto de hardware, contexto de software e espaço de endereçamento.

6) Apresente os estados de um processo.

Os estados de um processo são:

- Created/New – o processo esta sendo criado

- Ready – o processo esta esperando para ser atribuído a um processadores

- Waiting – o processo esta esperando que ocorra algum evento

- Running – as instruções estão sendo executadas

- Blocked – o processo esta bloquedo esperando que um evento esteja completo (I/O por exemplo).

- Terminated – o processo terminou sua execução.

7) Threads compartilham arquivos abertos? Por que?

8) Por que threads não compartilham contexto de hardware?

9) Explique o modelo de threads muitos-para-um

Quando duas ou mais threads no nível do usuário estão/são associadas a uma única thread de kernel.

10) O que são threads despachantes e threads operárias?

Dispatcher Thread – thread que envia e recebe mensagens para a thread Worker.

Worker Thread – thread que executa as mensagens recebidas.

11) Qual a diferença entre "exclusão mútua" e "região crítica"?

Exclusão mútua – dois ou mais processos não podem acessar um mesmo recurso simultaneamente. Enquanto um processo estiver acessando determinado recurso, todos os demais processos que queiram acessa-lo deverão esperar pelo término da utilização do processo em execução.

Região crítica – a parte do código do programa onde é feito o acesso ao rescurso compartilhado.