

**1) O que acontece se um processo tentar acessar diretamente um dispositivo de entrada e saída?**

Ele não terá permissão, sendo abortado, é disparada uma interrupção para que o OS encerre o processo. Para a realização adequada de E/S (ou em inglês I/O) é efetuada uma chamada de sistema que gera uma interrupção, está permite o usuário trocar de contexto para o “modo seguro”, tendo privilégios de supervisor (modo Kernel) para acessar periféricos.

**2) Quais são as formas de chavear um processo para o modo supervisor?**

Através de interrupções e chamadas de sistema. São nesses casos que o OS atua. E somente o OS opera em modo supervisor.

**3) Qual o método mais comum de proteção da memória?**

Os bits de proteção. Este é implementado em conjunto pelo hardware e sistema operacional, que disponibilizam para cada processo um espaço de memória exclusivo. O sistema deve impedir que um processo acesse ou modifique uma página do sistema que ele não possua permissão. As permissões são de leitura ou de gravação, sendo que sempre que uma página é referenciada o sistema operacional verifica no mapeamento de processos se a operação é permitida ou não.

**4) Por que as interrupções devem estar ligadas no momento da execução de um aplicativo?**

Por segurança, para o processo não tomar conta do processador deixando de atender novas requisições. As interrupções também impedem acessos a periféricos como o próprio HD, quaisquer permissões privilegiadas precisam antes passar pelas chamadas de sistema para gerar a interrupção e sua respectiva rotina.

**5) Qual a finalidade de um bloco descriptor?**

O bloco descriptor faz exatamente o que o nome diz, cria um registro do processo com todos os seus dados como: tempo de execução, estado, espaço de memória utilizado, localização, arquivos abertos, grupo de processos...

## 6) O que são threads?

São diversas execuções do mesmo processo com objetivo de acelerar o procedimento, pois os processos por segurança não se comunicam diretamente, as threads como pertencem ao mesmo processo sim, desta forma é possível executar “partes” do processo de forma separada em inúmeras tarefas e mesclá-las posteriormente. Assim o sistema como um todo torna-se mais eficiente.

## 7) O que são escalonadores de curto e longo prazo?

**Curto prazo:** Em um pequeno período de tempo o escalonador verifica se o processo em execução já desocupou o processador, colocando outro processo em execução.

**Longo prazo:** O tempo entre as inferências do escalonador são mais esparsas, assim o processador pode ficar ocioso em um determinado período de tempo, entre o término de um processo e a inclusão de um novo processo de execução, realizado pelo escalonador.

**OBS.:** A própria execução do escalonado de curto prazo ocupará a CPU. Assim, nem sempre o escalonar de curto prazo é vantajoso.

## 8) O que são algoritmos de escalonamento preemptivos e não preemptivos?

**Algoritmos Preemptivos:** são mais usuais em sistemas operacionais. Estes permitem que um processo seja interrompido durante sua execução, por exemplo por interrupções. Seu estado armazenado para posteriormente continuado. Isto é importante para evitar o monopólio da CPU por um determinado processo, oferecendo dinamismo para sistemas multiusuários ou com diversos processos concorrentes.

**Algoritmos não preemptivo:** não possui a capacidade de interromper um processo mediante um tempo determinado pelo relógio, assim o processo somente é liberado quando terminar ou fizer uma operação de E/S.