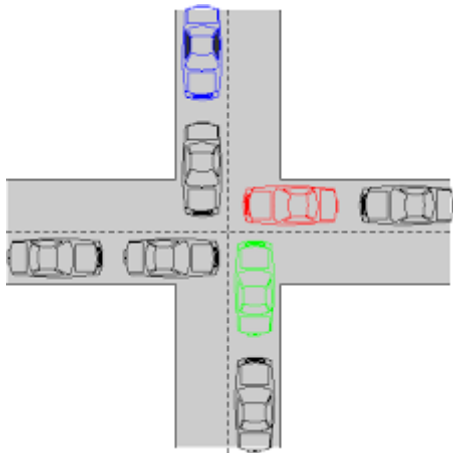


1 – Trancamento (Lockout)

Serve como indicação que um processo está utilizando um recurso que é compartilhado, se o recurso estiver sendo utilizado ele trava o mesmo para que outro processo não acesse o mesmo recurso, evitando dessa forma problemas de leitura e escrita, para se ter certeza do recurso que está sendo acessado não terá um valor diferente do que deveria ser.

2 – Impasse (Deadlock)

Um impasse ocorre quando todo processo pertencente ao conjunto estiver esperando por um evento que somente outro processo desse mesmo conjunto poderá fazer acontecer. Outra forma de se analisar esse problema é imaginar um cruzamento.



Nessa situação para que um carro possa trafegar, outro deve desocupar o caminho e a mesma situação ocorre com os processos, se todos precisam de um recurso e outro processo está usando ele e fica bloqueado isso vai gerar uma cadeia de espera e quando outros processos forem acessar esses recursos o problema aumentaria.

Para solucionar os deadlocks existem 3 técnicas mais famosas, que são: ignorar o problema e esperar que o sistema volte a sua rotina normal, que é implementado na maioria dos sistemas operacionais modernos, detecção do deadlock e recuperação do sistema, o deadlock vai ocorrer e quando for detectado que houve deadlock, o sistema retorna a rotina anterior e reorganiza os recursos para evitar o deadlock, outra solução é o sistema fica verificando a

cada momento regiões críticas e quando perceber que pode haver um deadlock, os recursos já são reorganizados para que o mesmo não ocorra.

3 – Inanição (Starvation)

Starvation ocorre quando um processo que precisa de um recurso para ser executado e por razão de não ter prioridade elevada, e processos com alta prioridade sempre solicitam o recurso dessa forma o processo com menor prioridade nunca terá a chance de acessar o recurso solicitado.

Um exemplo é o caso de fila de impressão, se a regra da fila implementada for o arquivo de menor tamanho será o primeiro da fila, quando houver um arquivo grande, e vários outros arquivos menores entrarem na fila, o arquivo de maior tamanho nunca será impresso.

Uma solução para esse problema é fazer a política de concessão do recurso seja de acordo com a ordem da solicitação do mesmo, fazendo assim que quem chegou primeiro seja o primeiro a executar e não terá seu estado bloqueado por outro processo, menos que seja uma chamada de sistema de interrupção.

4 – Indeterminismo

O funcionamento do escalonador gera uma incerteza em qual momento cada código será executado, fazendo assim que cada processo possa ou não ser executado, dessa forma, se um processo estiver em execução e for retirado da execução pelo escalonador, os valores que foram gerados são arbitrários, pois o processo não pode executar todo o seu corpo e se outro processador for acessar esse recurso, se ocorrerá uma leitura incorreta.

Uma forma de se resolver esse problema é garantir a exclusão mútua que trata do bloqueamento do recurso para evitar problemas de alterações do recurso.