1. É um software ou conjunto de softwares cuja função é administrar e gerenciar os recursos de um sistema, desde componentes de hardware e sistemas de arquivos a programas de terceiros, estabelecendo a interface entre o computador e o usuário.
2. O kernel é responsável por ser o elo do hardware (parte física) com o software (parte lógica) do computador. Em outras palavras, o principal objetivo é gerenciar o computador e permitir que os aplicativos sejam executados e façam uso dos recursos que a máquina tem.
3. Gerenciar recursos como processador, RAM, sistemas de arquivos e dispositivos de entrada e saída (monitor, teclado, mouse, impressora, etc.); Gerenciar processos (execução) de programas; Gerenciar o uso de dispositivos, memória do sistema e chamadas dos programas, definindo quais têm prioridade.
4. Porque está presente no processo dos dispositivos físicos, programas, compiladores, editores e aplicativos.
5. Sistema Operacional OS X, Sistema Operacional Windows, Sistema Operacional Linux.
6. Sistemas de Tempo Compartilhado (time-sharing):

Os sistemas de tempo compartilhado (time-sharing) permitem que diversos programas sejam executados a partir da divisão do tempo do processador em pequenos intervalos, denominados fatia de tempo (time-slice). Caso a fatia de tempo não seja suficiente para a conclusão do programa, ele é interrompido pelo sistema operacional e substituído por um outro, enquanto fica aguardando por uma nova fatia de tempo. O sistema cria para cada usuário um ambiente de trabalho próprio, dando a impressão de que todo o sistema está dedicado exclusivamente a ele.

Sistemas de Tempo Real (real-time)

Os sistemas de tempo real (real-time) são implementados de forma semelhante aos sistemas de tempo compartilhado. O que caracteriza a diferença entre os dois tipos de sistemas é o tempo exigido no processamento das aplicações. Enquanto em sistemas de tempo compartilhado o tempo de processamento pode variar sem comprometer as aplicações em execução, nos sistemas de tempo real os tempos de processamento devem estar dentro de limites rígidos, que devem ser obedecidos, caso contrário, poderão ocorrer problemas irreparáveis.

1. SISD ( Single Instruction Single Data ) - Sistemas que suportam uma única sequência de instruções e apenas uma sequência de dados. Os sistemas com um único processador estão nessa categoria. Estes sistemas simulam paralelismo com a utilização de uma técnica conhecida como pipeline.

SIMD ( Single Instruction Multiple Data ) - Sistemas que com uma única sequência de instruções tratam múltiplas sequências de dados. São sistemas que trabalham com vetores inteiros simultaneamente, possibilitando a execução de uma mesma instrução ou bloco de instruções sobre diferentes elementos de um ou mais vetores.

MISD ( Multiple Instruction Single Data ) - Sistemas que permitem a execução de múltiplas sequências de instruções em uma única sequência de dados. Não existe até o momento nenhum sistema computacional criado com esta arquitetura.

MIMD ( Multiple Instruction Multiple Data ) - Sistema que permite a execução de múltiplas sequências de instruções sobre múltiplas sequências de dados. Esta categoria engloba os sistemas com múltiplos processadores.

1. Aumento da Confiabilidade - Com mais de um processador, caso haja falha de hardware em um processador os demais processadores mantém os sistemas em funcionamento. Desta forma os sistemas multiprocessados aumentam a confiabilidade da infraestrutura de TI e dos serviços e sistemas da empresa.

Escalabilidade - Se antes era necessário trocar o sistema computacional por um outro sistema computacional com mais poder de processamento, e isso custava muito caro, com os sistemas multiprocessados basta adicionar novos processadores conforme a demanda de processamento. Ou seja, o sistema multiprocessado é escalável, ele pode ser expandido conforme a necessidade de poder computacional da organização com a simples adição de novos processadores.

Alta disponibilidade - Pelo mesmo motivo do aumento da confiabilidade, os sistemas multiprocessados oferecem alta disponibilidade dos serviços pois em caso de falha de um ou mais processadores, os outros processadores garantem a disponibilidade do serviço ou seja que o serviço não será interrompido.

Balanceamento de carga - Todo o processamento pesado pode ser distribuído pelos vários processadores disponíveis no sistema, dividindo a carga do processamento e conseguindo melhores resultados em desempenho. Esta abordagem não seria possível em um sistema com um único processador de alto desempenho.

1. Thread é um pequeno programa que trabalha como um subsistema, sendo uma forma de um processo se auto dividir em duas ou mais tarefas. É o termo em inglês para Linha ou Encadeamento de Execução. ... Os threads possuem vantagens e desvantagens ao dividir um programa em vários processos.
2. Programa é uma sequência de instruções codificadas (escritas) a serem executadas pelo computador. Thread é uma sequência de instruções sendo executadas.