**PERGUNTA 1**

1. Durante os estudos do capítulo 1 nós podemos discutir vários aspectos sobre os sistemas operacionais, tal discussão incluiu a conceituação e evolução dos sistemas operacionais, o gerenciamento de recursos o gerenciamento de processos e por fim estudamos a interação entre todos esses componentes.

Em relação a essa conceituação, explore como ocorre a interação entre esses componentes e ao mesmo tempo explique qual seja a responsabilidade de cada um desses componentes no funcionamento do microcomputador.  Produza sua resposta de maneira que ela aborte todos os principais conceitos envolvidos, sempre mantendo o foco e deixar claro como o microcomputador é controlado pelo sistema operacional durante o período em que o mesmo é utilizado pelo usuário.

Antes de tudo é preciso entender que os Sistemas Operacionais podem ser divididos para análise de algumas formas como a presença de Interface Gráfica ou apenas Linhas de Comando, se é um Software Livre ou Proprietário e quanto ao Gerenciamento de memória Física apenas como o CP/M e o MSDOS ou Gerenciamento de memória Física e Virtual como fazem os SOs atuais. A virtualização é a capacidade de um SO gerenciar mais memória do que a máquina possui fisicamente!

Quanto a estruturação do sistema temos o seguinte:

- Núcleo ou Kernel: é responsável por funções primordiais do sistema

- Drivers: fazem a comunicação com dispositivos periféricos

- Programas utilitários: são as ferramentas do sistema

- Códigos de carregamento do sistema: são reponsáveis pela inicialização

Os sistemas operativos ainda podem ser divididos em módulos. O primeiro deles é o **gerenciador de processos** que gerencia o tempo de cada processo da CPU, (um processo é um aplicativo sendo executado, conhecido como job). Cada processo pe representado por um bloco de controle, que consiste em um conjunto de metadados, normalmente referenciado por PCD (process control block), conteno informações sobre o número do processo que o identifica, o estado do processo, os registradores da CPU, dados de prioridade, status de entradas/saidas e o contador que aponda o endereço de memória que contém a proxima instrução. Os estados de um processo podem ser: Novo, Pronto, Esperando, Executando e Terminado. O gerenciador de processos ainda  pode definir arbitrariamente a prioridade de um processo. O quantum de um processo pode ser alterado pelo usuário, definindo novas prioridades e resultado na execução mais rápida de um processo. Usando uma linguagem de programação a comunicação entre aplicativos que ocorre atravez da API (application programing interface). O **﻿gerenciador de memória**﻿ é responsável pelo controle de memória física e virtual do sistema. O **﻿gerenciador de dispositivos**﻿ fornece as condições para dispositivos serem usados pelos processos. O **﻿gerenciador de arquivos**﻿ lida com a organização, criação, leitura e permissões de arquivos.

O **Gerenciamento de recursos** de hardware pode ser resumido a placa gráfica que solicita recursos de um monitor. Já os recursos de software gerenciados pelo SO tratam a solicitação de recursos da placa gráfica; A capacidade de uso da GUI com o motor gráfico com papel de manipulação de interfaces com criação de espaços de interação – janelas com objetos de interação dentro delas.

O **DEADLOCK**ocorre quando um processo A depende da liberação de um recurso que o processo B detem controle e este depende de um recurso sob controle do processo A, travando assim o sistema. A escolha de qual processo deve ir para o status de executando requer a estratégia de escalonamento ou política de escalonamento. Sendo elas:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| FCFS ou FIFO | SJF não preemptivo | SJF  preemptivo | Prioridade preemptivo | Round Robin |
| First come, first served ou first in, first out – Algoritmo não preemptivo. | Shortest job first – e executa o processo até seu termino | Shortest job first – e executa até terminar seu tempo | Conforme a prioridade dos processos obedecendo seu quantum | FIFO preemptivo – escalonamento por revezamento |

Para proteção do Sistema Operacional é separado um espaço de memória para o núcleo e um para os aplicativos de usuário. O design arquitetural é o seguinte:

|  |  |
| --- | --- |
| Núcleo de construção monolitica | Núcleo de contrução em camadas |
| * Todos os módulos estão no mesmo bloco * Cada parte do bloco pode ter finalidades e responsábilidades diferentes * Não existe a necessidade de protocolos de comunicação e politicas de segurança * Velocidade alta por ter acesso direto a regiões de memória diferentes * Programação mais complexa * Dependência entre diversas partes * Uma falha em uma das partes pode degenerar o núcleo rapidamente | * Camadas inferiores promovem serviços a camadas superiores * Quanto mais baixa a camada maior o envolvimento com hardware * Mudanças em uma macada são facilmente isoladas das superiores * Protocolo de comunicação para acesso estável, ordenado e com políticas de segurança para acesso a serviços providos pelas camadas inferiores * Dedicação de diferentes camadas a diferentes grupos de funções |