



Unidade 4

Seção 1

# Sistemas Operacionais

# **Webaula 1**

## **Gerenciamento de memória**

Memória é o local de armazenamento de informações no computador e o seu gerenciamento em um sistema operacional é importante para garantir a eficiência das aplicações que rodam no computador.

Nesta webaula vamos apresentar como é realizado o gerenciamento de memória nos sistemas operacionais e suas características.

Mostrar como se dá a monoprogramação sem troca de processos ou paginação e a multiprogramação com partições fixas.



## Gerenciador de memória

Na maioria dos computadores existe o conceito de hierarquia de memória, que pode ser representado da seguinte maneira:

Uma pequena quantidade de memória cache.

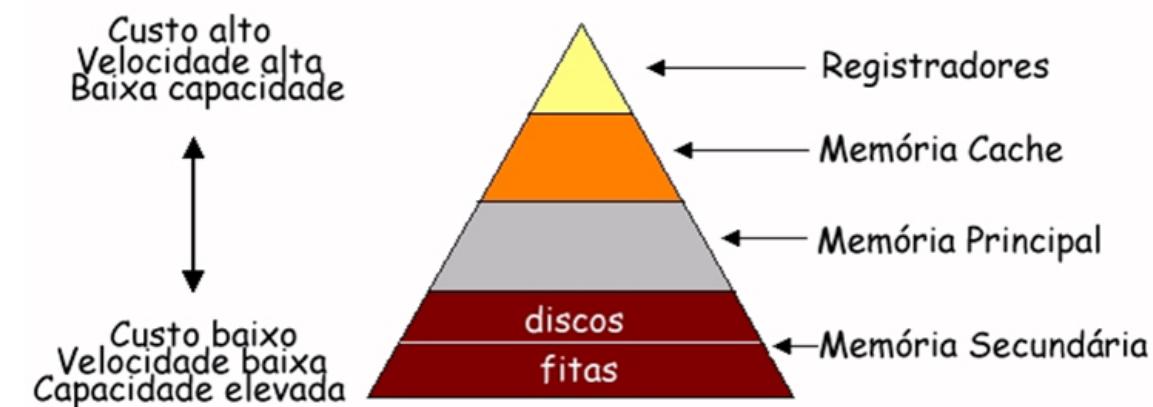
Uma grande quantidade de memória principal (RAM).

Uma memória secundária com as informações armazenadas em disco.





A hierarquia de memória pode ser representada pela pirâmide. Quanto mais alto as memórias estiverem na pirâmide, mais caras serão, possuirão alta velocidade de processamento e baixa capacidade de armazenamento. E quanto mais baixo, mais baratas serão, possuirão uma grande capacidade de armazenamento e baixa velocidade de processamento.



Fonte: <http://www.di.ufpb.br/raimundo/Hierarquia/Hierarquia.html>. Acesso em: 11 jun. 2018.





A hierarquia de memória é controlada pelo gerenciador de memória, que é responsável por gerenciar quais partes estão em uso e quais não estão alocando-a quando os processos precisarem, liberando-a após o término dos processos e controlando a troca de processos entre a memória e o disco quando a memória principal não é suficiente para manter todos os processos em execução.



O gerenciamento de memória pode ser dividido em duas classes:

- Sistemas que fazem troca de processos e paginação.
- Sistemas que não fazem troca de processos e paginação.

A troca de processos (*swapping*) carrega todo o programa para a memória principal, o executa por um determinado tempo e depois este retorna para o disco.

A paginação divide a memória em partições para a execução das aplicações de forma eficiente.





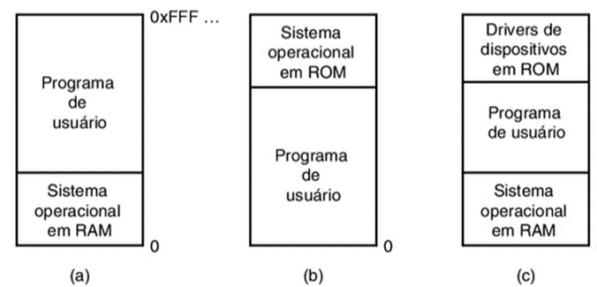
## Monoprogramação sem troca de processos ou paginação

Esse mecanismo de gerenciamento de memória é o mais simples, no qual somente um programa é executado por vez e a memória é compartilhada entre o sistema operacional e o programa.





A monoprogramação sem troca de processos ou paginação pode ocorrer em três formas conforme a figura a seguir. Explore a galeria para ver o detalhamento:



Fonte: Tanenbaum (2003, p. 140).

O sistema operacional está utilizando o espaço de endereçamento em RAM (Parte (a) da figura) – modelo aplicado aos mainframes e minicomputadores.





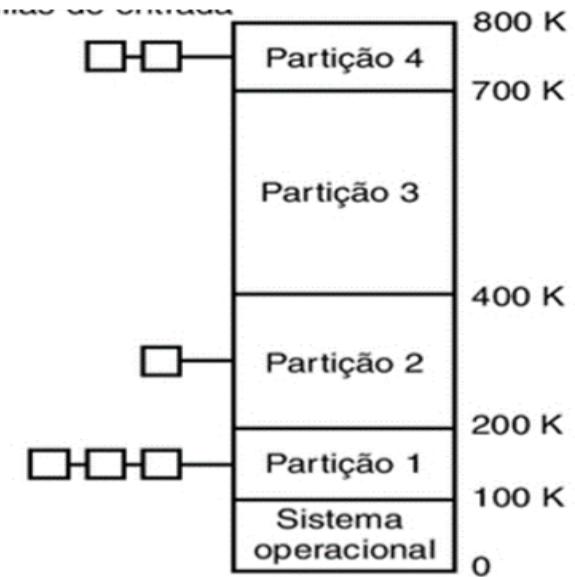
## Multiprogramação com partições fixas

Esse mecanismo de gerenciamento de memória está presente na maioria dos sistemas operacionais modernos. Ele permite que vários processos executem ao mesmo tempo (multiprogramação) e quando um processo é bloqueado aguardando uma informação de entrada/saída, outro processo poderá utilizar a CPU, aumentando a sua utilização.



Neste método, a memória é dividida em  $n$  partições de tamanhos diferentes, podendo ser definida quando o sistema for iniciado. Quando um processo chega para ser executado, ele é inserido em uma fila associada à menor partição suficiente para armazená-lo. Clique na figura.

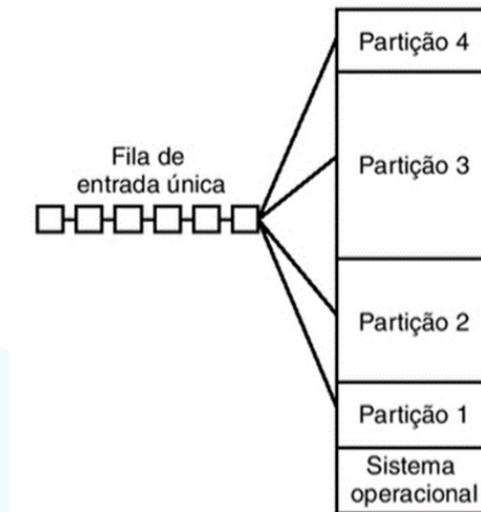
Partições fixas com múltiplas filas de entrada



Fonte: Tanenbaum (2003, p. 141).

Para solucionar o problema de espera na execução de um processo, podemos implementar uma fila única, assim um processo mais próximo do início da fila e que caiba na partição é carregado e executado. A figura a seguir mostra o esquema dessa partição.

#### Partições fixas com filas única de entrada



Fonte: Tanenbaum (2003, p. 141).

Para finalizarmos é importante destacar que nas linguagens de programação, o gerenciamento da memória é fundamental, pois a tendência das aplicações dos usuários é consumir cada vez mais este recurso. Em muitas linguagens de programação não é necessário se preocupar com o gerenciamento, porém é importante que sejam conhecidas as restrições e capacidades do gerenciador de memória para uma programação eficaz.



▼

^

▼



Bons estudos!

