

# NOÇÕES DE INFORMÁTICA

Noções de Sistemas Operacionais



Livro Eletrônico



# SUMÁRIO

Apresentação .....	4
Sistemas Operacionais .....	5
1. Fundamentos de Sistemas Operacionais.....	5
1.1. Funções Básicas de um Sistema Operacional .....	6
1.2. A Evolução dos Sistemas Operacionais.....	8
2. Estrutura do Sistema Operacional.....	17
3. Os Serviços do Sistema Operacional.....	22
3.1. Tipos de Serviços.....	22
3.2. Formas de Interação com o Sistema Operacional.....	24
3.3 Comportamento do Sistema Operacional.....	26
4. Gerenciamento de Processos .....	28
4.1. Processos.....	28
4.2. Ciclos de Processos .....	29
4.3. Estados de Processos .....	30
4.4. Estrutura do Processo .....	32
4.5. Bloco Descritor de Processo .....	34
4.6. Algoritmos de Escalonamento .....	34
4.7. Threads .....	37
4.8. Semáforos.....	39
5. Gerenciamento de Memória .....	40
5.1. Pirâmide das Memórias .....	40
5.2. Memória Virtual .....	40
5.3. Algoritmos para Estratégias de Alocação de Memória.....	43
5.4. Política de Busca de Páginas .....	44
5.5. Reentrância.....	45
6. Sistema de Arquivos.....	45
7. Controle de Acesso .....	47

8. Gerenciamento de Entrada e Saída .....	48
8.1. Classificação para os Dispositivos .....	48
8.2. Buffering .....	49
8.3. Deadlock .....	50
Serviço de Diretório .....	51
Resumo .....	62
Questões Comentadas na Aula .....	66
Questões de Concurso .....	71
Gabarito .....	87
Referências .....	88

## APRESENTAÇÃO

Olá, querido(a) amigo(a)!

A persistência realiza o impossível. Não existem limites para a sua vontade! Força nos estudos!

Rumo então à aula sobre Sistemas Operacionais.

Em caso de dúvidas, acesse o fórum do curso ou entre em contato.

Um forte abraço,

*Profª Patrícia Quintão*

*Instagram: @coachpatriciaquintao*

# SISTEMAS OPERACIONAIS

## 1. FUNDAMENTOS DE SISTEMAS OPERACIONAIS

Um **sistema operacional** é um conjunto de **módulos de software** que tem como principal **função** controlar o funcionamento de um computador, gerenciando a utilização e o compartilhamento dos seus diversos recursos (processador, memória e dispositivos de entrada e saída).

De forma genérica, um **sistema de computação** pode ser dividido em:

- **usuários**: pessoas, equipamentos etc.
- **programas aplicativos**: definem como os recursos de hardware são empregados na solução de um problema (compiladores, jogos, gerenciadores de banco de dados, ...);
- **sistema operacional**: controla e coordena o uso do hardware entre vários programas aplicativos e diferentes usuários;
- **hardware do computador**: recursos básicos (memória, dispositivos de E/S, CPU).

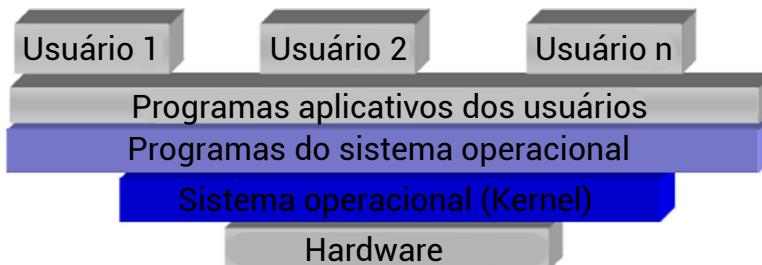


Figura. Sistema de Computação.

### O que se espera de um sistema de computação?

- execução de programas de usuários;
- possibilitar a solução de problemas.

O **sistema operacional (S.O.)** atua como **uma interface entre o usuário de um sistema de computação e o hardware do computador**, simplificando o uso da máquina e otimizando o seu **desempenho global**.

Um S.O. **não** é executado de forma linear, ou seja, suas rotinas são executadas **concorrentemente** em função de eventos assíncronos (eventos que podem ocorrer em momentos que não são pré-estabelecidos).

**Obs.:** Relembrando: **Sistema operacional** é o software que controla a alocação e utilização dos recursos de hardware, como memória, uso da CPU, espaço em disco e periféricos. Além disso, é o **software fundamental do computador**, sobre o qual todos os outros softwares são executados e para o qual os softwares são desenvolvidos. Isso quer dizer que, quando um programador desenvolve um software, ele o faz para um sistema operacional específico.

## 1.1. FUNÇÕES BÁSICAS DE UM SISTEMA OPERACIONAL

As diversas **funções** de um S.O. podem ser resumidas em duas:

- **facilitar a utilização do computador pelo usuário:**
  - esconde detalhes internos;
  - reduz o tempo necessário à construção de programas.
- **gerenciar os recursos do computador** de modo eficiente e protegido:
  - mais trabalho realizado pelo mesmo recurso;
  - melhor distribuição dos recursos;
  - compartilhamento dos recursos de forma protegida.

O S.O. apresenta ao usuário uma **máquina estendida**, equivalente ao hardware, porém muito mais simples de programar e utilizar.

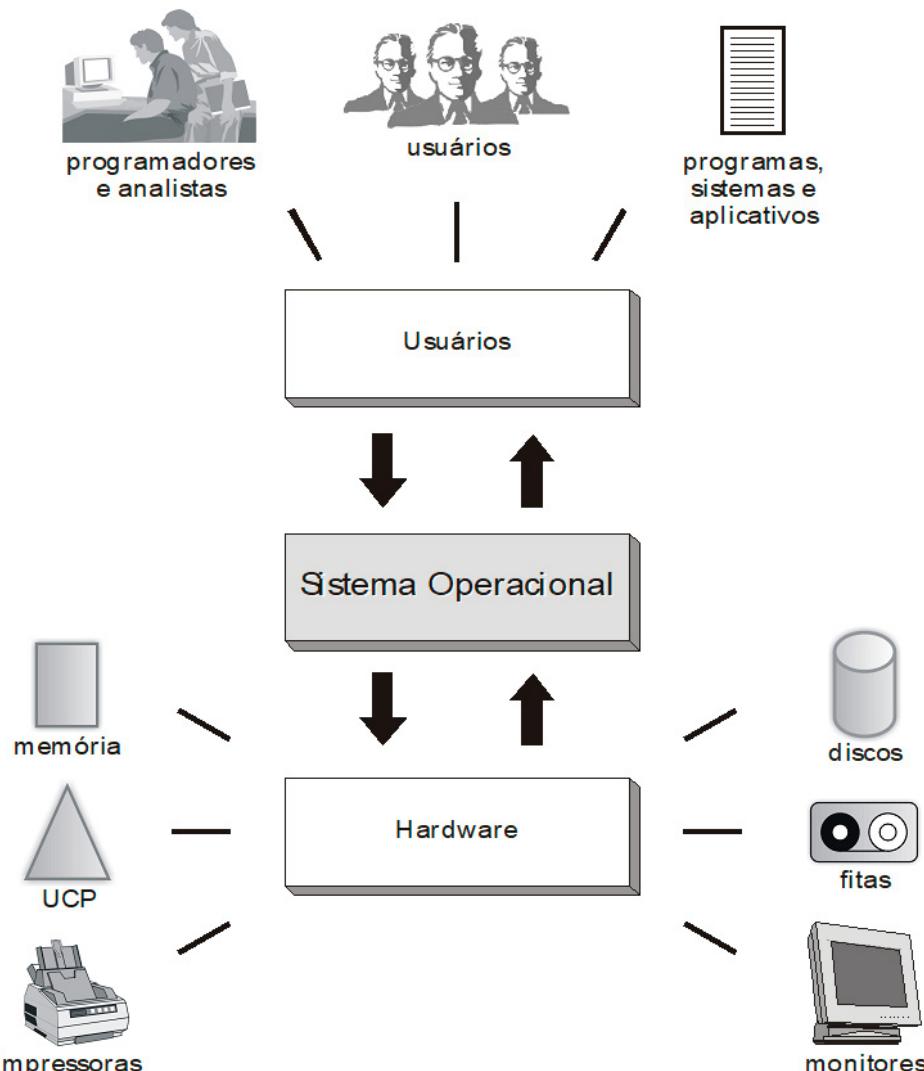


Figura. Visão do Sistema Operacional. Fonte: Machado e Maia (2007)

**O que o sistema operacional realiza em relação aos tópicos listados a seguir?****--- Em relação à gerência de processos?**

- A criação e a exclusão de processos do sistema e do usuário.
- A suspensão e a retomada de processos.
- A provisão de mecanismos para o sincronismo de processo, comunicação entre processos e tratamento de **deadlock**<sup>1</sup>.

**--- Em relação à gerência de memória?**

- Acompanhar quais partes da memória estão atualmente sendo usadas e por quem.
- Decidir quais processos devem ser carregados para a memória quando o espaço na memória se tornar disponível.
- Alocar e desalocar o espaço de memória conforme a necessidade.

**--- Em relação à gerência de arquivos?**

- A criação e a exclusão de arquivos e diretórios.
- O suporte de primitivas para manipulação de arquivos e diretórios.
- O mapeamento de arquivos no armazenamento secundário.
- O backup de arquivos em meio de armazenamento estável (não volátil).

**--- Em relação à gerência do armazenamento secundário?**

- Gerência de espaço livre em dispositivos de armazenamento.
- Alocação de espaço e escalonamento de disco.

**--- Em relação à gerência de subsistemas de entrada/saída (E/S ou I/O) ligados ao computador?**

- Inclui o envio de sinais que informam as ações que o usuário espera que o dispositivo realize, o tratamento das interrupções e erros gerados pelos dispositivos, entre outros.
- Gerência de *buffers*, cache e *spooling*.
- Prover a interface com *drivers* de dispositivos de I/O.

 **DIRETO DO CONCURSO**

**001.** (FCC/TRT- 16<sup>a</sup> REGIÃO (MA)/ANALISTA JUDICIÁRIO/TECNOLOGIA DA INFORMAÇÃO/2014) Um Sistema Operacional (SO) realiza o gerenciamento <sup>.....</sup> que inclui o fornecimento do sistema de arquivos para a representação de arquivos e diretórios e o gerenciamento do espaço em dispositivos com grande capacidade de armazenamento de dados.

<sup>1</sup> Situações de bloqueio, caracterizado por uma espera circular. Exemplo: o processo P<sub>1</sub> está de posse do recurso r<sub>1</sub>, e precisa do recurso r<sub>2</sub>; o processo P<sub>2</sub> está de posse de r<sub>2</sub>, e precisa de r<sub>3</sub>; P<sub>3</sub> está de posse de r<sub>3</sub>, e precisa de r<sub>1</sub> para progredir.

III, que são a unidade básica de trabalho do SO. Isso inclui a sua criação, sua exclusão e o fornecimento de mecanismos para a sua comunicação e sincronização.

II, controlando que partes estão sendo usadas e por quem. Além disso, é responsável pela alocação e liberação dinâmica de seu espaço.

As lacunas I, II e III são, correta e respectivamente, preenchidas por:

- a) de armazenamento - de processos - de memória
- b) em memória secundária - de serviços - em memória principal
- c) de arquivos - de barramentos - de discos
- d) de discos – de threads- de cache
- e) de I/O - de tempos de CPU - de RAM



Vimos que um Sistema Operacional (SO) realiza:

- o **gerenciamento de armazenamento** que inclui o fornecimento do sistema de arquivos para a representação de arquivos e diretórios e o gerenciamento do espaço em dispositivos com grande capacidade de armazenamento de dados;
- o **gerenciamento de processos**, que são a **unidade básica de trabalho do SO**. Isso inclui a sua criação, sua exclusão e o fornecimento de mecanismos para a sua comunicação e sincronização;
- o **gerenciamento de memória**, controlando que partes estão sendo usadas e por quem. Além disso, é responsável pela alocação e liberação dinâmica de seu espaço.

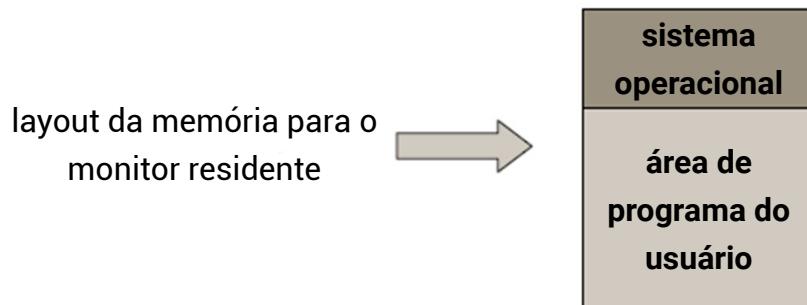
Letra a.

## 1.2. A EVOLUÇÃO DOS SISTEMAS OPERACIONAIS

A evolução dos S.O. está histórica e intimamente ligada às arquiteturas das máquinas nas quais eles rodam. A evolução foi motivada por: melhor utilização dos recursos, avanços tecnológicos (novos tipos de hardware), adição de novos serviços etc.

- **Processamento Serial:**
  - Sistema operacional INEXISTENTE.
  - O usuário é o programador e operador da máquina.
  - Job: programa a ser compilado e executado, acompanhado dos dados de execução (cartões perfurados).
  - Muito tempo gasto para preparar o computador para executar o programa.
  - Alocação do computador através de **reserva antecipada**.
- **Processamento em Lote (Batch):**
  - Introdução de operadores profissionais.
  - Jobs com necessidades semelhantes eram organizados em **lotes**.

- O **monitor residente** desenvolvido em meados de 1950 é considerado o **primeiro sistema operacional (em lote)**.
- O monitor residente fica permanentemente na memória.



- Permite a execução de **APENAS um programa de cada vez**.
- Neste ambiente de execução, a **CPU fica frequentemente ociosa**.
- **Spooling** (*Simultaneous Peripheral Operation On-Line*): com o surgimento da tecnologia de disco, os *jobs* eram gravados nos discos e a leitura dos *jobs* pelo sistema operacional era realizada a partir do disco. O mesmo ocorria para a saída, que era gravada no disco antes de ir para o dispositivo de saída.

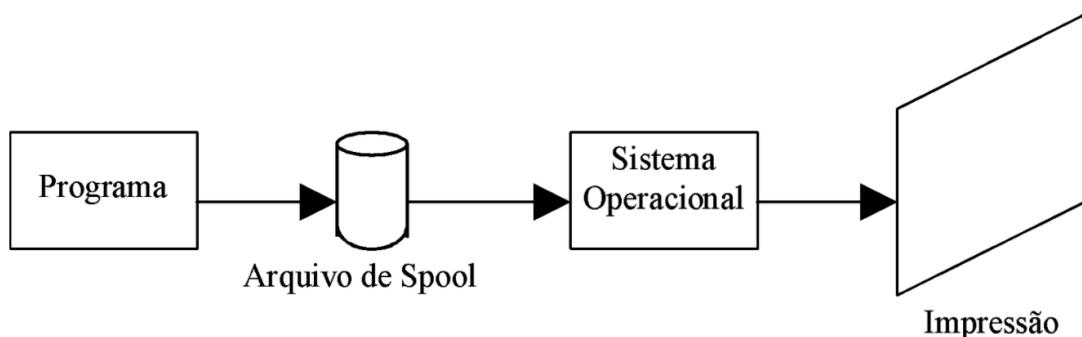
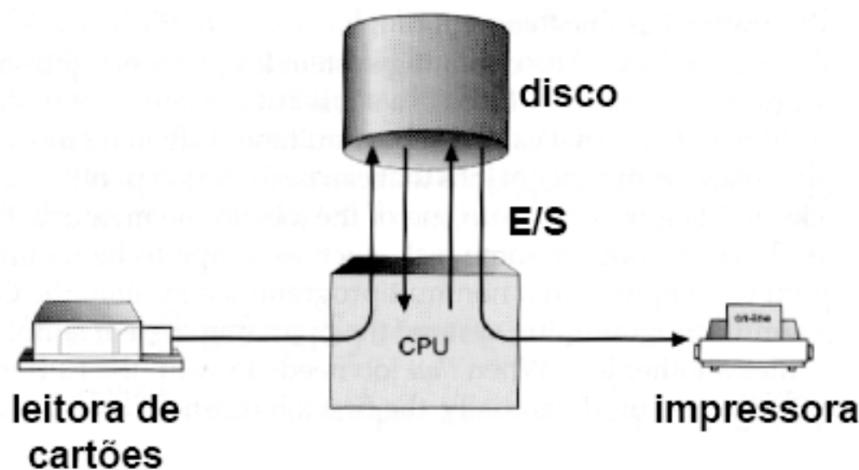
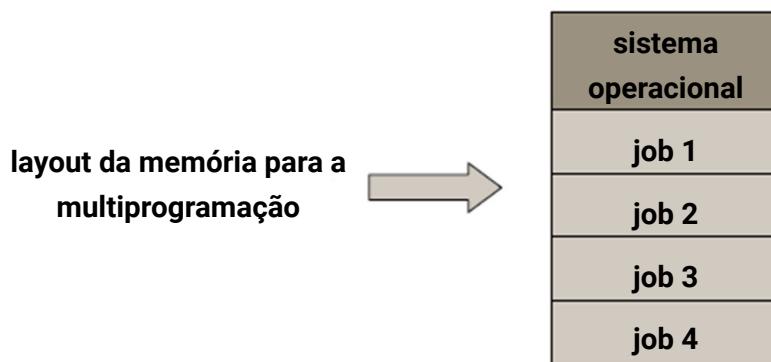


Figura - Técnica de Spooling. Fonte: Zanine (2000)

- **Sistemas Multiprogramados (Multiprogramação):**

- Partiu da ideia de **manter diversos programas na memória ao mesmo tempo e, enquanto um programa realiza E/S, outro pode ser executado.**
- Características necessárias para a multiprogramação:
  - rotinas de E/S fornecidas pelo sistema operacional;
  - a memória principal pode ser alocada aos diversos *jobs* pelo sistema operacional;
- **Escalonamento de jobs (job scheduling):** o sistema operacional escolhe dentre os *jobs* em disco aquele que irá para a memória;
- **Escalonamento de CPU (CPU scheduling):** o sistema operacional escolhe dentre os *jobs* que estão na memória, prontos para serem executados, aquele que realmente será executado.



- **Sistemas de Tempo Compartilhado (Time-Sharing):**

- Time-sharing é uma extensão da multiprogramação.
- O usuário possui um terminal (teclado e monitor), a partir do qual interage com o programa em execução – computação interativa (sistemas on-line).
- A CPU é multiplexada entre vários *jobs* que estão na memória e no disco (a CPU é alocada para um *job* somente se ele estiver na memória).
- Cada usuário tem a ilusão de que a máquina está dedicada exclusivamente à execução do seu programa, enquanto **na realidade um único computador está sendo compartilhado entre vários usuários**.
- A **multiprogramação e o time-sharing são os temas centrais dos sistemas operacionais modernos.**

- **Sistemas de Tempo Real:**

- Empregados para o controle de procedimentos que **devem responder dentro de um certo intervalo de tempo**. Ex.: experimentos científicos, tratamento de imagens médicas, aplicações de controle de tráfego aéreo, de usinas termoelétricas e nucleares, etc.
- A noção de tempo real é dependente da aplicação (milisegundos, minutos, horas, etc.)

- **Sistemas Monotarefa e Multitarefa:**

- **Sistemas monoprogramáveis/ monotarefa:** capazes de executar apenas uma tarefa de cada vez. Ex.: MS-DOS.

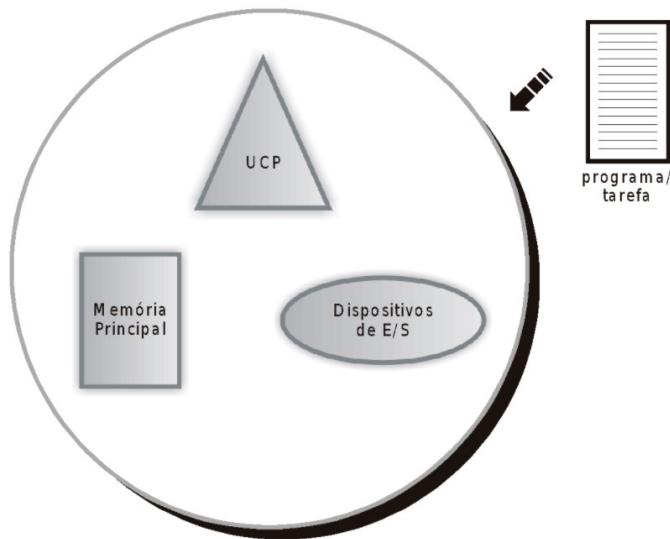


Figura - Sistemas monoprogramáveis/monotarefa. Fonte: Machado e Maia (2007)

- **Sistemas multiprogramáveis/multitarefa:** capazes de executar várias tarefas simultaneamente. Ex.: Microsoft Windows, Unix, Linux. Podem **ser classificados pela forma com que suas aplicações são gerenciadas**, podendo ser divididos em **sistemas batch, de tempo compartilhado ou de tempo real**. Um sistema operacional pode suportar um ou mais desses tipos de processamento (Machado e Maia, 2007).

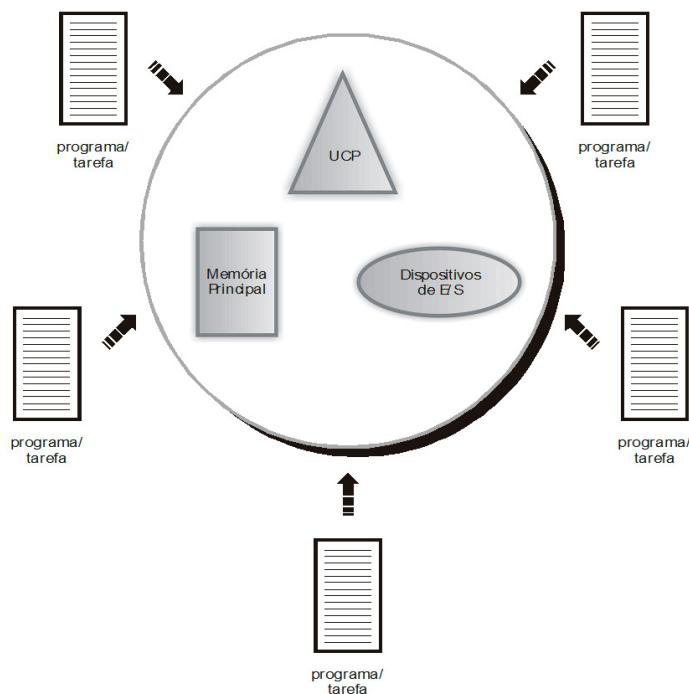


Figura. SOs multiprogramáveis/multitarefa. Fonte: Machado e Maia (2007)

- **Sistemas Monousuário e Multusuário:**

- **Sistemas monousuário:** projetados para serem usados por um único usuário. Ex.: MS-DOS, Windows 9x, Windows Millenium.
- **Sistemas multusuário:** projetados para suportar várias sessões simultâneas de usuários em um computador. Ex.: Unix, Linux, Windows NT, Windows 2000.

	<b>Um usuário</b>	<b>Dois ou mais usuários</b>
<b>Monoprogramação/ Monotarefa</b>	Monousuário	N/A
<b>Multiprogramação/ Multitarefa</b>	Monousuário	Multusuário

Tabela - Sistemas X Usuários. Fonte: Machado e Maia (2007)

- **Sistemas com Múltiplos Processadores (Multiprocessamento):**

- Os sistemas com múltiplos processadores caracterizam-se por possuir duas ou mais UCPs interligadas, trabalhando em conjunto.
- Possibilita a **execução de mais de um programa simultaneamente**.
- Além dos benefícios da multiprogramação, possui outras características como:
  - **escalabilidade**: é a capacidade de ampliar o poder computacional do sistema apenas adicionando novos processadores;
  - **disponibilidade**: é a capacidade de manter o sistema em operação mesmo em casos de falhas. Se um dos processadores falhar, os demais podem assumir suas funções de maneira transparente aos usuários e suas aplicações, embora com menor capacidade de computação; e
  - **balanceamento de carga**: é a possibilidade de distribuir de forma adequada o processamento entre os diversos processadores considerando a carga de trabalho de cada processador.
- Os sistemas com múltiplos processadores podem economizar dinheiro, não duplicando fontes de alimentação, gabinetes e periféricos. Eles podem executar programas mais rapidamente e podem ter maior confiabilidade. Eles também são mais complexos em hardware e software do que sistemas monoprocessadores.
- Um fator-chave no desenvolvimento de sistemas operacionais com múltiplos processadores é a **forma de comunicação entre as UCPs e o grau de compartilhamento da memória e dos dispositivos de entrada e saída**. Em função desses fatores, podemos classificar os sistemas em **fortemente acoplados** ou **fracamente acoplados**.

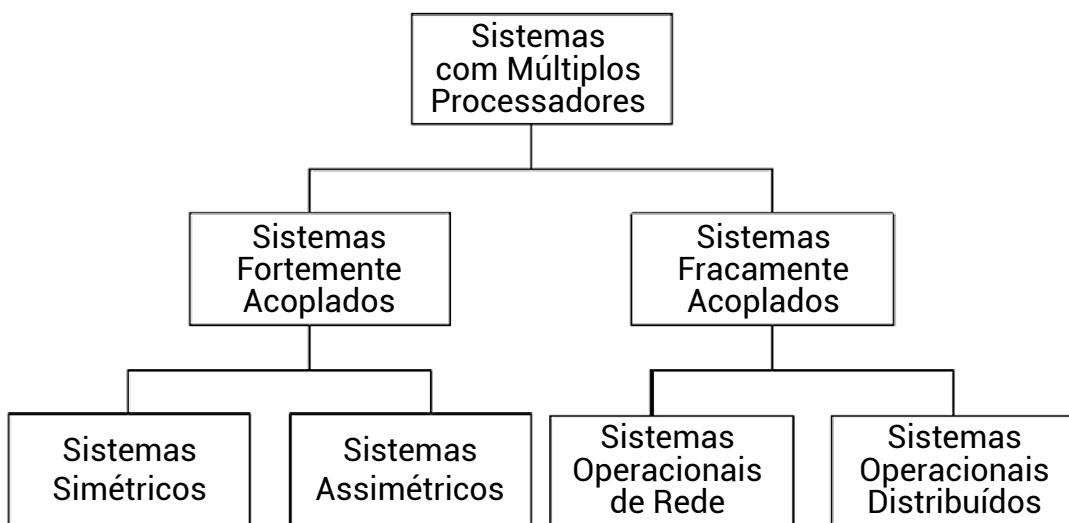


Figura - Sistemas com Múltiplos Processadores. Fonte: Zanine (2000)

- **Sistemas fortemente acoplados (*tightly coupled*)**: vários processadores compartilham uma única memória física e dispositivos de E/S. Ex.: Sistemas Paralelos.

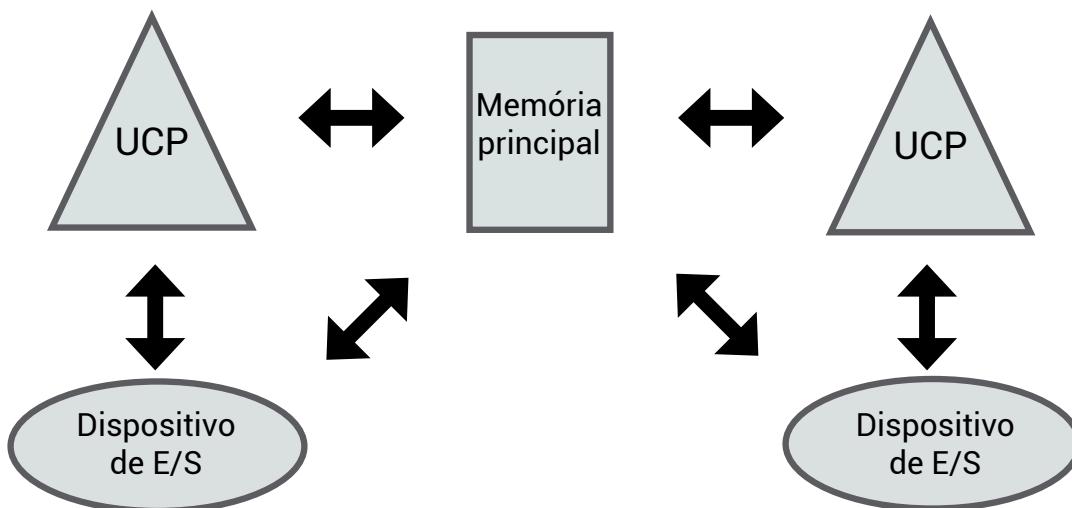


Figura. Sistemas Fortemente Acoplados

### a) Sistemas Assimétricos

Na organização assimétrica ou mestre/escravo (master/slave), somente um processador (mestre) pode executar serviços do sistema operacional, como, por exemplo, realizar operações de entrada/saída. Sempre que um processador do tipo escravo precisar realizar uma operação de entrada/saída, terá de requisitar o serviço ao processador mestre. Dependendo do volume de operações de entrada/saída destinadas aos processadores escravos, o sistema pode se tornar ineficiente, devido ao elevado número de interrupções que deverão ser tratadas pelo mestre.

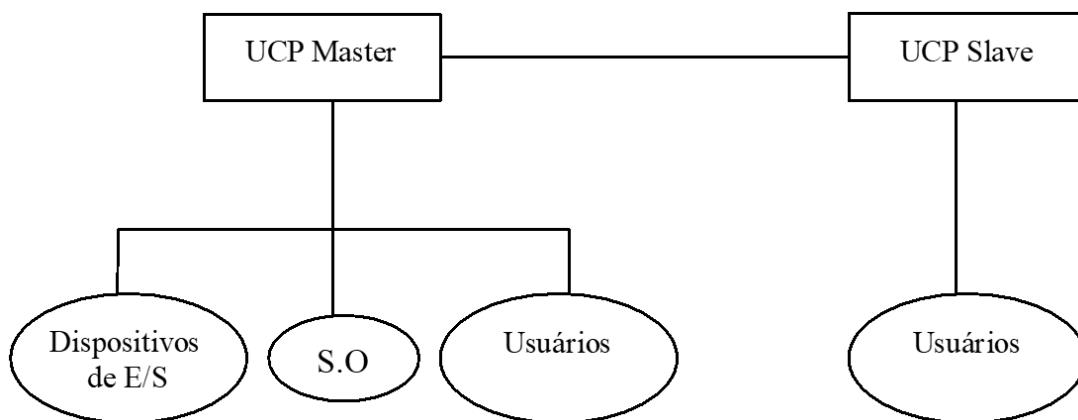


Figura - Sistemas Assimétricos. Fonte: Zanine (2000)

Se o processador falhar, todo o sistema ficará incapaz de continuar o processamento. Neste caso, o sistema deve ser reconfigurado, fazendo um dos processadores escravos assumir o papel do mestre.

Mesmo sendo uma organização simples de implementar e quase uma extensão dos sistemas multiprogramáveis, esse tipo de sistema não utiliza eficientemente o hardware, devido à assimetria dos processadores, que não realizam as mesmas funções.

### b) Sistemas Simétricos

O multiprocessamento simétrico (*Simmetric Multiprocessing- SMP*), ao contrário da organização mestre/escravo, **implementa a simetria dos processadores, ou seja, todos os processadores realizam as mesmas funções**. Apenas algumas poucas funções ficam a cargo de um único processador, como, por exemplo, a inicialização (boot) do sistema.

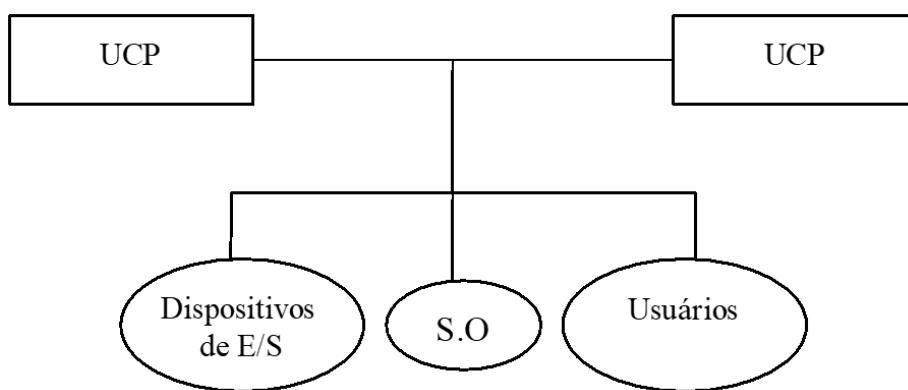


Figura - Sistemas Simétricos. Fonte: Zanine (2000)

Como vários processadores estão utilizando, independentemente, a mesma memória e o mesmo sistema operacional, **é natural a ocorrência de acessos simultâneos às mesmas áreas de memória**. A solução desses conflitos fica a cargo do hardware e do sistema operacional.

No processamento simétrico, um programa pode ser executado por qualquer processador, inclusive por vários processadores ao mesmo tempo (paralelismo). Além disso, quando um processador falha, o sistema continua em funcionamento sem nenhuma interferência manual, porém com menor capacidade de computação.

Os sistemas simétricos são mais poderosos que os assimétricos, permitindo um melhor balanceamento do processamento e das operações de entrada/saída, apesar de sua implementação ser bastante complexa.

- **Sistemas fracamente acoplados (*loosely coupled*)**: dois ou mais sistemas de computação conectados através de linhas de comunicação.

Exs.: Sistemas operacionais de rede (SOR); sistemas distribuídos (*clusters* e computação em grade – *grid computing*).

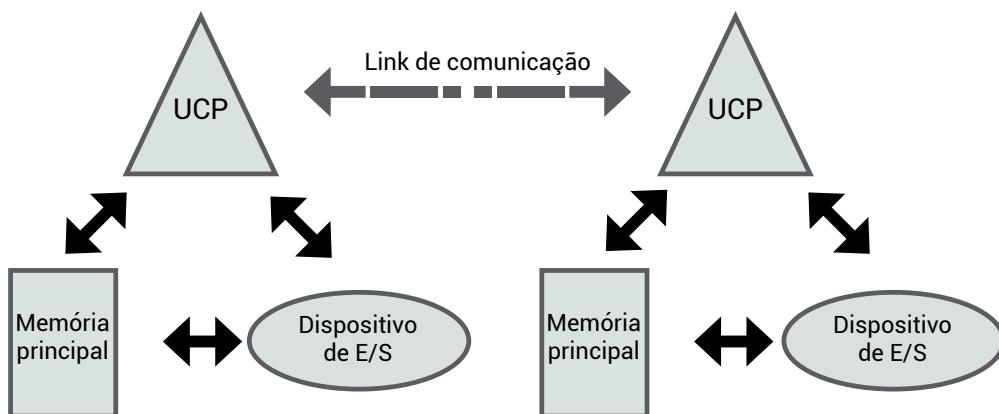


Figura. Sistemas Fracamente Acoplados

### a) Sistemas Operacionais de Rede (SOR)

Em SOR, cada nó possui seu próprio sistema operacional, além de um hardware e software que possibilitam ao sistema ter acesso a outros componentes da rede, compartilhando seus recursos.

O SOR possibilita entre outras funções:

- cópia remota de arquivos;
- compartilhamento de arquivos;
- emulação de terminal;
- impressão remota;
- correio eletrônico, etc.

Cada nó é totalmente independente do outro, podendo inclusive possuir sistemas operacionais diferentes. Caso a conexão entre os nós sofra qualquer problema, os sistemas podem continuar operando normalmente, apesar de alguns recursos se tornarem indisponíveis.

O melhor exemplo da utilização dos SOR são as **redes locais**. Nesse ambiente, cada estação pode compartilhar seus recursos com o restante da rede. Caso uma estação sofra qualquer, os demais componentes da rede podem continuar o processamento, apenas não dispondo dos recursos oferecidos por ela.

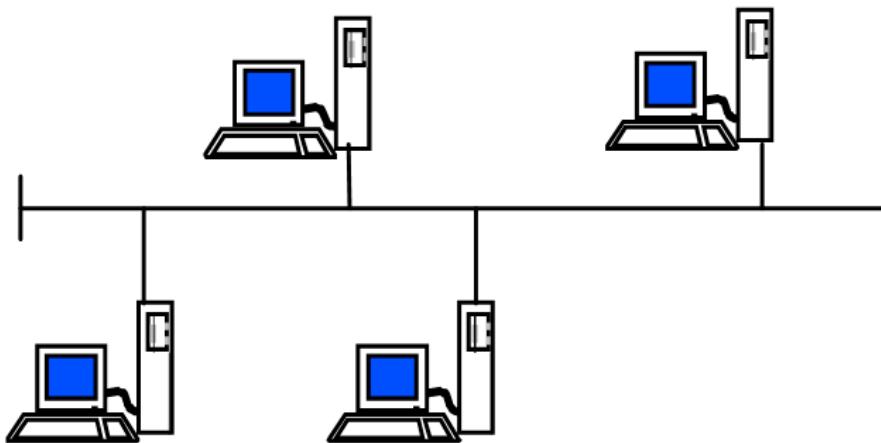


Figura - Sistemas Operacionais de rede. Fonte: Zanine (2000)

#### d) Sistemas Distribuídos

- Esse sistema **distribui a computação entre vários processadores físicos**.
- Em sistemas distribuídos, **cada componente da rede também possui seu próprio sistema operacional, memória, processador e dispositivos**. O que define um sistema distribuído é **a existência de um relacionamento mais forte entre os seus componentes, em que geralmente os sistemas operacionais são os mesmos** (Zanine, 2000).
- Os processadores não compartilham memória ou um relógio. Em vez disso, cada processador possui sua própria memória local. Eles se comunicam entre si através de diversas linhas de comunicação, como um barramento de alta velocidade ou uma rede local.

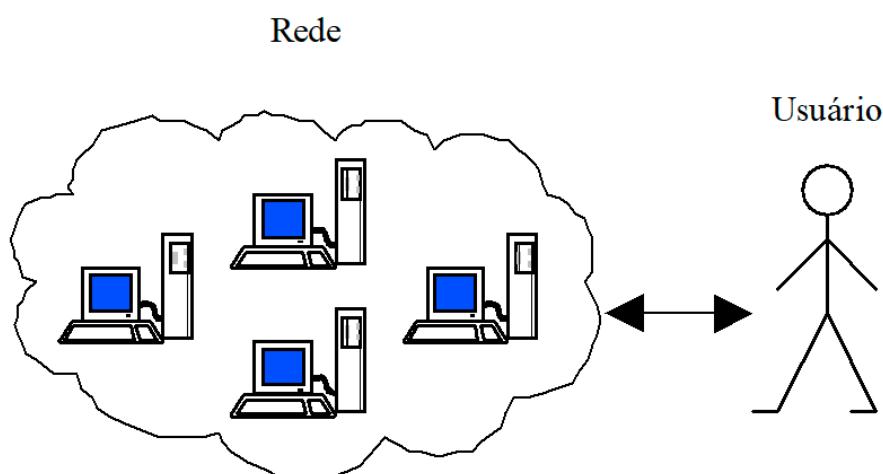


Figura - Sistemas Operacionais Distribuídos. Fonte: Zanine (2000)

- Os sistemas distribuídos **permitem, por exemplo, que uma aplicação seja dividida em partes e que cada parte seja executada por hosts diferentes da rede de computadores.** Para o usuário e suas aplicações, é como se não existisse uma rede de computadores, mas sim um único sistema centralizado (Machado e Maia, 2007).
  - **Outro exemplo de sistemas distribuídos são os clusters.** Conforme citam Machado e Maia (2007), em um *cluster* existem dois ou mais servidores ligados, normalmente, por algum tipo de conexão de alto desempenho. O usuário não conhece os nomes dos membros do *cluster* e não sabe informar quantos são. Quando ele necessita de algum serviço, basta solicitar ao *cluster* para obtê-lo. Sistemas em *cluster*, atualmente, são utilizados para serviços de banco de dados e Web, garantindo alta disponibilidade, escalabilidade e balanceamento de carga à solução.
- 
- **Sistema Embarcado / Incubado / Embutido:**
    - **Inseridos em hardware com recursos limitados.**
      - Portáteis;
      - Tarefas predefinidas;
      - Requisitos específicos.

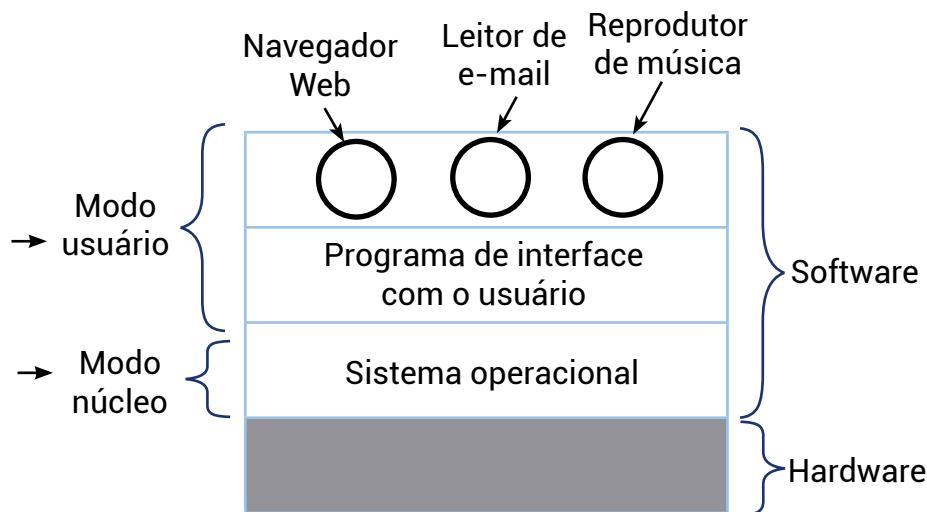
## 2. ESTRUTURA DO SISTEMA OPERACIONAL

O **kernel** (núcleo) do sistema operacional é um conjunto de rotinas que oferece serviços aos usuários, às suas aplicações e também ao próprio sistema.

As rotinas do **kernel** não são executadas de forma sequencial, mas sim **de forma concorrente e sem uma ordem específica**, com base em eventos assíncronos (dissociados do tempo) relacionados ao *hardware* e a tarefas internas do próprio sistema operacional.

Ao **kernel** do S.O. é permitida a execução de **instruções privilegiadas da CPU** (que controlam aspectos do *hardware* como, por exemplo, as interrupções do temporizador). Para que isto seja possível, o **kernel** é executado no **modo supervisor** (também chamado de **modo kernel**).

Com o objetivo de proteger o *hardware* da ação direta do usuário final da máquina, **os demais programas e aplicativos rodam no modo usuário** (com permissão de executar apenas **instruções não privilegiadas**).



## DIRETO DO CONCURSO

**002.** (CESPE/STF-ANALISTA JUDICIÁRIO DE TI-2013) No que se refere aos sistemas operacionais, julgue os itens subsecutivos.

No modo de operação do processador denominado modo usuário, instruções privilegiadas não podem ser executadas. Se houver tentativa de execução nesse caso, o hardware automaticamente gerará a interrupção e acionará o sistema operacional.



**Instruções privilegiadas** são **instruções que só devem ser executadas pelo sistema operacional ou sob sua supervisão**, impedindo, assim, a ocorrência de problemas de segurança e integridade do sistema. As instruções não privilegiadas não oferecem risco ao sistema.

**Quando o processador trabalha no MODO KERNEL OU SUPERVISOR a aplicação pode ter acesso ao conjunto total de instruções do processador, enquanto no MODO USUÁRIO, uma aplicação só pode executar instruções não privilegiadas**, tendo acesso a um número reduzido de instruções. Se houver tentativa de execução de instruções privilegiadas no modo usuário, o hardware automaticamente gerará a **interrupção** e acionará o sistema operacional.

**Certo.**

A estrutura do sistema operacional (organização do código do sistema e inter-relacionamento entre seus componentes) pode variar, conforme a concepção do projeto:

- **Estrutura Simples (Única) ou Monolítica:** todas as funções do S.O. estão combinadas em uma única entidade (espaço de endereçamento e domínio de proteção). Simplesmente **não** há estruturação visível na organização monolítica.

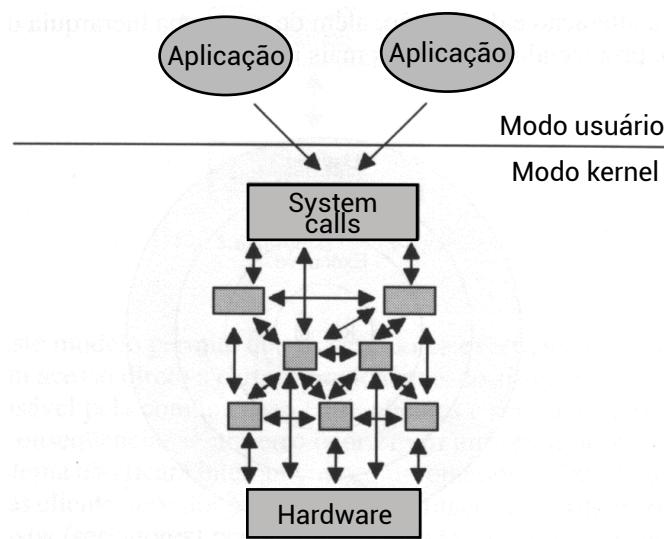


Figura - Sistemas Monolíticos. Fonte: Zanine (2000)

- **Estrutura em Camadas:** forma de modularização do S.O., em que este é dividido em uma série de camadas (níveis), cada qual construída sobre camadas inferiores.

A camada inferior (camada 0) é o hardware e a camada superior (camada N) é a interface de usuário.

Um sistema em camadas divide o sistema operacional em camadas sobrepostas. Cada módulo oferece um conjunto de funções que podem ser utilizadas por outros módulos. Módulos de uma camada podem fazer referência apenas a módulos das camadas inferiores. Ex. de sistema em camadas: Sistema Multics.

5	Operador
4	Programas de Usuário
3	Entrada/Saída
2	Comunicação
1	Gerência de memória
0	Multiprogramação

Figura - Sistema MULTICS. Fonte: Zanine (2000)

O **sistema em camadas** é mais fácil de depurar e modificar, pois as mudanças afetam apenas seções limitadas do sistema, em vez de tocar em todas as seções do sistema operacional. As informações são mantidas apenas onde são necessárias e são acessíveis apenas dentro de uma área definida e restrita, de modo que quaisquer *bugs* que afetam esses dados precisam ser limitados a um módulo ou camada específica (Zanine, 2000).

- **Microkernel:** mantém somente funcionalidades básicas no kernel. Todas as outras funcionalidades são mantidas em módulos que se comunicam através da passagem de mensagens (sem espaço de endereçamento compartilhado ou domínio de proteção).

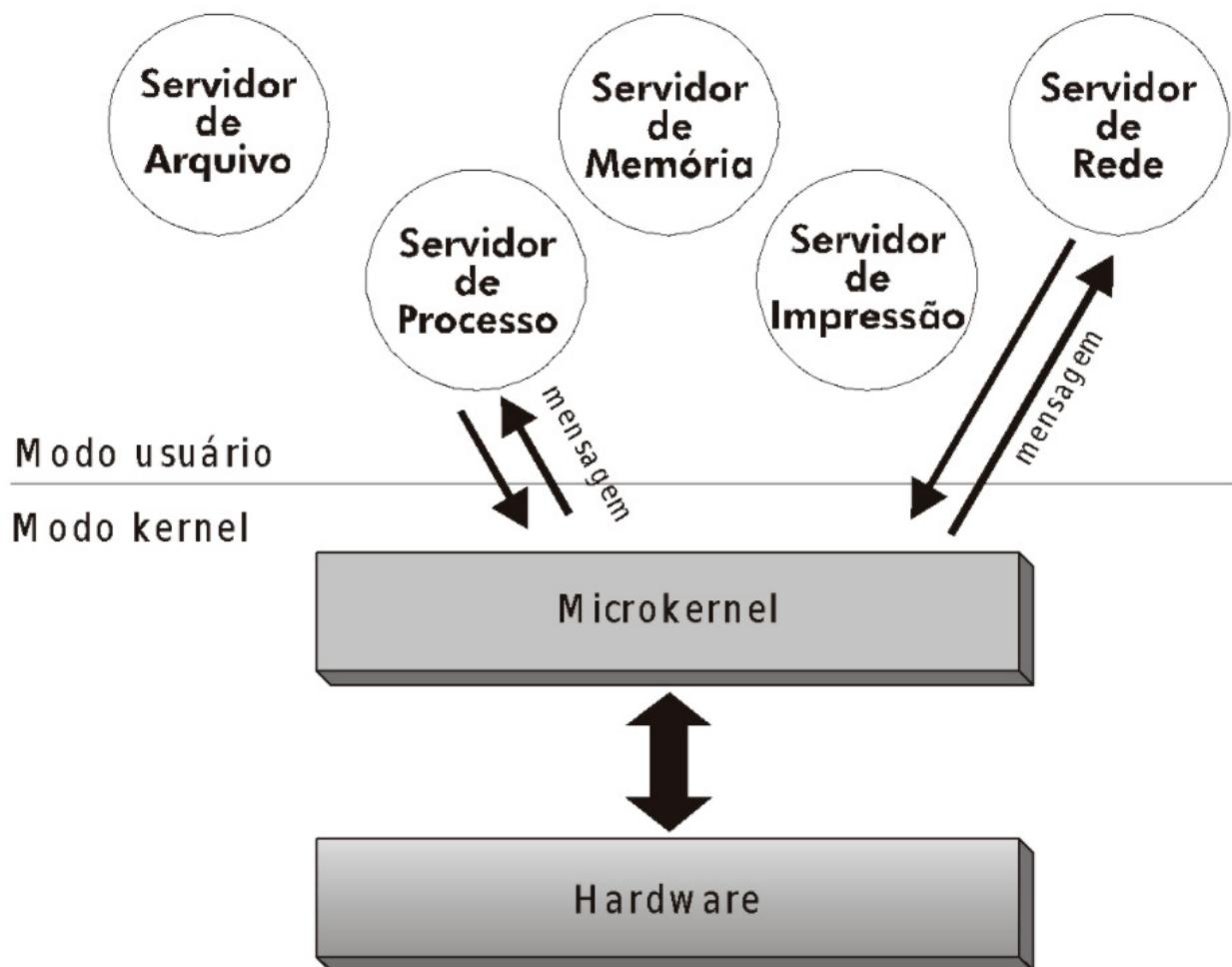


Figura. Arquitetura MicroKernel. Fonte: Machado e Maia (2007)

Os **benefícios** normalmente incluem o seguinte:

- (a) acrescentar um novo serviço não exige modificação do kernel;
- (b) é mais seguro, à medida que mais operações são feitas no modo usuário do que no modo kernel; e

(c) projeto e funcionalidade de *kernel* mais simples normalmente resultam em um sistema operacional mais confiável.

- **Máquina Virtual:** fornece a cada usuário a ilusão de que ele possui a sua própria máquina (em sistemas multiusuários). O sistema é mais fácil de depurar e os problemas de segurança são fáceis de solucionar.

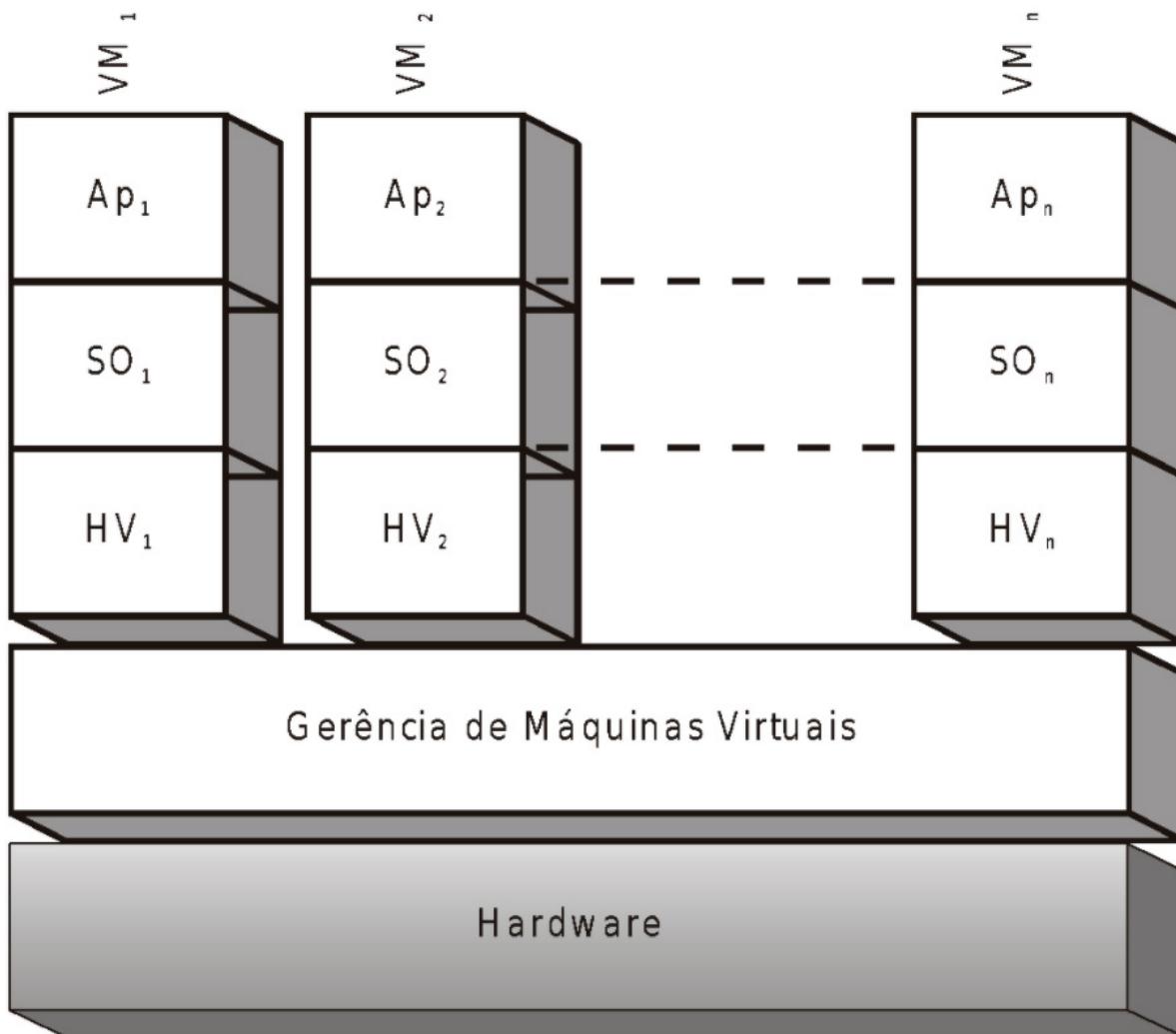


Figura. Máquina Virtual. Fonte: Machado e Maia (2007)

- **Modelo Cliente-Servidor:** o sistema é dividido em **processos**, sendo cada um responsável por oferecer um conjunto de serviços, como serviços de arquivos, serviços de criação de processos, serviços de memória, serviços de escalonamento, etc. **Sempre que uma aplicação deseja algum serviço, ela solicita ao processo responsável.** Neste caso, a aplicação que solicita um serviço é chamada de **cliente**, enquanto o processo que responde à solicitação é chamado **servidor**.

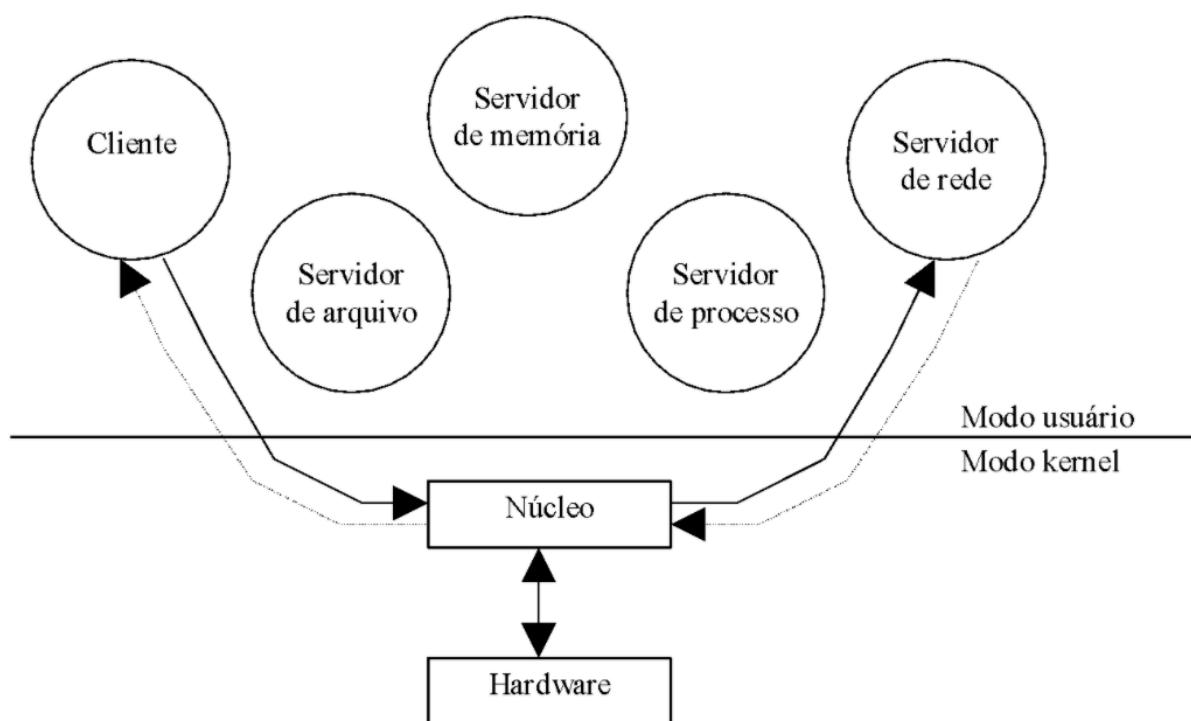


Figura -Sistemas Cliente-servidor. Fonte: Zanine (2000)

### 3. Os SERVIÇOS DO SISTEMA OPERACIONAL

O sistema operacional também pode ser considerado **um administrador e fornecedor de recursos (incluindo serviços)**. Ele cuida de todos os recursos que estão disponíveis no computador, permitindo ao usuário utilizar a máquina (hardware + S.O.) de maneira amigável. Isso é fundamental em qualquer máquina, tornando-se mais crítico naquelas que permitem a utilização simultânea por mais de um usuário.

#### 3.1. TIPOS DE SERVIÇOS

O sistema operacional (SO) fornece um **ambiente para a execução de programas** através da disponibilização de serviços para os programas e para os usuários desses programas.

Alguns serviços não têm como preocupação apenas tornar a máquina mais confortável para o usuário, mas também, para que o próprio sistema seja mais eficiente e seguro.

Apesar da forma como esses serviços são oferecidos podem variar de sistema para sistema, existem **algumas classes de serviços que são comuns a todos os sistemas operacionais**. São elas:

- **Execução de programa:** o SO carrega o conteúdo (ou seções) de um arquivo para a memória e inicia sua execução. Um programa em nível de usuário poderia não ser confiável para alocar devidamente o tempo de CPU.

- **Operações de E/S.** Discos, fitas, linhas seriais e outros dispositivos precisam ter comunicação em um nível muito baixo. O usuário só precisa especificar o dispositivo e a operação que atuará sobre ele, enquanto o sistema converte essa requisição em comandos específicos do dispositivo ou controlador. Não se pode confiar que os programas no nível do usuário acessarão apenas os dispositivos aos quais devem ter acesso e que somente os acessará quando não estiverem sendo usados.
- **Manipulação do sistema de arquivos.** Existem muitos detalhes na criação, exclusão, alocação e nomeação de arquivos que os usuários não terão de realizar. Os programas do usuário não poderiam garantir a aderência aos métodos de proteção dos arquivos, nem se pode confiar neles para alocar apenas blocos livres e desalocar blocos na exclusão do arquivo.
- **Comunicações.** A troca de mensagens entre os sistemas requer que as mensagens sejam transformadas em pacotes de informação, enviadas ao controlador de rede, transmitidas por um meio de comunicação e remontadas pelo sistema de destino. É preciso que haja ordem nos pacotes e correção de dados. Os programas do usuário não poderiam coordenar o acesso ao dispositivo da rede, ou então poderiam receber pacotes destinados a outros processos.
- **Detecção de erros nos níveis de Hardware e de Software.**
- **No nível de Hardware,** todas as transferências de dados precisam ser inspecionadas para garantir que os dados não tenham sido adulterados em trânsito. Todos os dados na mídia precisam ser verificados para garantir que não sejam alterados desde que foram gravados na mídia.
- **No nível de Software,** a mídia precisa ser verificada em busca da consistência dos dados; por exemplo, se a quantidade de blocos de armazenamento alocados e não alocados combina com a quantidade total no dispositivo. Os erros são constantemente independentes do processo (por exemplo, a adulteração dos dados em um disco), de modo que é preciso haver um programa global (SO) que trate de todos os tipos de erros. Além disso, processando os erros pelo SO, os processos não precisam conter código para apanhar e corrigir todos os erros possíveis em um sistema.
- **Alocação de recursos:** o SO é o responsável pela **alocação dos diversos recursos em sistemas com um ou mais usuários**. Estes recursos incluem a memória principal, a própria CPU, arquivos e os dispositivos de E/S. A alocação deve ser feita da forma mais eficiente possível para não prejudicar o desempenho do sistema.
- **Proteção:** o SO é o responsável pela **proteção a todo o sistema computacional**. Essa proteção se torna necessária tanto em sistemas monousuários quanto em sistemas multiusuários. A única diferença é a sua complexidade. Quando vários usuários estão usando o sistema a execução de um programa não pode interferir na execução de outro. Além disso, o próprio sistema operacional deve ser protegido de erros cometidos pelos usuários. A segurança no acesso ao sistema computacional é fundamental.

## 3.2. FORMAS DE INTERAÇÃO COM O SISTEMA OPERACIONAL

A seguir, são abordadas as duas formas de interação com o sistema operacional: as **chamadas de sistema** e os **programas utilitários** (programas do sistema).

- **Chamadas de Sistema (*system Calls*):**

O nível mais fundamental de serviços fornecido pelo sistema operacional é realizado através de **chamadas de sistema (*system calls*)**.

As chamadas de sistema fornecem uma interface entre um programa em execução e o sistema operacional.

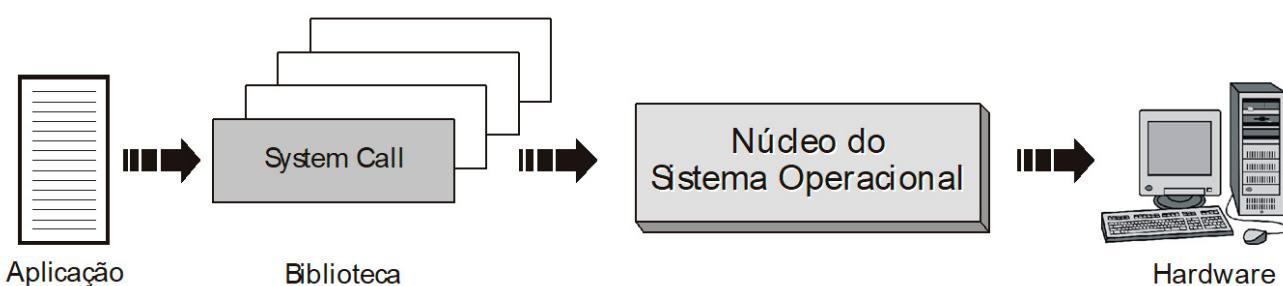


Figura. System Call. Fonte: Machado e Maia (2007)

Estão, geralmente, disponíveis como instruções nas linguagens de baixo nível ou até mesmo em linguagens de alto nível, como C.

As **chamadas de sistema podem ser classificadas em duas categorias principais:**

- **controle de processos:** nessa categoria existem **chamadas ao sistema para a criação e a finalização de processos**, a manipulação de tempo para manter a sincronização entre processos concorrentes, o carregamento e a execução de programas, a obtenção e a ativação de atributos dos processos, entre outras.
- **gerenciamento de arquivos e de dispositivos de E/S:** nessa categoria existem chamadas ao sistema para criar, apagar, abrir e fechar um arquivo, ler e escrever em arquivos, e ainda obter e modificar os atributos de um arquivo, entre outras. Os atributos de um arquivo incluem, por exemplo, o seu nome, tipo, códigos de proteção e tamanho. Em geral, a única informação necessária para o sistema realizar um acesso a um arquivo é o seu nome. No caso de dispositivos de E/S, existem as chamadas para requisitar e liberar um dispositivo, entre outras.

Na utilização de uma chamada ao sistema, normalmente são necessárias mais informações do que simplesmente identificar a chamada desejada.

A quantidade e o tipo exato de informações varia de acordo com a chamada e o sistema operacional em particular.

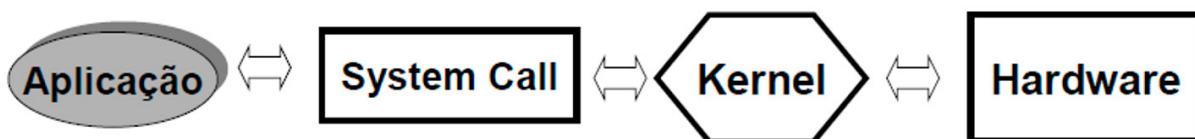
## DIRETO DO CONCURSO

**003.** (CESGRANRIO/PETROBRAS/2012/ANALISTA DE SISTEMAS JÚNIOR – INFRA) O mecanismo pelo qual programas dos usuários solicitam serviços ao núcleo do sistema operacional é denominado

- a) biblioteca do sistema
- b) chamada do sistema
- c) editor de ligação
- d) shell de comandos
- e) ligação dinâmica



Para proteger o núcleo do sistema e como consequência os seus serviços, é implementado um mecanismo chamado de **System Calls**, ou **chamadas do sistema**.



As System Calls podem ser divididas em grupos:

- **Gerência de Processos** (ex.: Kill <pid>);
- **Gerência de Memória** (ex.: msgrcv, processo busca msg na memória);
- **Gerência de Entrada e Saída** (ex.: filecopy);

Letra b.

- **Programas Utilitários (Programas do Sistema – System Programs):**

Outros tipos de serviços oferecidos pelo sistema operacional são os chamados programas utilitários, ou *system programs*.

Esses **serviços são de mais alto nível e fornecem uma interface entre o usuário e o sistema operacional**, tornando o ambiente mais conveniente para o desenvolvimento e execução de programas.

Pode-se agrupar os programas utilitários da seguinte forma:

- **manipulação de arquivos:** apagar, copiar, ver o conteúdo, renomear e imprimir um arquivo e ainda verificar a existência do arquivo, além dos programas que manipulam diretórios;

- **informações sobre o sistema:** obter e modificar data e hora da máquina, conhecer a quantidade de memória secundária disponível, número de usuários e outras informações similares;
- **suporte para linguagens de programação:** junto com o sistema operacional podem ser fornecidos, ou vendidos separadamente, programas como compiladores, montadores, interpretadores para as linguagens de programação mais comuns como Pascal, Fortran ou C;
- **carregamento e execução de programas:** uma vez compilado, montado e livre de erros, um programa deve ser carregado na memória principal e executado. O sistema pode oferecer instrumentos para que essas duas operações sejam efetuadas mais eficientemente (carregadores absolutos ou realocáveis, depuradores, entre outros).

Existe ainda um **programa utilitário** muito importante para o sistema operacional:

- **interpretador de comandos (*SHELL*):** sua função principal é obter e executar o próximo comando especificado pelo usuário. Esse programa começa a executar quando o sistema é iniciado e fica esperando que um usuário digite um comando para interpretá-lo.

Existem **duas alternativas possíveis na implementação do interpretador de comandos:** o interpretador pode conter o código que executa o comando pedido (ativa os parâmetros e invoca as chamadas ao sistema necessárias) ou então cada comando é implementado por um programa independente.

A visão que os usuários têm do sistema operacional é definida mais pelos programas utilitários que pelas chamadas ao sistema, particularmente pelo interpretador de comandos.

A visão que um projetista do sistema operacional tem do sistema como um todo é bastante diferente. Ele vê os recursos físicos (disco, impressora, teclado, vídeo) e deve convertê-los em facilidades lógicas fornecidas aos usuários.

### 3.3 COMPORTAMENTO DO SISTEMA OPERACIONAL

Os sistemas operacionais têm uma característica bastante interessante. Se não existir nenhum programa a executar, nenhum dispositivo de entrada/saída a ser atendido e nenhum usuário aguardando uma resposta, o sistema operacional fica esperando a ocorrência de algum **evento**.

Esses eventos são, em geral, representados por **interrupções**. Quando uma interrupção ocorre, o hardware transfere o controle para o sistema operacional. Neste momento, o sistema operacional salva o estado da máquina (registradores e contador de programa) e determina qual foi o tipo de interrupção que ocorreu, já que existem diferentes tipos de interrupções.

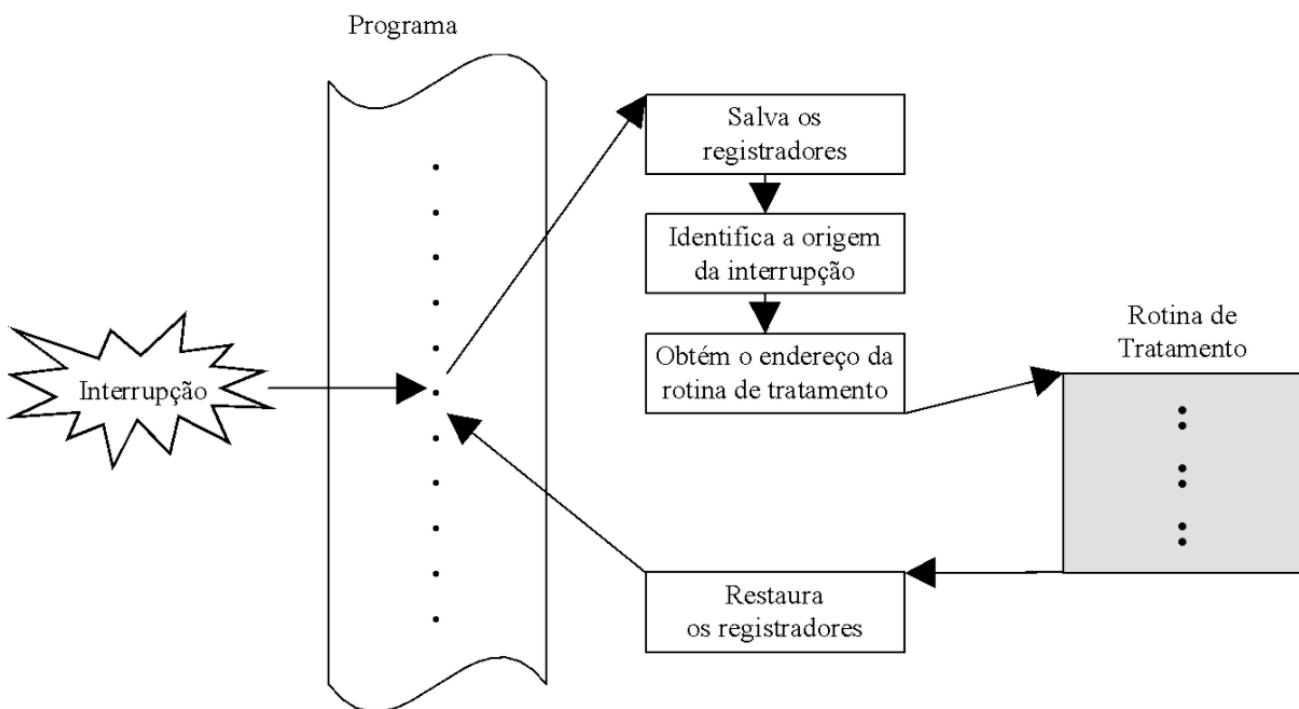


Figura - Mecanismo de Interrupção. Fonte: Zanine (2000)

Para cada tipo de interrupção um tratamento diferente tem que ser realizado.

As **interrupções** são uma forma de melhorar a eficiência do processamento, e **podem ser divididas nas seguintes classes**:

- **Interrupções de Programa:** são geradas por alguma condição que ocorre devido ao resultado da execução de uma instrução, como um *overflow aritmético*, divisão por zero, tentativa de executar uma instrução de máquina inválida ou uma referência fora do espaço de memória liberado para o usuário;
- **Interrupções do Temporizador (Timer):** são geradas por um temporizador dentro do processador. Isto permite ao sistema operacional executar certas funções em períodos regulares;
- **Interrupções de Dispositivos de E/S:** são geradas por um controlador de E/S, para sinalizar a conclusão normal de uma operação ou para sinalizar uma variedade de condições de erro;
- **Interrupções por Falha no Hardware:** são geradas por uma falha como uma falha de energia ou erro de paridade de memória.

A maioria dos eventos que ocorrem pertencem à classe de interrupções dos dispositivos de E/S. Uma operação de E/S é resultante de uma **chamada de sistema** requisitando tal serviço. Uma situação prática seria um programa abrir um arquivo e escrever alguma informação

nele. Neste caso, o dispositivo de E/S usado poderia ser um disco e várias operações de E/S e uma sequência de chamadas ao sistema seriam realizadas.

Uma vez que uma operação de entrada/saída tem início, dois fatos podem ocorrer: o primeiro é quando o controle só retorna para o programa do usuário, que gerou a chamada ao sistema, quando a operação tiver sido terminada; no segundo, o controle retorna ao programa do usuário sem esperar que a operação de entrada/saída tenha sido terminada. Ambas as situações têm vantagens e desvantagens.

O primeiro caso é mais simples, pois apenas uma operação de entrada/saída fica pendente a cada momento. Em compensação, limita a quantidade de operações simultâneas que podem ser feitas. O segundo caso é mais complexo, pois várias operações de entrada/saída podem ficar pendentes ao mesmo tempo e o sistema operacional precisa identificá-las para poder tratá-las.

Outro tipo de interrupção que merece uma atenção especial são as **interrupções de erro**. Elas são geradas quando o programa que está sendo executado tenta realizar algo não permitido pelo sistema operacional. Por exemplo, realizar o acesso a uma posição de memória protegida. Sempre que uma interrupção de erro ocorre o sistema operacional deve terminar o programa em execução de forma anormal. Uma mensagem de erro é enviada e a área de memória utilizada pode ser copiada num arquivo (*dump*) para que o usuário possa tentar descobrir a causa do erro.

## 4. GERENCIAMENTO DE PROCESSOS

- **Programa:**

Entidade **estática e permanente**:

- Sequência de instruções;
- Passivo, sob o ponto de vista do sistema operacional.

- **Processo:**

Entidade **dinâmica (ativo)**: altera o seu estado à medida que avança a sua execução. Informalmente é um programa em execução, descrevendo um dos possíveis fluxos do programa, só que o seu conceito é muito mais abrangente.

Composto por **código, dados e contexto**.

### 4.1. PROCESSOS

Um conceito chave da teoria dos sistemas operacionais é o conceito de **processo**, que é a forma pela qual o sistema operacional "enxerga" um programa e torna possível a sua execução.

**Um processo é mais do que o código do programa.** Ele inclui também a sua atividade atual, representada pelo **valor do PC** (Program Counter – contador do programa) e o **conteúdo dos**

**registradores** do processador. Um processo geralmente também inclui o **stack** do processo (contém dados temporários como parâmetros de subrotinas, endereços de retorno e variáveis temporárias) e uma **seção de dados** contendo variáveis globais.

Normalmente cada processo é identificado pelo sistema operacional através de um número inteiro único (PID – *Process Identifier*).

**Um mesmo programa pode ter várias instâncias em execução, ou seja, diferentes processos.** Apesar do código (programa) ser o mesmo, dados e momentos de execução (contexto) são diferentes.

Os processos executam:

- Programas de usuário;
- Programas de sistema.

## 4.2. CICLOS DE PROCESSOS

Os processos apresentam **dois ciclos básicos de operação**. São eles:

- **Ciclo de processador (CPU-burst)**: tempo que ocupa a CPU;
- **Ciclo de entrada e saída (I/O-burst)**: tempo em espera pela conclusão de um evento.

O primeiro ciclo é sempre de processador e as trocas de ciclos ocorrem por:

- Chamada de sistema (CPU → E/S);
- Interrupção (CPU → E/S ou E/S → CPU);
- Ocorrência de Evento (E/S → CPU).

O **tamanho dos ciclos de CPU e de E/S** varia de processo para processo. Dessa forma, podemos identificar dois tipos de processos. Uns que requisitam mais E/S e têm ciclos de CPU muito pequenos e outros que gastam mais tempo executando instruções e têm ciclos de E/S pequenos:

- **CPU-bound: ciclo de processador >> ciclo de E/S;**
- **I/O-bound: ciclo de E/S >> ciclo de processador.**

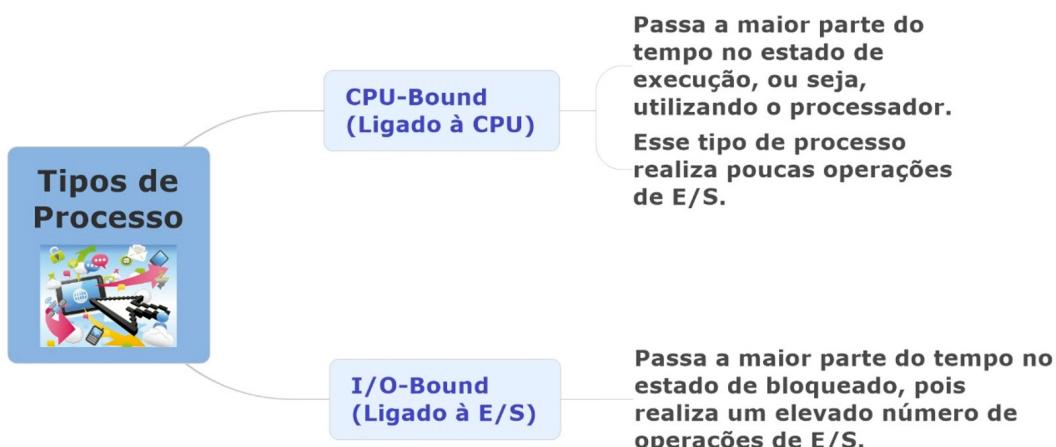


Figura. Tipos de Processos. Fonte: Quintão (2020)

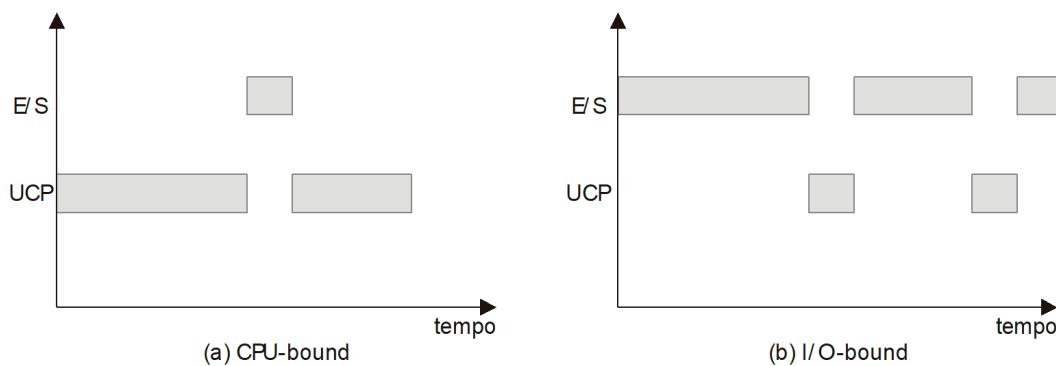


Figura. CPU-Bound x I/O Bound. Fonte: Machado e Maia (2007)

### 4.3. ESTADOS DE PROCESSOS

À medida que o processo é executado, ele troca de estado.

- **Novo (new)**: o processo está sendo criado;
- **Execução (running)**: o processo está usando o processador (instruções estão sendo executadas). Em sistemas monoprocessados no máximo um processo pode estar neste estado em qualquer instante de tempo;
- **Suspenso (waiting)**: o processo está esperando pela ocorrência de um evento. Alguns autores utilizam a nomenclatura “em espera” ou “bloqueado” para identificar este estado;
- **Pronto (ready)**: o processo está pronto para ser designado para a CPU. Alguns autores referenciam este estado como “Apto”;
- **Terminado (terminated)**: o processo acabou sua execução (ou seja, a última instrução do programa foi executada).

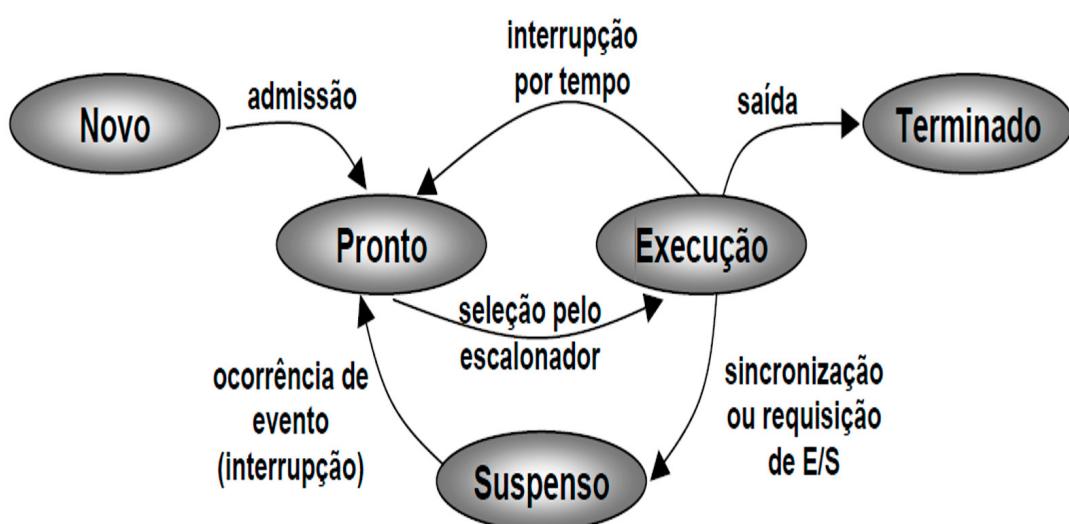


Figura. Diagrama de Estados de Processos

## DIRETO DO CONCURSO

**004.** (ESAF/MDIC/ANALISTA DE COMÉRCIO EXTERIOR/2012) Em sistemas multiprogramáveis, são estados do processo:

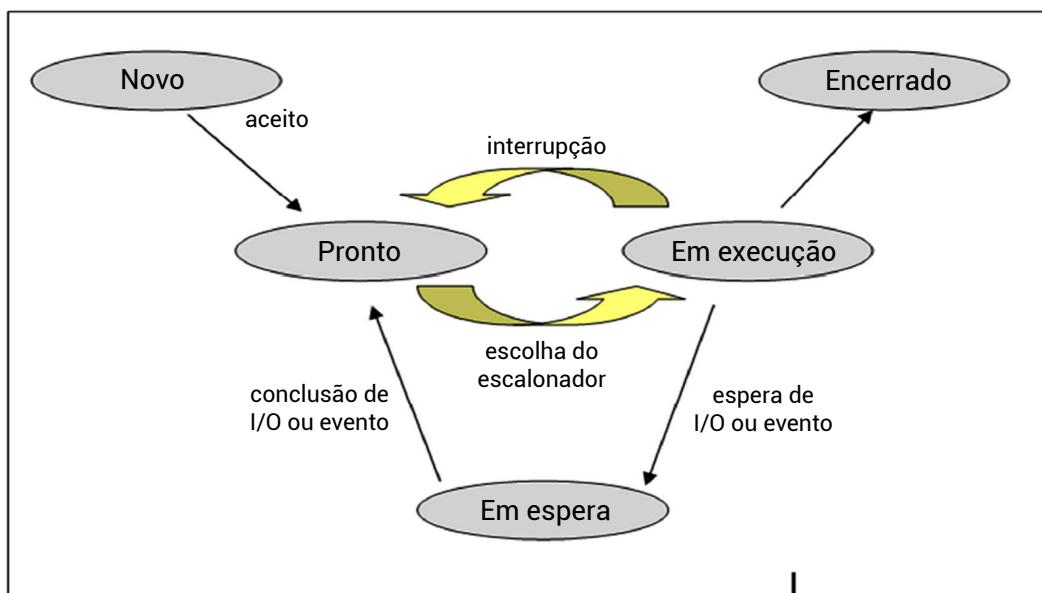
- A. Execução (executing). Disponível (available). Espera (log).
- b) Execução (running). Pronto (ready). Espera (wait).
- c) Direção (driving). Ponto (float). Espera (waste).
- d) Execução (running). Inserção (insert). Avaliação (controll).
- e) Pronta execução (ready running). Compartilhamento (sharing). Espera (wait).



**Sistemas operacionais multiprogramáveis/multitarefa** são capazes de executar várias tarefas simultaneamente. Ex.: Microsoft Windows, Linux.

À medida que o processo executa, ele muda de estado. O estado de um processo é definido em parte pela atividade atual desse processo. **Cada processo pode estar em um dos seguintes estados em sistemas multiprogramáveis:**

<b>Novo</b>	O processo está sendo criado.
<b>Em execução (Running)</b>	As instruções estão sendo executadas.
<b>Em espera (Wait)</b>	O processo está esperando a ocorrência de algum evento (como conclusão de operação de I/O).
<b>Pronto (Ready)</b>	O processo está esperando para ser atribuído a um processador.
<b>Encerrado</b>	O processo terminou sua execução.



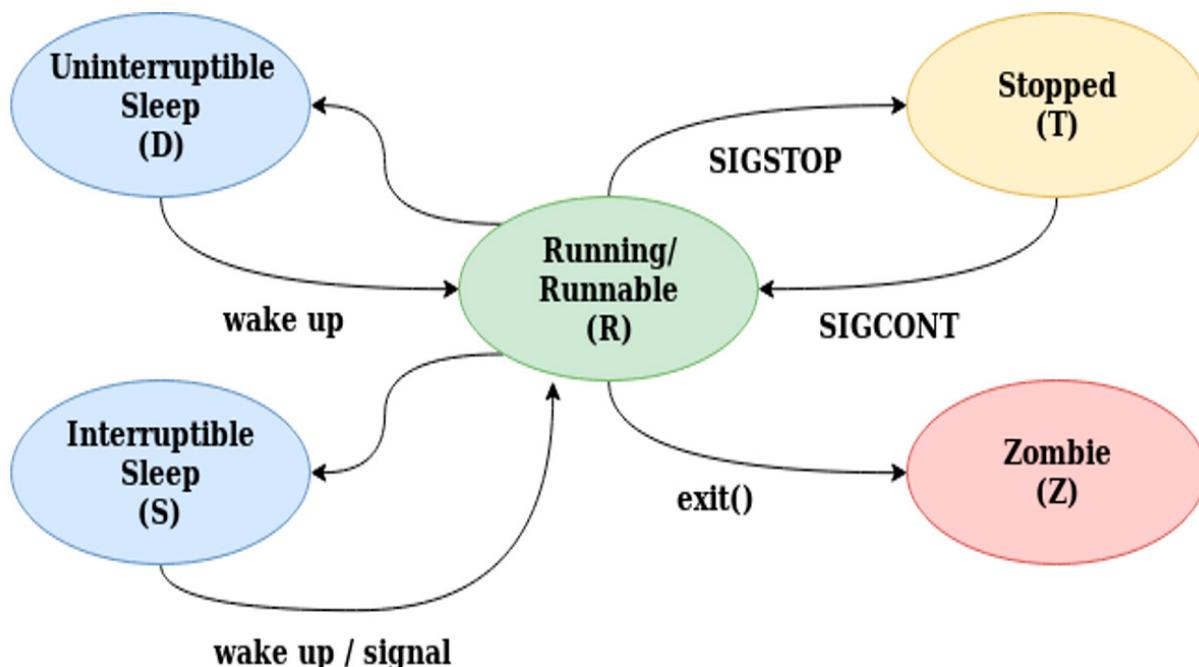
Os estados de processo variam dependendo do sistema operacional. É importante observar que apenas um processo pode estar em execução em qualquer processador a qualquer instante. No entanto, muitos processos podem estar prontos e em espera.

### Letra b.

**NOTA:** O sistema operacional Linux apresenta 5 estados para os processos:

- Executando ou Executável (*Running/Runnable*),
- Dormente (*Sleeping*), que se divide em dois: *Interruptible Sleep* e *Uninterruptible Sleep*;
- Parado (*Stopped*); e
- Zumbi.

Os processos se relacionam através de uma **máquina de estados**, conforme ilustrado a seguir:



Fonte: <https://medium.com/@cloudchef/linux-process-states-and-signals-a967d18fab64>

## 4.4. ESTRUTURA DO PROCESSO

Um processo é formado por três partes, conhecidas como contexto de hardware, contexto de software e espaço de endereçamento, que juntos mantêm todas as informações necessárias à execução de um programa.

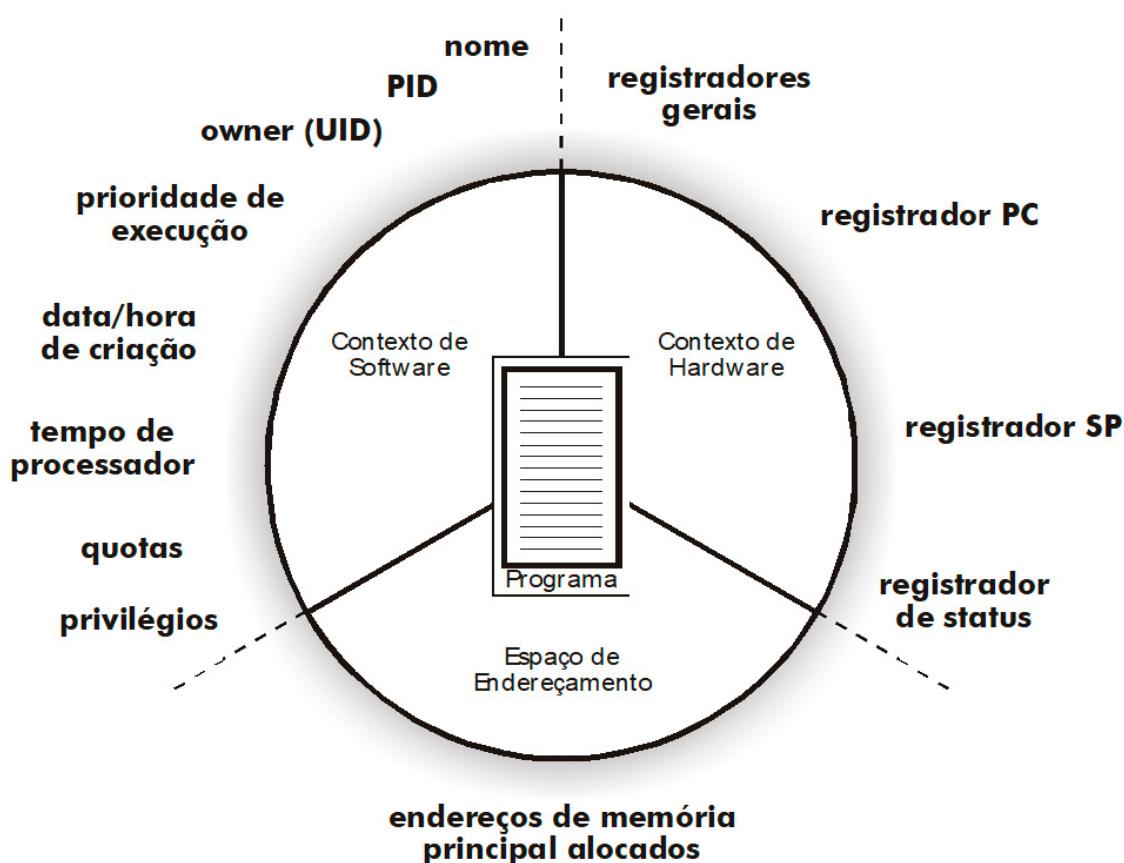


Figura. Estrutura do Processo. Fonte: Machado e Maia (2007)

- O **contexto de hardware** armazena o conteúdo dos registradores gerais da UCP, além dos registradores de uso específico como *program counter* (PC), *stack pointer* (SP) e *regrador de status*.

Quando um processo está em execução, o seu contexto de hardware está armazenado nos registradores do processador. No momento em que o processo perde a utilização da UCP, o sistema salva as informações no contexto de hardware do processo.

- No **contexto de software** são especificadas características e limites dos recursos que podem ser alocados pelo processo, como o número máximo de arquivos abertos simultaneamente, prioridade de execução e tamanho do buffer para operações de E/S.

O contexto de software é **composto por três grupos de informações sobre o processo: identificação, quotas e privilégios**.

- O **espaço de endereçamento** é a área de memória pertencente ao processo em que as instruções e dados do programa são armazenados para execução. Cada processo possui seu próprio espaço de endereçamento, que deve ser devidamente protegido do acesso dos demais processos.

## 4.5. BLOCO DESCRIPTOR DE PROCESSO

O processo é implementado pelo sistema operacional através de uma estrutura de dados chamada **bloco de controle do processo (Process Control Block – PCB)**. A partir do PCB, o sistema operacional mantém todas as informações sobre o contexto de hardware, contexto de software e espaço de endereçamento de cada processo.

Um PCB contém muitas informações associadas a um processo específico, incluindo:

- **Estado do processo;**
- **Contador de programa (PC):** indica o endereço da próxima instrução a ser executada;
- **Registradores de CPU;**
- **Informações para escalonamento de CPU:** inclui a prioridade do processo, ponteiros para as filas de escalonamento e outros parâmetros de escalonamento;
- **Informações para gerenciamento de memória:** inclui os valores dos registradores base e limite, e outras informações relacionadas.
- **Informações de contabilidade:** inclui a quantidade de tempo de CPU e tempo real utilizado, dentre outras.
- **Informações de estado de E/S:** inclui a lista de dispositivos de E/S alocados para o processo, uma lista de arquivos abertos, dentre outras.

A **estrutura de dados** responsável por habilitar o sistema operacional a localizar e acessar rapidamente o **bloco de controle de processo (PCB)** de um processo é denominada **Tabela de Processo**.

## 4.6. ALGORITMOS DE ESCALONAMENTO

**Obs.:** O problema do **escalonamento** é definido da seguinte forma: dados os vários processos que devem ser executados por um computador, deve-se definir qual a ordem em que os processos serão executados.

Os **algoritmos de escalonamento** podem ser divididos em duas classes:

- **preemptivos:** o processo que está sendo executado pela CPU pode ser interrompido e perder o controle do processador para outro processo de maior prioridade;
- **não preemptivos:** o processador não pode ser retirado do processo que está sendo executado, a não ser quando o seu ciclo de CPU terminar.

### DIRETO DO CONCURSO

**005.** (FGV/SEFAZ-RJ/AUDITOR-FISCAL DA RECEITA ESTADUAL - PROVA 1/2011) No contexto dos sistemas operacionais Windows, os mais modernos operam com base em um recurso, baseado em dois aspectos, descritos a seguir:

I – É fornecido um tempo às aplicações em execução, sendo o controle da CPU feito pelo próprio sistema operacional. Dessa forma, independentemente do fato de uma dada aplicação ter ou não terminado o que estava fazendo, esgotado seu *time slice*, o sistema operacional retoma o controle da CPU e o repassa para o próximo da fila.

II – Cada aplicação “roda” em um espaço próprio, de modo que, em caso de problemas, a aplicação em pane é finalizada, permanecendo as demais ativas em execução.

Esse recurso é conhecido por

- a) Multitarefa Preemptiva.
- b) Segmentação Paginada.
- c) Escalonamento Round Robin.
- d) Multiprocessamento Particionado.
- e) Multiprogramação com Swapping.



**Essa questão explora um conceito importante, envolvendo a multitarefa preeemptiva. Vamos lá! Dividindo os conceitos para conquistar :-!)**

**Sistemas monotarefa:** capazes de executar apenas uma tarefa de cada vez. Ex.: MS-DOS.

**Sistemas multitarefa:** capazes de executar várias tarefas simultaneamente. Ex.: Microsoft Windows, Unix, Linux.

**Multitarefa** é o nome que se dá à **possibilidade de executar mais de uma tarefa ou programa ao mesmo tempo**. Para isto é necessária a atuação conjunta do sistema operacional, que gerencia a execução dos programas, e do microprocessador (ou CPU), que executa as instruções. **Há duas formas de controlar o tempo fornecido aos programas**. São elas:

- **Multitarefa preemptiva:**

Nesse caso, o controle da CPU é feito pelo próprio sistema operacional.

Assim, independentemente do fato de um dado programa ter ou não terminado o que estava fazendo, esgotada sua “fatia de tempo – time slice”, o sistema operacional retoma o controle da CPU e o repassa para o próximo da fila.

Sua grande **vantagem** é que nenhum programa, por mais mal comportado que seja, pode se apoderar do controle da máquina e travar o sistema. **O Windows 10, por exemplo, utiliza multitarefa preemptiva!**

- **Multitarefa cooperativa (Não Preemptiva):**

Depende da cooperação de cada programa. Em outras palavras: se um programa mal feito tomar para si o controle da CPU e se recusar a devolvê-lo por um tempo demasiadamente longo, parece que os demais programas travaram. O Windows 3.1 e o Windows 98 ainda usavam multitarefa cooperativa.

**Letra a.**

A seguir, serão apresentados os **principais algoritmos de escalonamento de CPU**.

### 1. Primeiro a Chegar, Primeiro a ser Atendido (First In First Out (FIFO))

O algoritmo "Primeiro a Chegar, Primeiro a ser Atendido" (ou FCFS – *First-Come, First-Served*) é um algoritmo de escalonamento **não preemptivo** em que o primeiro processo que requer a CPU é o primeiro para o qual a CPU é alocada.

É o algoritmo mais simples para escalonamento de CPU e de fácil implementação pois trabalha com uma fila **FIFO** (*first-in, first-out*, ou primeiro a entrar, primeiro a sair).

### 2. Menor Job Primeiro (Shortest Job First)

O algoritmo "**Menor Job Primeiro**" (ou SJF – *Shortest-Job-First*) é um algoritmo de escalonamento que lida com os processos de acordo com a extensão dos seus ciclos de CPU. Quando a CPU está disponível, ela é alocada ao processo que possui o próximo ciclo de CPU menor.

**Obs.:** | **Shortest Job First:** o menor processo é executado primeiro (há o problema de estimar a duração do processo).

### 3. Escalonamento por Prioridade (Fila de Prioridades)

O algoritmo de "**Escalonamento por Prioridade**" faz com que **programas com prioridade maior sejam processados primeiros e sem interrupção até que seu ciclo de CPU seja concluído ou que ocorra uma espera natural (o programa entre em um ciclo de E/S)**. Caso dois processos da fila de prontos, a serem selecionados, possuam prioridades iguais, o processador é alocado àquele que chegou primeiro (ou seja, para desempate é utilizado o FCFS).

**Obs.:** | **Fila de prioridades:** a cada processo é atribuída uma prioridade, dependendo da natureza do processo. Os processos mais urgentes são executados primeiro.

### 4. Round-Robin

O algoritmo "**Round-Robin**" (RR), ou **Rodízio**, é um algoritmo de escalonamento naturalmente **preemptivo** projetado especialmente para sistemas *time-sharing* (de tempo compartilhado).

No escalonamento RR **cada processo pode ocupar a CPU durante uma quantidade de tempo definida pelo sistema**. Esta **fatia de tempo** recebe o nome de **time-slice ou time quantum (quantum de tempo)**.

Ao final deste tempo o processo que está sendo executado perde o direito de continuar usando a CPU, e é inserido no final da fila de prontos. Neste caso, o processo passa do estado executando para o estado pronto, sem ter passado pelo estado suspenso.

**Obs.:** | **Round Robin:** cada processo é executado por um período fixo de tempo (quantum), e depois volta para o final da fila.

## 4.7. THREADS

Muitos sistemas operacionais modernos estenderam o conceito de processo para permitir que um processo tenha **múltiplos fluxos de execução**, ou **threads**. Assim, o processo pode executar mais de uma tarefa de cada vez.

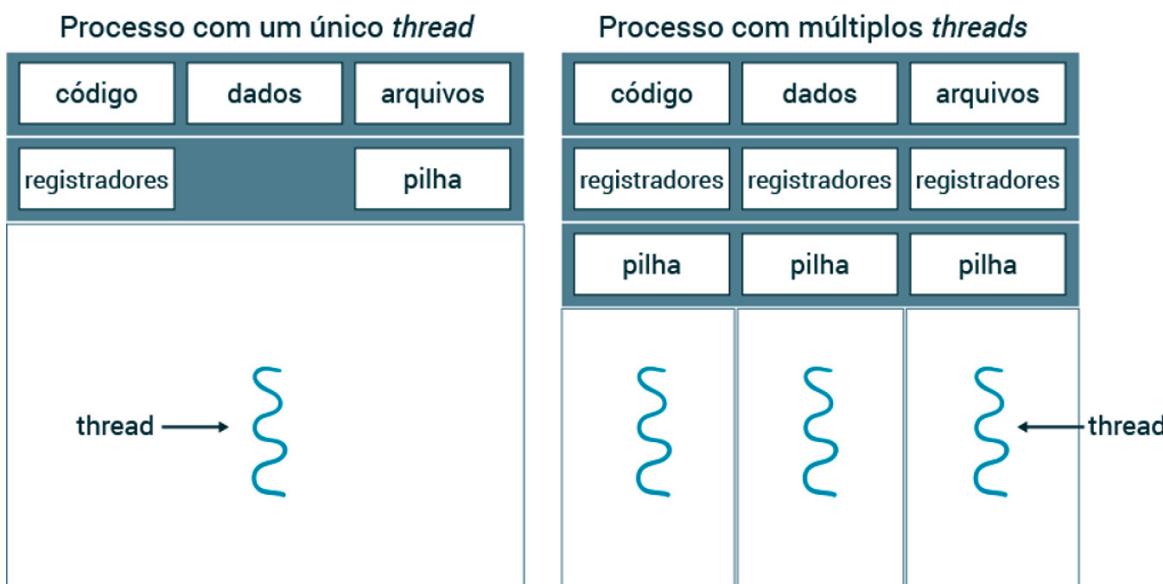
Os *threads* aumentam a eficiência do sistema operacional através da redução da sobrecarga (overhead) causada pela troca de contexto dos processos. Por isso, às vezes um thread é também chamado de **processo leve** (*lightweight process*).

Cada *thread* representa um **fluxo de controle separado**, com sua pilha e seu estado da máquina. Um *thread* pode então ser vista como a menor unidade de execução do sistema, consistindo de:

- um ID de *thread*;
- um contador de programa;
- um conjunto de registradores;
- uma área de stack (pilha).

Um *thread* compartilha com os outros *threads* de um mesmo processo:

- a seção de **código**;
- a seção de **dados**;
- os **recursos do sistema operacional**, como arquivos abertos.



Fonte: SILBERSCHATZ (2008)

O chaveamento da CPU entre *threads* de um mesmo processo é mais rápido e eficiente. Entretanto, o chaveamento entre *threads* de processos diferentes gasta um tempo maior, pois envolve o processo todo.

Os *threads* são eficientes para a exploração de concorrência dentro de uma aplicação e podem se comunicar usando a memória comumente compartilhada. Deve-se ter muito cuidado no que diz respeito a sincronização em regiões críticas.

Os **benefícios** básicos da programação com múltiplos *threads* são:

- aumento da capacidade de resposta em uma aplicação interativa;
- compartilhamento de recursos entre *threads* de um mesmo processo;
- economia de recursos e de tempo na criação e troca de contexto.

Em sistemas multiprocessados ocorre um incremento significativo nos benefícios mencionados anteriormente.

## DIRETO DO CONCURSO

**006.** (FCC/TRE-RN/ANALISTA JUDICIÁRIO/ANALISTA DE SISTEMAS/2011) Nos sistemas operacionais, múltiplas execuções que ocorrem no mesmo ambiente do processo com um grande grau de independência uma da outra é uma característica do modelo de processo que contempla o conceito de

- a) bus.
- b) switch.
- c) thread.
- d) disk array.
- e) split-cylinder.



Muitos sistemas operacionais modernos estenderam o conceito de processo para permitir que um processo tenha **múltiplos fluxos de execução**, ou ***threads***.

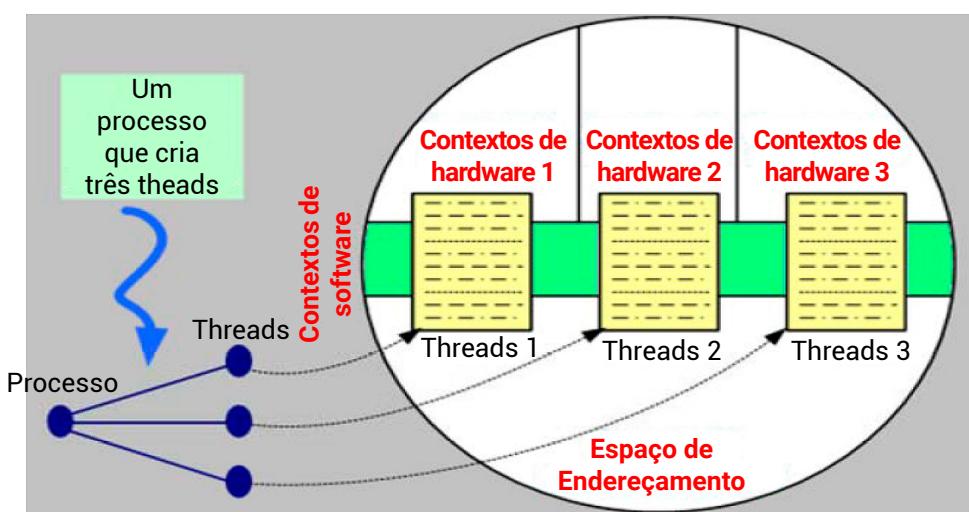


Figura. Exemplo de um processo que cria três threads.

**Letra c.**

**007.** (CESPE/STF-TÉCNICO JUDICIÁRIO DE TI/2013) No que se refere aos sistemas operacionais, julgue os itens subsequentes.

No contexto do sistema operacional, processo é uma abstração que reúne uma série de atributos, entre os quais estão o espaço de endereçamento, as permissões de acesso, as quotas e o fluxo de execução. Fluxo de execução é uma thread e multithreading é a associação de vários fluxos de execução a um único processo.



**Processo** é uma abstração que reúne uma série de atributos, entre os quais estão o espaço de endereçamento, as permissões de acesso, as quotas e o fluxo de execução. **Um thread** é um fluxo de execução e parte integrante de um processo e o **multithreading** é a associação de vários fluxos de execução a um único processo.

Certo.

## 4.8. SEMÁFOROS

- São tipos de variáveis que podem ser verificadas e alteradas em **instruções atômicas**, **sem possibilidades de interrupções**. Esse tipo de variável é empregado em tarefas como o **compartilhamento de recursos entre processos** (CESPE/ABIN - 2010).
- Em um semáforo existem as operações *down* e *up* (ou *sleep* e *wakeup*).
  - A chamada de sistema **UP** adiciona uma unidade ao valor corrente de um semáforo.
  - A operação **down** verifica se o valor é maior que 0.
    - Se for, ele decrementa um e continua.
    - Se o **valor for 0**, o processo (ou a *thread*) é “colocado para dormir” (bloqueado) sem completar a operação **down**.
- Se o valor do semáforo é **zero**, uma chamada de sistema **DOWN** não será completada e o processo será suspenso.
- É garantido que iniciada uma operação de semáforo, nenhum outro processo pode acessar o semáforo até que a operação tenha terminado ou sido bloqueada (ação atômica). Isso evita as **condições de corrida**.
- Quando um processo inicia a execução de uma chamada de sistema **UP** ou **DOWN**, nenhum outro processo terá acesso ao semáforo até que o processo complete a execução ou seja suspenso.

## 5. GERENCIAMENTO DE MEMÓRIA

### 5.1. PIRÂMIDE DAS MEMÓRIAS

Em ordem de velocidade tem-se, da memória mais veloz para a mais lenta: registradores, memória cache, memória principal (RAM) e memória secundária (auxiliar, como o disco rígido, por exemplo), conforme ilustrado na figura seguinte.

**Quanto mais no topo da pirâmide, mais nos aproximamos do processador (cache e registradores são internos) e na base da pirâmide estão os periféricos (HD, DVD etc.).**

No topo, estão memórias de baixa capacidade (poucos bits: 8, 16, 32, 64) e a capacidade vai **aumentando em direção à base** (atualmente TeraBytes).

A velocidade aumenta quando nos aproximamos do topo da pirâmide (os registradores estão na velocidade da CPU) e já as memórias na base são as mais lentas, pois normalmente dependem de acionamento eletromecânico.

Em relação ao custo, é fácil entender que fica muito mais fácil e barato colocarmos outro HD no nosso computador do que alterar todo o projeto da CPU para inserir mais alguns registradores.

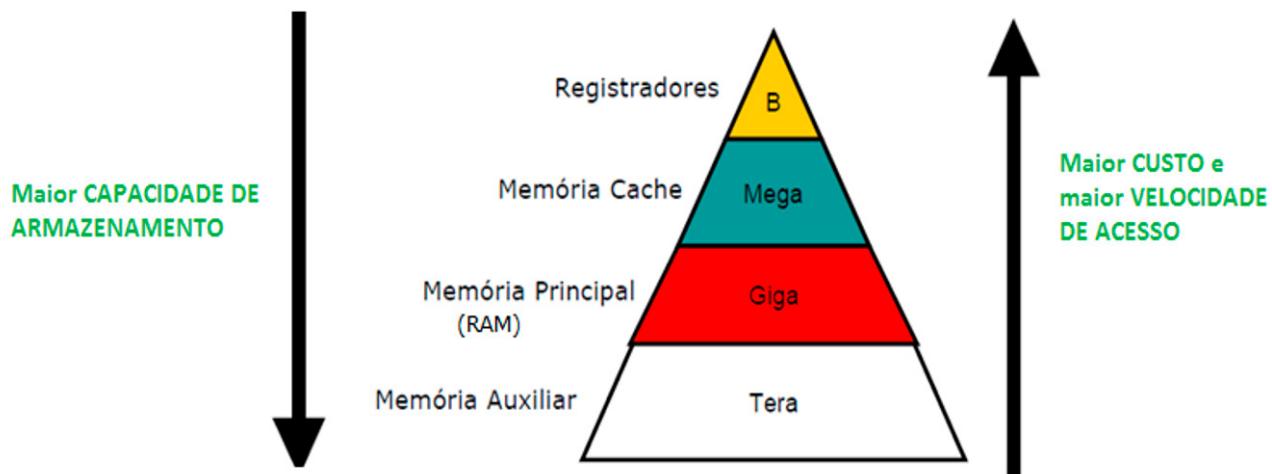


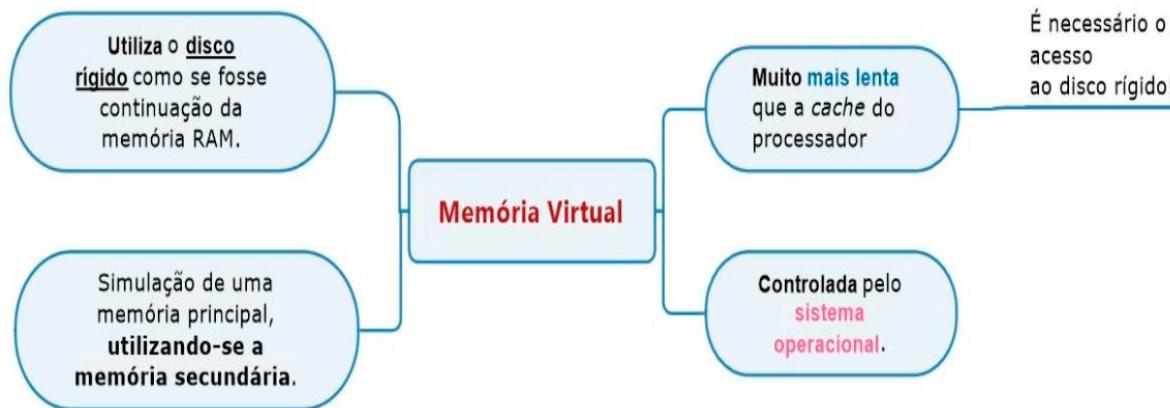
Figura. Relação entre os diversos tipos de dispositivos de armazenamento e seus tamanhos

### 5.2. MEMÓRIA VIRTUAL

**Consiste numa “parte do disco rígido” (HD), utilizada como uma extensão da memória RAM.**  
 Na verdade, a memória virtual é um arquivo conhecido como **Arquivo de Troca (Swap File)**.

A memória virtual é criada por ordem do Sistema Operacional assim que carregado, como uma “prevenção” para o caso de a RAM não ser suficiente.

Assim que o micro passa a utilizar a memória virtual, seu desempenho cai consideravelmente! (é aí que, na maioria das vezes, aparece a "ampulheta"). **A memória virtual não foi criada para aumentar a velocidade da RAM, mas para aumentar sua capacidade, já que o HD por ser memória magnética, é mais lento que a RAM.**



**Existem várias técnicas de se fazer memória virtual.** Cada sistema operacional utiliza a que foi programado para usar. São elas: **a paginação, a segmentação, a segmentação com paginação e a troca (swapping).**

**Paginação:** nesse caso, a memória virtual é dividida em **blocos (páginas) de tamanhos fixos**.

Permite que partes do programa estejam gravados na memória física e outras no disco.

**Segmentação:** memória dividida em **blocos (segmentos) de tamanhos variáveis**. Permite que partes do programa estejam gravados na memória física e outras no disco.

**Segmentação com paginação:** o espaço de endereçamento é dividido em segmentos, e estes, por sua vez, por páginas.

**Troca (Swapping):** é a técnica mais arcaica das três, que consiste na **troca dos programas (não usados) entre a RAM e o disco rígido**. Em outras palavras, é uma técnica que permite a um computador executar programas e manipular arquivos de dados maiores que a memória principal. O sistema operacional copia o máximo de dados possível na memória principal e deixa o restante no disco. Quando o sistema operacional precisa de dados do disco, ele troca uma parte dos dados (chamados de página ou segmento) na memória principal por uma parte dos dados no disco

## DIRETO DO CONCURSO

**008. (CESPE/STF/TÉCNICO JUDICIÁRIO DE TI/2013)** No que se refere aos sistemas operacionais, julgue os itens subsequentes.

Embora a técnica de paginação auxilie a multitarefa, permitindo a alocação de vários processos em áreas diferentes de memória, ela não elimina o problema de que tal alocação deva ser contígua.



A alocação de memória com paginação **não necessita que o processo esteja armazenado de forma contígua na memória principal.**

A paginação é uma organização de memória que permite que a memória física seja vista como se estivesse dividida em **blocos de tamanho fixo**, chamados **frames** (traduzidos por alguns autores como "quadros"). A memória lógica, que contém as instruções/dados do programa, também é dividida em blocos de mesmo tamanho, denominados **páginas (pages)**.

Os frames e as páginas possuem o **mesmo tamanho**.

Para a alocação de um processo na memória é necessário expressar seu tamanho em número de páginas. Se o processo tiver tamanho  $n$ , ele necessitará de no máximo  $n$  frames. Um frame será alocado para cada página do processo.

A **vantagem** principal de armazenar processos em áreas não contíguas é o **melhor aproveitamento da memória principal, já que um frame vazio pode ser utilizado por qualquer página de qualquer processo.**

Não há fragmentação externa entre frames e ocorre pouca fragmentação interna. Na paginação, a fragmentação interna ocorre somente na última página caso o processo não utilize todas as páginas completamente.

O sistema operacional trabalha com tabelas para monitorar a utilização dos frames.

**Errado.**

**009.** (CESGRANRIO/BNDES/ANALISTA DE SUPORTE/2008) O uso de memória secundária (de menor velocidade, comparada com a memória principal) como memória virtual acaba ocasionando perda de desempenho. Neste contexto, supondo que um computador da linha PC, monoprocessado, admita expansibilidade de seus componentes, a solução que atualmente oferece melhor relação custo-benefício para atenuar o problema é:

- a) adicionar um disco rígido de maior capacidade.
- b) adicionar memória RAM.
- c) aumentar o tamanho do arquivo de troca (swap) do sistema operacional.
- d) substituir o processador por um mais veloz.
- e) trocar o computador todo por um mais moderno, pois não há solução para o problema.



Quando a capacidade da memória principal é totalmente utilizada, usamos a **memória virtual**.

- Recurso gerenciado pelo Sistema Operacional.
- Aloca um espaço no disco rígido do computador fazendo com que ele (o HD ou disco rígido) funcione como um complemento da memória principal.

A memória virtual não tem influência no tempo de acesso médio à memória principal. Sua velocidade é inferior à da memória RAM real, portanto ela não aumenta a velocidade, e sim apenas a capacidade.

**Vamos aos itens:**

**Item a) Errado.** A técnica de memória virtual é uma técnica de gerenciamento que combina a memória principal e a secundária dando ao usuário a ideia de existir uma memória maior que a memória principal. Aumentar o disco não resolveria o problema, pois alguns dados e instruções ainda precisariam ser transferidos entre a memória principal e a secundária, o que ocorre de forma lenta, já que o acesso à memória secundária é mais lento.

**Item b) Certo.** O aumento da memória RAM (que é barata e isso é fácil de ser realizado) evitaria a utilização do HD (disco rígido) como extensão da memória principal.

**Item c) Errado.** Custo alto, pois envolveria alteração do Kernel (núcleo do sistema operacional), e os benefícios não seriam tão visíveis.

**Item d) Errado.** Isso não resolveria o problema, a memória virtual depende da memória principal, e não do processador a ser utilizado.

**Item e) Errado.** A troca do computador por um mais moderno poderia atenuar o problema, mas seria uma solução mais cara que a apresentada na alternativa B.

**Letra b.**

---

### 5.3. ALGORITMOS PARA ESTRATÉGIAS DE ALOCAÇÃO DE MEMÓRIA

Como exemplos de algoritmos de estratégias de alocação de memória podemos citar:

**Lista de Áreas Livres, Best-fit, Worst-fit e First-fit.**

Tanembaum chama as técnicas de "Gerenciamento de Memória com Listas Encadeadas" e traduz os termos da seguinte forma:

- **Algoritmo Primeiro Ajuste (first-fit):** "o gerenciador de memória varre toda a lista de segmentos até localizar uma lacuna suficientemente grande." É um algoritmo rápido, pesquisando o mínimo possível.
- **Próximo ajuste (lista de áreas livres):** "funciona da mesma maneira que o primeiro ajuste, exceto que monitora a posição em que ele está sempre que encontra uma lacuna adequada. Da próxima vez que é chamado para localizar uma lacuna, ele começa pesquisando na lista a partir do lugar em que deixou da última vez, em vez de sempre partir do começo".
- **Melhor ajuste (best-fit):** "pesquisa na lista inteira e pega a menor lacuna que seja adequada".
- **Pior ajuste (worst-fit):** "sempre pega a maior lacuna disponível para evitar o problema de dividir lacunas quase exatas entre um processo e uma lacuna minúscula".
- **Ajuste rápido:** "mantém listas separadas para os tamanhos de lacunas mais comuns".

## DIRETO DO CONCURSO

**010.** (CESGRANRIO/CASA DA MOEDA/ANALISTA DE NÍVEL SUPERIOR - BANCO DE DADOS/2009) Determinado sistema operacional apresenta a seguinte lista de segmentos de memória disponíveis, em sequência: 10 KB, 20 KB, 40 KB, 80 KB, 160 KB.

Nesse momento, um processo solicita alocação de memória para 16 KB. Considerando-se o uso do algoritmo first fit, que segmento será utilizado na alocação?

- a) 10 KB.
- b) 20 KB.
- c) 40 KB.
- d) 80 KB.
- e) 160 KB.



O **First-Fit** é um representante dos algoritmos para "Estratégias de Alocação de Memória". Nele, **a primeira posição livre de tamanho suficiente para carregar o programa é escolhida**.

O enunciado informa uma lista de segmentos livres e indica que um segmento de 16KB deve ser alocado conforme regras do algoritmo **first fit**.

Então, o primeiro segmento em que couber o segmento será o utilizado, ou seja, a letra B que possui 20KB disponíveis. É importante observar que o segmento de 20KB será dividido em 2 segmentos (uma vez que não é do mesmo tamanho do segmento a ser armazenado): um de 16KB, que será utilizado, e outro com os 4KB restantes. Assim, a nova lista de segmentos disponíveis será 10 KB, 04 KB, 40 KB, 80 KB, 160 KB.

**Letra b.**

## 5.4. POLÍTICA DE BUSCA DE PÁGINAS

A **política de busca de páginas** determina quando uma página deve ser carregada para a memória. Basicamente, tem-se 2 estratégias: paginação por demanda e paginação antecipada.

- **Na paginação por demanda (*demand paging*)**, as páginas dos processos são transferidas da memória secundária para a principal apenas **quando referenciadas**.
- **Na paginação antecipada (*anticipatory paging ou prepaging*)**, o sistema **carrega para a memória principal, além da página referenciada, outras páginas que podem ou não ser necessárias** ao processo ao longo do seu processamento. Se imaginarmos que o programa está armazenado sequencialmente no disco, existe uma grande economia de tempo em levar um conjunto de páginas da memória secundária, ao contrário de carregar uma de cada vez. No entanto, caso o processo não precise das páginas carregadas antecipadamente, o sistema terá perdido tempo e ocupado memória principal desnecessariamente.

## 5.5. REENTRÂNCIA

**Reentrância** é a capacidade de um código de programa (código reentrante) poder ser compartilhado por diversos usuários, exigindo que apenas uma cópia do programa esteja na memória.

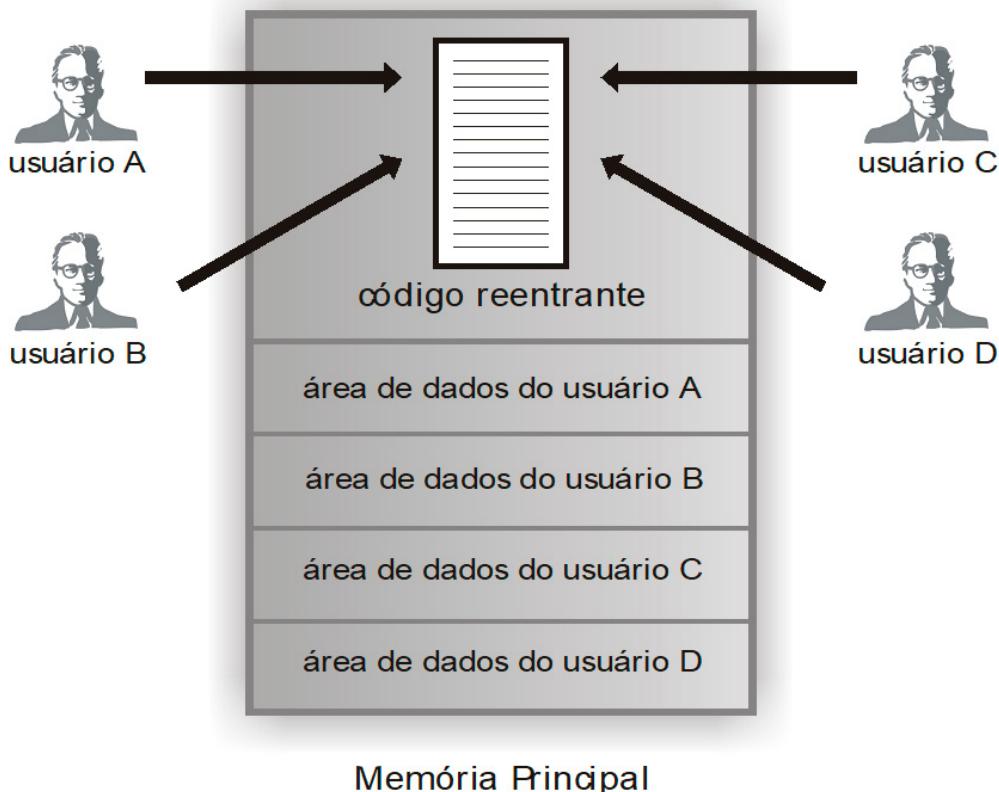


Figura. Reentrância. Fonte: Machado e Maia (2007)

Uma característica da reentrância é que o código não pode ser modificado por nenhum usuário no momento em que está sendo executado.

A reentrância permite que cada usuário possa estar em um ponto diferente do código reentrante, manipulando dados próprios, exclusivos de cada usuários.

## 6. SISTEMA DE ARQUIVOS

**Sistema de arquivos** é o conjunto de regras que determina como os dados serão escritos em um disco (ou partição).

- Normalmente, cada sistema operacional tem o(s) seu(s) próprio(s) sistema(s) de arquivo(s).
- Durante o processo de formatação da unidade é que se escolhe o sistema de arquivos desejado.

- Um disco rígido com várias partições pode ter um sistema de arquivos em cada uma delas. Isso é comum quando se instalam dois sistemas operacionais que usam sistemas de arquivos diferentes no mesmo computador.
- Um arquivo grande, ao ser gravado no disco, é fragmentado em vários *clusters*.
- **Cluster** é a menor unidade de alocação de arquivos reconhecida pelo sistema operacional.
- **Um arquivo pode ocupar vários clusters, mas um cluster não pode conter mais de um arquivo.**

Sistema Operacional	Sistemas de arquivos aceitos
<b>MS-DOS e Windows 95</b>	FAT 16
<b>Windows 98 e ME</b>	FAT 16 e FAT 32
<b>Windows NT</b>	FAT 16 e NTFS
<b>Windows 2000, XP, Vista, Windows 7</b>	FAT 16, FAT 32 e NTFS

## DIRETO DO CONCURSO

**011.** (FCC/TRF-5A REGIÃO/ANALISTA/2003) O sistema de arquivos utilizado na formatação dos discos e reconhecido por TODAS as versões do sistema operacional Windows é denominado:

- EXT2;
- FAT 16;
- FAT 32;
- NTFS;
- HPFS.



**Item a) Errado.** Os sistemas de arquivos EXT2, EXT3 e ReiserFS são suportados pelo Linux.

**Item b) Certo.** O **FAT 16** é o sistema de arquivos utilizado pelo MS-DOS e pelo Windows 95. Todos os sistemas operacionais da Microsoft conseguem trabalhar com esse sistema de arquivos. Possui limitação de 2 GB para o tamanho da partição (divisão existente no disco rígido que marca onde começa e termina um sistema de arquivos). Ainda, nesse sistema de arquivos, o nome dos arquivos tem, no máximo, 11 letras (oito para o nome e três para a extensão). A extensão é o que identifica o tipo de arquivo, por exemplo, a extensão ".doc", no nome projeto.doc, identifica que o arquivo projeto é um documento de texto do Word.

**Item c) Errado.** O sistema **FAT 32** surgiu como opção ao FAT 16, que só consegue entender partições de discos de até 2 GB. Nem o MS-DOS, nem o Windows 95, nem o Windows NT dão suporte a esse sistema. Uma partição formatada com FAT 32 pode ter, no máximo, 2.048

gigabytes de capacidade (2 TB); o tamanho máximo do arquivo que o sistema consegue reconhecer é de 4 GB.

**Item d) Errado.** NTFS (*New Technology File System*) é o sistema de arquivos padrão para o Windows NT e seus derivados (Windows 2000, Windows XP, Windows Vista, Windows Server 2003, em diante). Portanto, todos os sistemas baseados no NT – que possuem o núcleo do Windows NT – dão suporte ao NTFS!

**Item e) Errado.** HPFS (*High Performance File System*) é o sistema de arquivos utilizado pelo sistema operacional OS/2, da IBM.

**Letra b.**

## 7. CONTROLE DE ACESSO

O **controle de acesso**, na segurança da informação, é composto dos processos de **autenticação, autorização e auditoria (accounting)**. A **autenticação** identifica quem acessa o sistema, a **autorização** determina o que um usuário autenticado pode fazer, e a **auditoria** diz o que o usuário fez.

Neste contexto o **controle de acesso** pode ser entendido como a **habilidade de permitir ou negar a utilização de um objeto** (uma entidade passiva, como um sistema ou arquivo) **por um sujeito** (uma entidade ativa, como um indivíduo ou um processo).

As **técnicas de controle de acesso** são normalmente categorizadas em **discricionárias** e **obrigatórias**.

- O **controle de acesso discricionário** (*Discretionary access control* ou DAC) é uma **política de controle de acesso determinada pelo proprietário (owner) do recurso** (um arquivo, por exemplo). O proprietário do recurso decide quem tem permissão de acesso em determinado recurso e qual privilégio ele tem.

O DAC tem dois conceitos importantes:

- **Todo objeto em um sistema deve ter um proprietário.** A política de acesso é determinada pelo proprietário do recurso. Teoricamente um objeto sem um proprietário é considerado não protegido.
- **Direitos de acesso e permissões são dadas pelo proprietário do recurso** a usuários individuais ou grupos de usuários.

Controles de acesso discricionários podem ser aplicados através das seguintes **técnicas**:

- **Listas de controle de acesso (ACLs)** definem os direitos e permissões que são dados a um sujeito sobre determinado objeto.
- **Controles de acesso baseados em papéis (roles)** definem os direitos e permissões baseados no papel que determinado usuário desempenha na organização. Esta estratégia simplifica o gerenciamento das permissões dadas aos usuários.

**No Controle Discricionário:**

- São concedidos aos usuários privilégios específicos e individualizados de acesso aos objetos;
- Esses privilégios podem ser distintos para cada usuário em um mesmo objeto.
- No controle de acesso obrigatório (*mandatory access control ou MAC*) a política de acesso é determinada pelo sistema e não pelo proprietário do recurso. Este controle é utilizado em sistemas de múltiplos níveis cujos dados são altamente sensíveis, como algumas informações governamentais e militares. Um sistema de múltiplos níveis é constituído de um computador que manipula múltiplos níveis de classificação entre sujeitos e objetos.

**No Controle Mandatório:**

- Cada objeto tem um determinado nível de classificação e cada usuário, um nível de liberação;
- Um usuário só pode ter acesso de leitura a um objeto se seu nível de liberação é maior ou igual ao nível de classificação do objeto;
- Todos os objetos atualizados (incluídos e alterados) por um usuário assumem automaticamente o nível de classificação correspondente ao seu nível de liberação.

 **DIRETO DO CONCURSO**

**012.** (CESPE/TCU/2009) Com relação aos sistemas operacionais Linux e Windows, julgue os itens a seguir.

[O NTFS, como o UFS, é um sistema de arquivos hierarquizado com arquivos e diretórios, em que cada arquivo tem um proprietário, mas não pertence a um grupo. O acesso não é regulado por um conjunto de bits de permissão, mas por listas de acesso, implementando um modelo discricionário.]



Isso mesmo! No Controle Discricionário são concedidos aos usuários privilégios específicos e individualizados de acesso aos objetos.

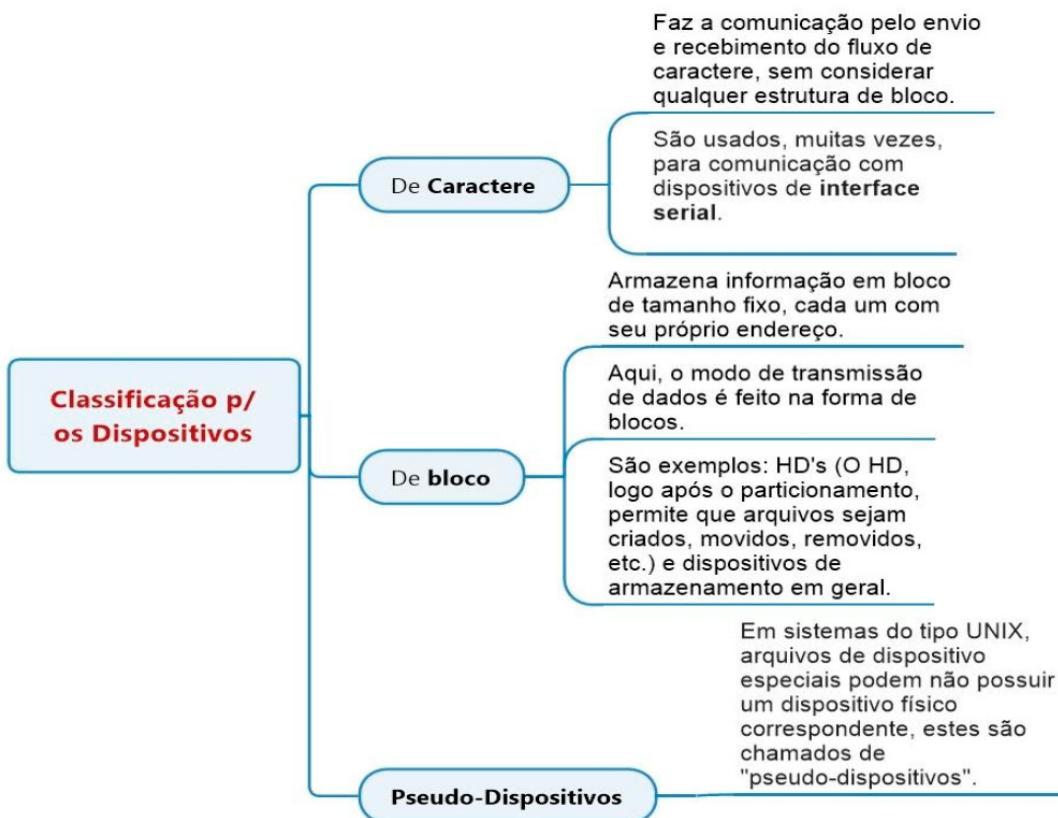
Certo.

---

## 8. GERENCIAMENTO DE ENTRADA E SAÍDA

### 8.1. CLASSIFICAÇÃO PARA OS DISPOSITIVOS

De maneira geral, os dispositivos de entrada e saída podem ser divididos em:



## 8.2. BUFFERING

A técnica de *buffering* consiste na **utilização de uma área de memória para a transferência de dados entre os periféricos e a memória principal** denominada **buffer** (Zanine, 2000).

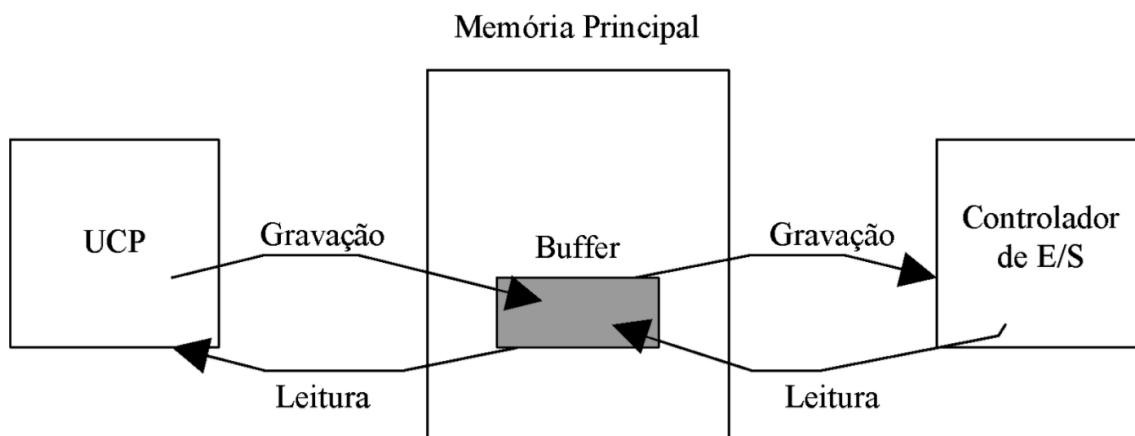


Figura - Operações Utilizando Buffer. Fonte: Zanine (2000)

O *buffering* é utilizado para **minimizar o problema da disparidade da velocidade de processamento existente entre a UCP e os dispositivos de E/S**. O objetivo do *buffering* é manter, na maior parte do tempo, UCP e dispositivos de E/S ocupados (Zanine, 2000).

## 8.3. DEADLOCK

Para entender o que é **deadlock**, observe a seguinte **analogia**: duas pessoas, em margens opostas, querem atravessar um rio. A única ligação entre as duas margens é um caminho de pedras. A restrição que existe é que cada pedra só pode ser pisada por um pé de cada vez. Se as duas pessoas começarem a atravessar o rio ao mesmo tempo, elas se encontrarão em algum ponto e não poderão prosseguir a travessia, o que corresponde a uma **situação de deadlock (ou impasse)**. Nesse ponto, ou ambas voltam para trás e tentam recomeçar a travessia novamente ou uma delas retorna e a outra prossegue.

Nessa analogia, cada pessoa pode ser vista como um **processo**, cada pedra como um **recurso** e o ato de pisar numa pedra como a **aquisição de um recurso**.

Uma **situação de deadlock (impasse)** em um ambiente multiprogramado **ocorre quando dois ou mais processos estão esperando indefinidamente por um evento que só pode ser causado por um dos processos que também está esperando por um evento**.

As **condições obrigatórias** (necessárias e suficientes) para ocorrência do **deadlock** são:

**1. Exclusão mútua (*mutual exclusion*):**

- Ou um recurso está associado a um processo ou está disponível.
- Um recurso que não pode ser usado por mais de um processo de cada vez.

**2. Monopolização de recursos ou Posse e espera (*hold and wait*):**

- Processos que retêm recursos podem solicitar novos antes da liberação.
- Processos que já possuem recursos podem solicitar novos recursos mantidos por outros processos.

**3. Não preempção (*no preemption*):**

- Nenhum recurso pode ser retirado à força de um processo, os recursos só podem ser liberados pela ação explícita do processo.
- Recursos concedidos devem ser liberados pelo processo requisitante e não tomados.
- Não haverá interferência do SO.
- Preemptível: pode ser retirado do processo, não há **deadlock**.
- Não preemptível: não pode ser retirado, há **deadlock**.

**4. Espera circular (*circular wait*):**

- Dois ou mais processos formar uma **cadeia circular** em que cada processo espera por um recurso que o próximo processo na cadeia detém.

### DIRETO DO CONCURSO

**013. (CESGRANRIO/TRANSPETRO/ANALISTA DE SISTEMAS JÚNIOR – SOFTWARE/2011)**  
Os Sistemas Operacionais estão sujeitos a um fenômeno denominado deadlock. Para que

uma situação de deadlock seja criada, as seguintes condições devem acontecer simultaneamente:

- a) exclusão mútua (mutual exclusion), monopolização de recursos (hold and wait), não preempção (no preemption) e espera circular (circular wait).
- b) exclusão mútua (mutual exclusion), transferência excessiva de páginas (thrashing), superposição de processos (process overlapping) e espera circular (circular wait).
- c) transferência excessiva de páginas (thrashing), superposição de processos (process overlapping), monopolização de recursos (hold and wait) e não preempção (no preemption).
- d) exclusão mútua (mutual exclusion), monopolização de recursos (hold and wait), superposição de processos (process overlapping) e falha de escalonamento (scheduling fail)
- e) transferência excessiva de páginas (thrashing), não preempção (no preemption), espera circular (circular wait) e falha de escalonamento (scheduling fail).



"Um conjunto de processos está em **deadlock** se cada processo no conjunto está esperando por um evento que somente um outro processo do conjunto pode causar. Fonte: (<http://www2.ic.uff.br/~boeres/slidesSOI/Deadlock6a.pdf>.)"

Conforme visto, as **condições obrigatórias** (necessárias e suficientes) para ocorrência do deadlock são:

1. Exclusão mútua.
2. Posse e espera (Monopolização de recursos).
3. Não preempção.
4. Espera circular.

Letra a.

## SERVIÇO DE DIRETÓRIO

Um diretório é uma estrutura hierárquica que armazena informações sobre objetos na rede.

Um **serviço de diretório** funciona de modo semelhante a uma agenda, e pode ser utilizado para **organizar e principalmente ter um local centralizado para a busca de informações** necessárias no dia a dia, para nossos trabalhos.

Principais soluções de serviço de diretório existentes no mercado:

- 1- Open Ldap para **Sistemas Open Source**.
- 2- EDirectory para Sistemas Novell.
- 3- Active Directory para **Sistemas Microsoft** e com suporte para todos acima citados.

Referência: <http://www.linhadecodigo.com.br/artigo/2422/o-que-e-o-active-directory.aspx#ixzz62LDIX4Jp>

## a) LDAP (Lightweight Directory Access Protocol)

- **Protocolo para pesquisa e gerenciamento de diretórios rodando sobre TCP/IP**, criado como uma alternativa ao protocolo *Directory Access Protocol* (DAP), que era mais difícil de utilizar, e era derivado da família de padrões X.500 da ITU-T.
- **O LDAP é um protocolo útil pois permite a centralização, em um único local da rede, de informações acerca de usuários, senhas e diretórios, entre outras. Além disso, o acesso a esses dados pode ser feito de forma segura, pois a maioria dos servidores LDAP oferece conexões criptografadas usando SSL.**
- Cabe destacar que o **LDAP é independente de plataforma**, pode ser usado tanto no **Active Directory do Windows**, como em várias implementações que se tem para **Linux**, como Open LDAP, por exemplo. Além disso, LDAP pode oferecer conexões criptografadas para trafegar as informações de forma segura.
- **Um diretório LDAP segue o modelo X.500**, que é uma estrutura hierárquica de nós chamada de *Directory Information Tree* (DIT), cada um consistindo de um conjunto de atributos com seus respectivos valores.
- **LDAP: protocolo que organiza as informações como uma árvore** e permite localizar **pessoas, recursos e serviços, entre outros**. Descrito na RFC 2251, é **uma versão simplificada do serviço X.500 do OSI**.
- A estrutura de diretórios do LDAP possui o formato de uma árvore em que **cada elemento da estrutura ocupa uma posição**, e é definida nos termos da **linguagem ASN.1**.
- O endereço completo da entrada desejada é unicamente identificado pelo *Distinguished Names (DN)*, e os nós que ficam no caminho até a entrada são chamados de *Relative Distinguished Names (RDN)*. Os objetos da estrutura podem ser do tipo *container*, contendo objetos subordinados, ou do tipo *folha*, sem objetos subordinados.
- Dentre os atributos comuns do protocolo LDAP está o atributo para **armazenamento do sobrenome do usuário**. A sigla utilizada para este atributo é **SN**.
- Os elementos do LDAP permanecem na base de dados no formato LDIF (*LDAP Data Interchange Format*). Cada elemento da estrutura de dados do LDAP associada a uma *ObjectClass*. Cada *ObjectClass* possui um *Object ID (OID)* e um nome.
- Nos sistemas operacionais Unix-Like, há um processo *daemon* que atende as requisições ao LDAP. Nesses sistemas, o arquivo base para a configuração do protocolo é o **slapd.conf**.
- O acesso aos diretórios do LDAP pode ser configurado delimitando grupos e tipos de acesso.
- **O LDAP é gerenciado por comandos**. Exemplos:
  - **Connect**: permite a conexão com um servidor LDAP;
  - **Bind**: utilizado para autenticar o cliente no servidor LDAP. Ele envia o DN (*Distinguished Name*), a senha do usuário e a versão do protocolo que está sendo usada;

- **Modify:** permite alterar registros no LDAP;
- **Search:** o servidor busca e devolve as entradas do diretório que obedecem ao critério da busca;
- **Unbind:** realiza operação de fechamento da conexão entre cliente e servidor.

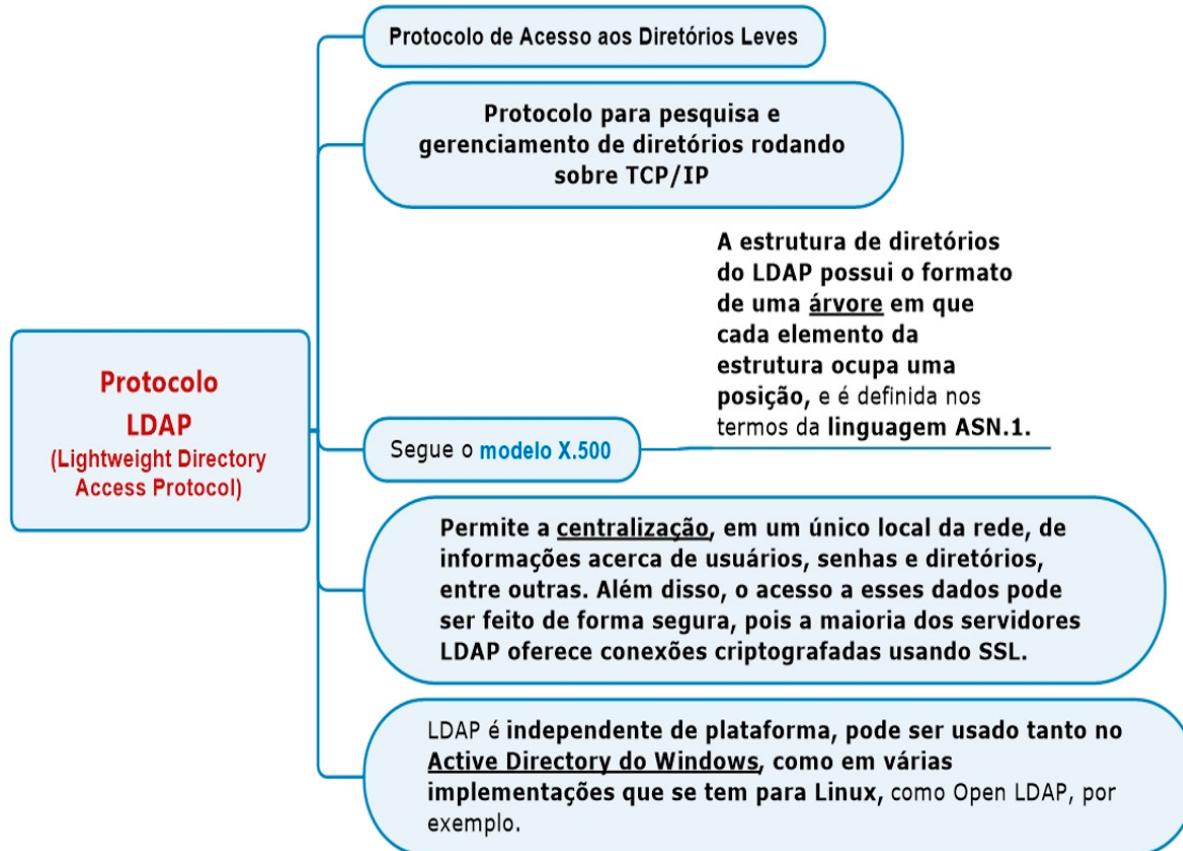
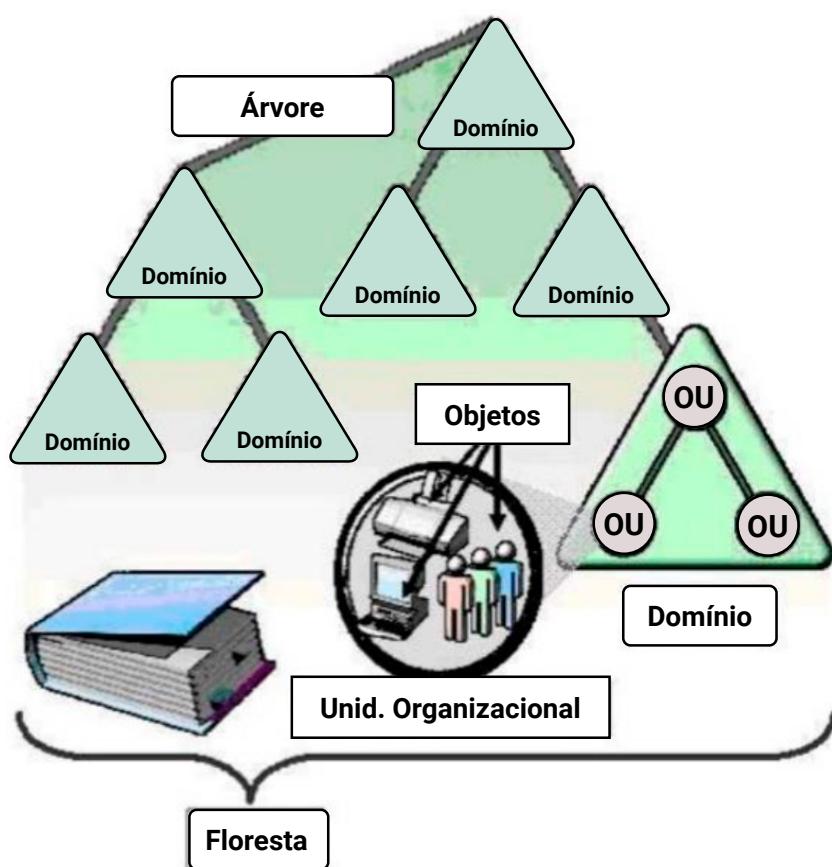


Figura. LDAP. Fonte: Quintão (2020)

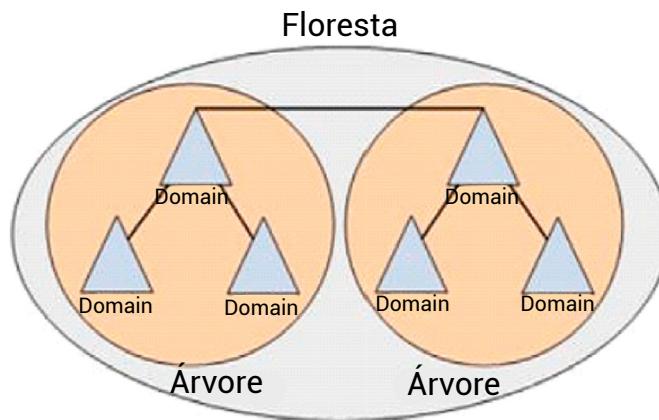
### b) Active Directory

- É o **serviço de diretórios da Microsoft** (Windows 2000/2003/2008/2016 etc.);
- Utiliza **padrões de mercado** e é compatível com LDAP.
- O **Active Directory (AD)** está localizado nas máquinas **controladoras de domínio**.
- Um formato conhecido como um Active Directory com **menos recursos** é o **AD LDS ou Active Directory Lightweight Directory Services**.
- O **repositório AD** armazena diversos objetos e recursos, como: Contas de usuários, Grupos, Computadores, Impressoras.

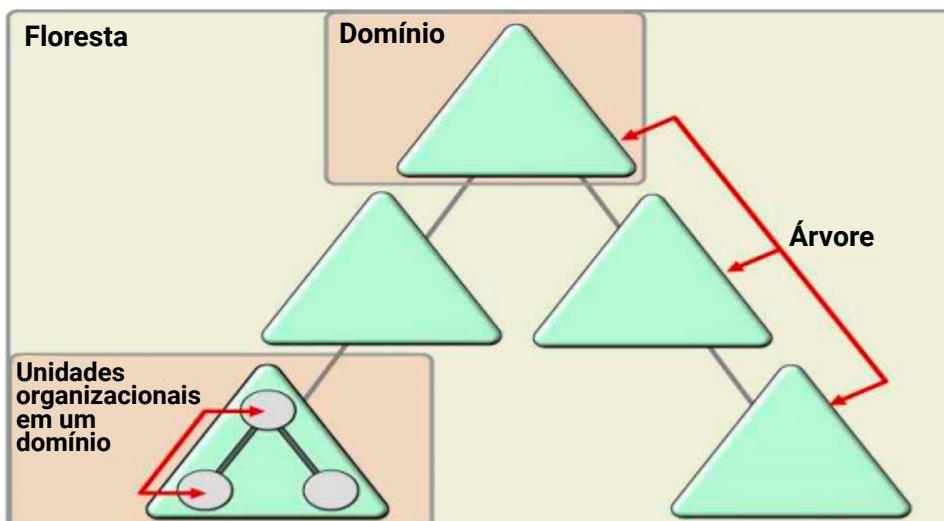


- Cada **objeto** tem um conjunto de propriedades que também é armazenado no AD.
  - Os **elementos que definem as permissões de controle de acesso sobre os objetos** no Active Directory incluem descritores de segurança, herança de objetos e autenticação do usuário.
- **Unidades Organizacionais** (UOs ou OUs em Inglês): no AD, são os objetos do tipo contêineres, que podem conter objetos usuários, grupos, computadores, impressoras, aplicativos, diretrivas de segurança e compartilhamento de arquivos. Uma UO pode ainda conter outra UO.
- Um **domínio** provê área de atuação e limites administrativos e de segurança. Permite que os recursos sejam agrupados **logicamente**, de acordo com os critérios de cada **local**.
  - Com a utilização de **domínios**, podemos fazer com que nossa rede reflita a estrutura de uma empresa.
  - Quando utilizamos vários domínios temos o conceito de **relação de confiança**. A relação de confiança permite que os usuários de ambos os domínios acessem os recursos localizados nesses domínios.
  - Os domínios em uma árvore são unidos através de **relações de confiança transitiva bidirecional**. Uma relação de confiança significa que o Domínio A confia em B, e o Domínio B confia em A.

- **Árvore:** conjunto de um ou mais domínios. Trata-se, portanto, de uma estrutura lógica mais abrangente que o domínio. A árvore é criada quando é criado um domínio. Todos os domínios da mesma árvore compartilham informações e recursos.



- **Floresta:** conjunto de uma ou mais árvores. O uso de florestas é bastante comum em grupos de empresas, em que cada uma das empresas do grupo mantém uma autonomia de identidade em relação às outras.



- O **Kerberos** versão 5, definido pela RFC 1510, é o principal protocolo de segurança e autenticação do Active Directory.
  - Estrutura hierárquica permite benefícios na organização dos dados;
  - Autenticação utiliza **Kerberos**.
- Base de dados pode ser distribuída entre vários servidores:
  - performance;
  - tolerância a falhas.
- Com o **SERVIDOR DNS** instalado, é possível instalar e configurar o AD. O AD depende totalmente do serviço DNS para a resolução de nomes dos objetos do domínio.

- Desde o Windows 2000, os nomes de domínio do AD são, geralmente, os nomes DNS (*domain name service*) completos dos domínios.
- **No Active Directory, desde o Windows Server 2008, já é possível fazer a definição de mais de uma política de senha para o mesmo domínio.** Como a alteração dessa política afetará todos os usuários relacionados ao domínio será necessária a interrupção do serviço para que seja efetivada.
- Veja a seguir alguns serviços do AD:

<b>AD DS (Domain Services)</b>	<b>Armazena informações sobre usuários, computadores, dispositivos etc.</b>
<b>AD LDS (Lightweight Directory Services)</b>	Provê praticamente a mesma funcionalidade do AD DS, mas não requer o desenvolvimento de domínios ou DCs (é mais "light").
<b>AD CS (Certificate Services)</b>	Para criação e gerenciamento de certificados de chaves públicas.
<b>AD FS (Federation Services)</b>	Criação de identidade de acesso que opera através de múltiplas plataformas (Windows ou não).
<b>AD RMS (Rights Management Services)</b>	Serviços para habilitar a criação de soluções com proteção de informação.

- Durante o processo de instalação do **Active Directory**, são criados os arquivos:

<b>Ntds.dit</b>	<b>Arquivo de banco de dados do Active Directory</b> (Nota: a extensão DIT é de <i>Directory Information Tree</i> ).
<b>Edb.log</b>	Arquivo de <i>log</i> . Armazena todas as transações feitas no AD.
<b>Edb.chk</b>	Arquivo de <i>checkpoint</i> utilizado para a recuperação ( <i>recovery</i> ) de um estado. Controla quais transações do arquivo Edb.log já foram <i>committed</i> no arquivo Ntds.dit.
<b>Res1.log e Res2.log</b>	Arquivo de reserva para assegurar que as alterações realizadas no AD possam ser gravadas na base(Ntds.dit) no caso de falta de espaço em disco.
<b>Schema.ini</b>	Inicializa o Ntds.dit durante a promoção inicial do DC (servidor membro vira DC), depois de pronto não é mais utilizado.

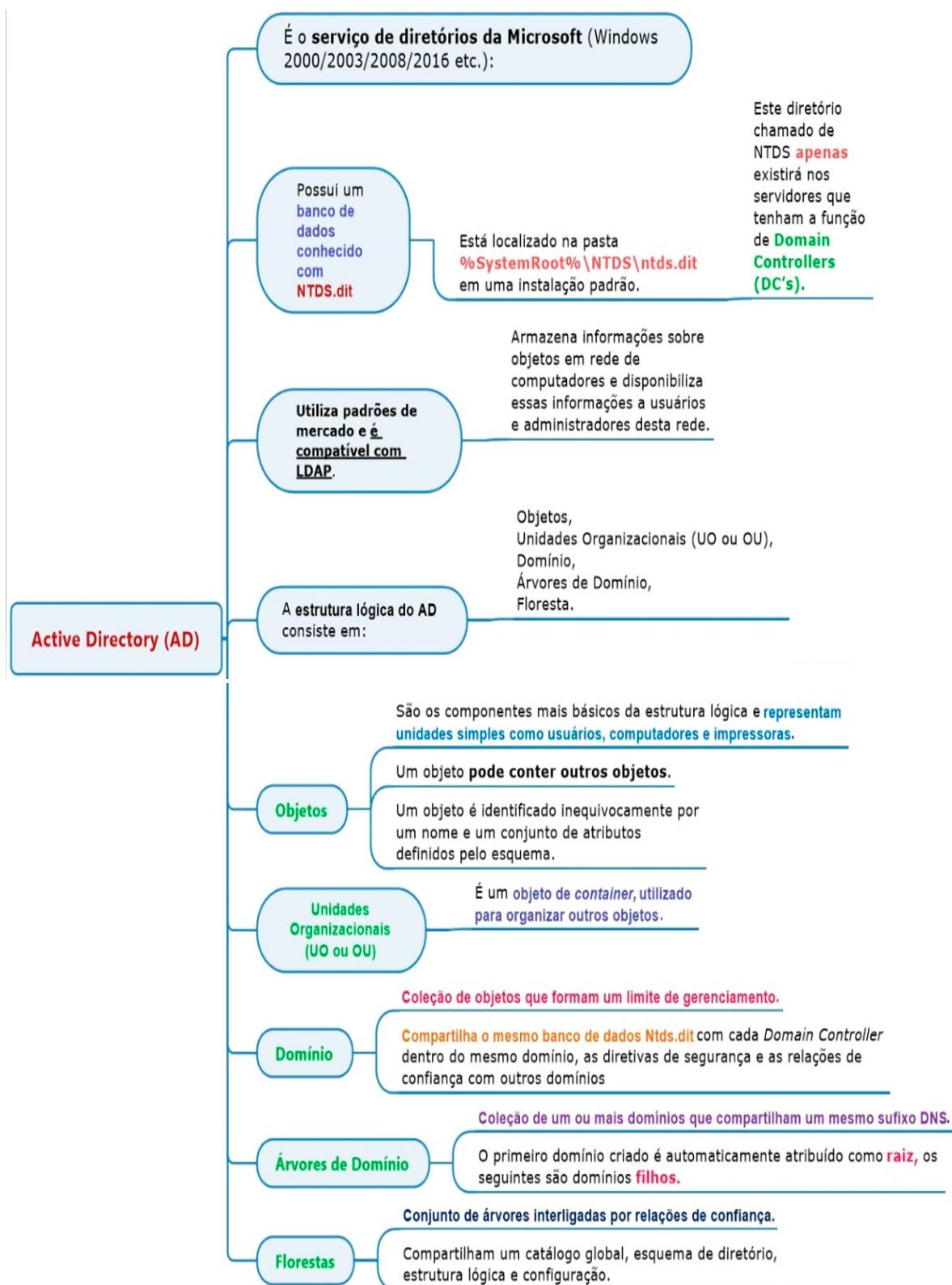


Figura. Active Directory. Fonte: Quintão (2020)

A estrutura física consiste em *Domain Controllers e Sites*.

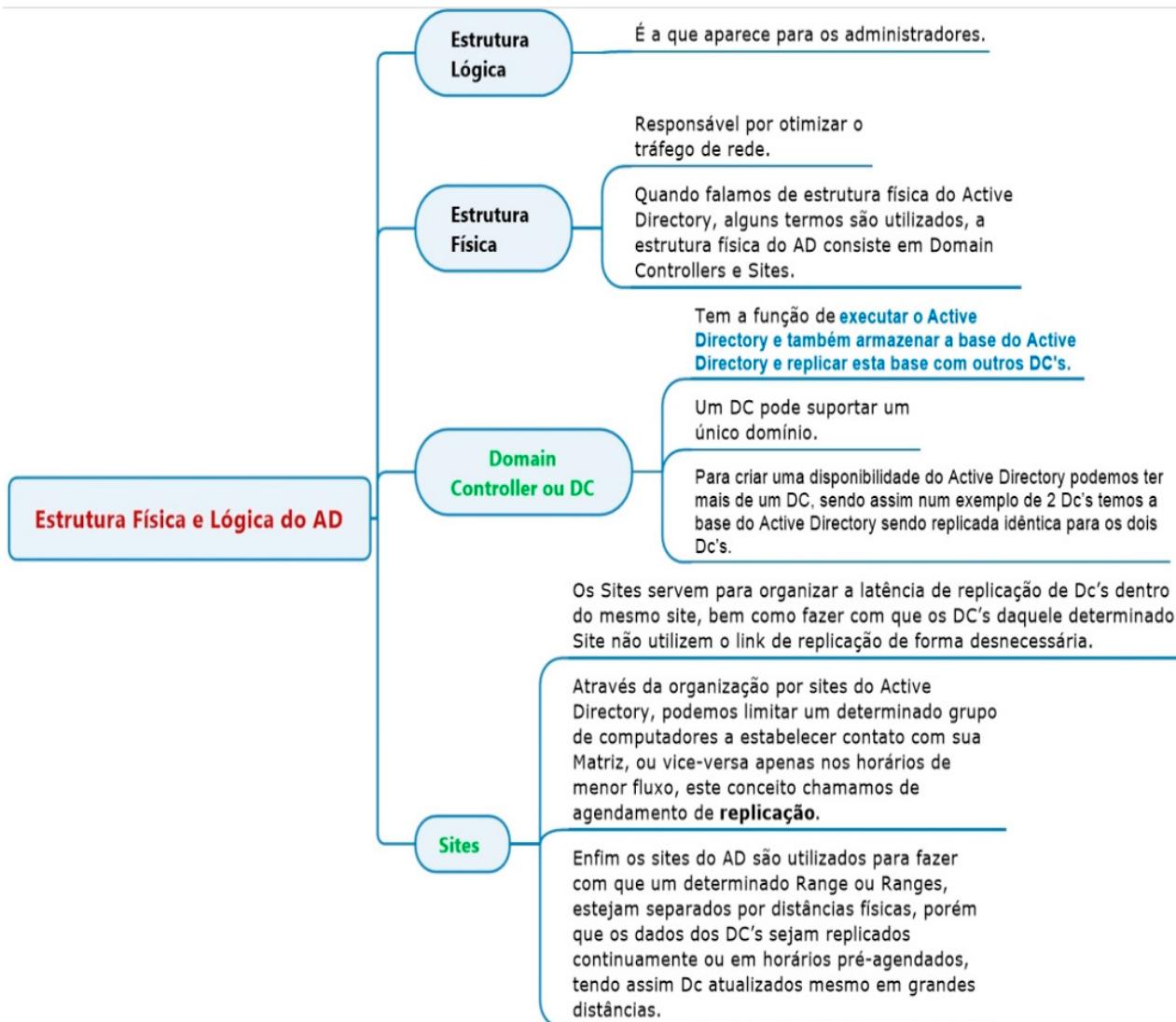


Figura. Estrutura Física e Lógica do AD. Fonte: Quintão (2020)

- A **estrutura física do AD** é totalmente independente da estrutura lógica do AD e é responsável por otimizar o tráfego de rede e manter segurança em locais físicos distintos.
- A **GPO (Group Police Object ou Diretiva de Grupo)**: funcionalidade dos Servidores Windows para controlar permissões e direitos de contas, e é gerenciada por intermédio do Gerenciamento de Diretivas de Grupo. No Windows Server 2008/2016/..., o principal modo de aplicar as configurações de diretiva em um **GPO (Group Policy Objects)** para usuários e computadores é **vinculando o GPO a um contêiner no Active Directory**.

Nesse caso, os GPOs contêm regras que podem ser vinculadas a: sites do AD, domínios e unidades organizacionais.

- Um Auditor que cuida do Active Directory (AD) de uma rede pode **realizar uma série de ações para proteger o AD**, como:

- definição do mínimo privilégio possível aos usuários do AD;
- redução dos usuários dos Grupos Administradores do domínio;
- realização de auditorias importantes (*logon*, alterações no AD, pastas etc.);
- realização de monitoramento de *logs* referentes ao AD etc.

## DIRETO DO CONCURSO

**014.** (FCC/PREFEITURA DE SÃO LUÍS-MA/AUDITOR-FISCAL DE TRIBUTOS I/TECNOLOGIA DA INFORMAÇÃO (TI)/2018)

Um Auditor que cuida do Active Directory (AD) de uma rede realizou uma ação que contraria as melhores práticas para proteger o AD. Esta ação criou oportunidade para que um invasor se inserisse no domínio e conseguisse senhas e privilégios de usuários importantes. Uma ação realizada pelo Auditor que pode ter sido a causa do problema é a

- a) ativação do usuário administrador local nas estações.
- b) definição do mínimo privilégio possível aos usuários do AD.
- c) redução dos usuários dos Grupos Administradores do domínio.
- d) realização de auditorias importantes (*logon*, alterações no AD, pastas etc.).
- e) realização de monitoramento de logs referentes ao AD.



A maioria dos ataques contra o diretório começa como ataques contra *hosts* individuais. Assim, em todas as versões do Windows atualmente no suporte base, a conta de administrador local é desabilitada por padrão, o que torna a conta inutilizável para uso em roubos de credenciais, e prejudicar a organização.

**Letra a.**

Veja a seguir mais termos relacionados ao AD:

Serviço de <b>Federação</b> do Active Directory ( <b>AD FS</b> )	Fornece tecnologias de SSO ( <i>logon único</i> ) para autenticar um usuário em vários aplicativos Web relacionados, durante uma única sessão online.
<b>Active Directory Rights Management Services</b> ( <b>AD RMS</b> )	Disponibiliza serviços que permitem a criação de soluções de proteção de informações.
Serviços de <b>Certificados</b> do Active Directory ( <b>AD CS</b> )	Fornece serviços para a criação e o gerenciamento de certificados de chave pública utilizados em sistemas que empregam tecnologias de chave pública.

<b>Active Directory Domain Services ou Serviço de Domínio do Active Directory (AD DS)</b>	Armazena informações sobre computadores, usuários e outros dispositivos existentes na rede. O AD DS ajuda os administradores a gerenciar com segurança essas informações e facilita o compartilhamento de recursos e a colaboração entre usuários.
<b>Active Directory Lightweight Directory Access Protocol (AD LDS)</b>	Serviço de diretório que utiliza o protocolo LDAP, e <b>fornecer um serviço de diretório leve e flexível</b> , sem algumas restrições presentes no domínio AD DS. Não requer a implantação de domínios ou controladores de domínio.

## DIRETO DO CONCURSO

**015. (FCC/TRT - 6ª REGIÃO (PE)/ANALISTA JUDICIÁRIO/TECNOLOGIA DA INFORMAÇÃO/2018)**

No Active Directory do Windows Server 2012, o serviço que provê as funcionalidades para a criação de domínios e repositórios de dados para o armazenamento de informações sobre objetos na rede é o AD

- a) DS.
- b) LDS.
- c) CS.
- d) RMS.
- e) FS.



O **Serviço de Domínio do Active Directory (AD DS)** armazena informações sobre **computadores, usuários e outros dispositivos existentes na rede** e ajuda os administradores a gerenciar com segurança essas informações, facilitando o compartilhamento de recursos e a colaboração entre usuários.

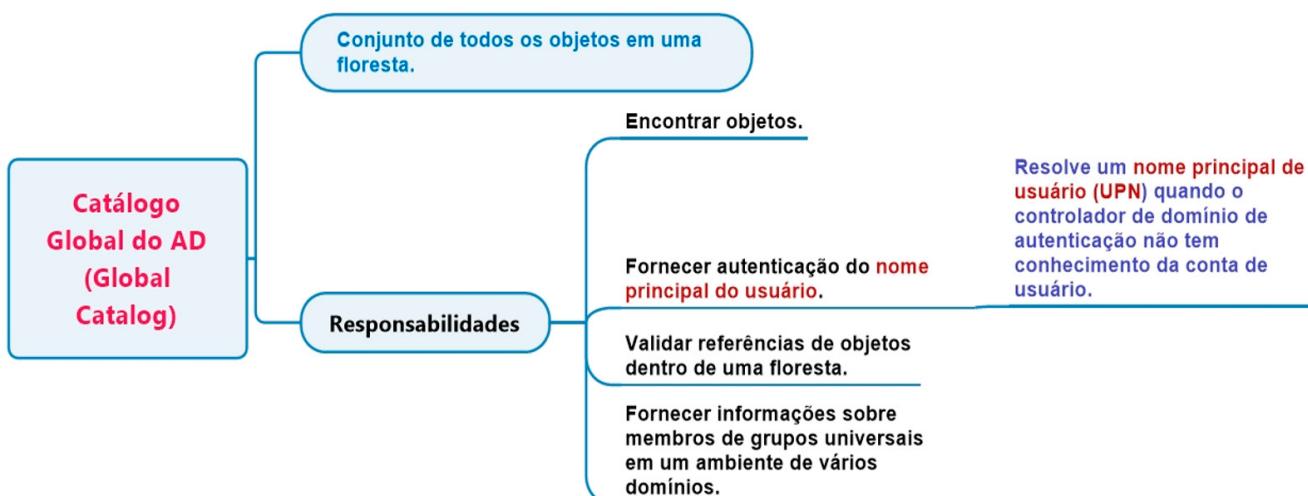
**Letra a.**

- **Global Catalog (Catálogo Global do AD):**
  - Armazena o **conjunto de todos os objetos em uma floresta dos Serviços de Domínio Active Directory (AD DS)**.
  - É um controlador de domínio que armazena uma cópia completa de todos os objetos no diretório para seu domínio de host e uma **cópia parcial, somente leitura de todos os objetos para todos os outros domínios** na floresta.
  - Fornece a autenticação do nome principal do usuário para o controlador de domínio no processo de *logon*.
  - **Um Domain Controller pode ser um Global Catalog, mas um Global Catalog sempre é um Domain Controller.**

## DIRETO DO CONCURSO

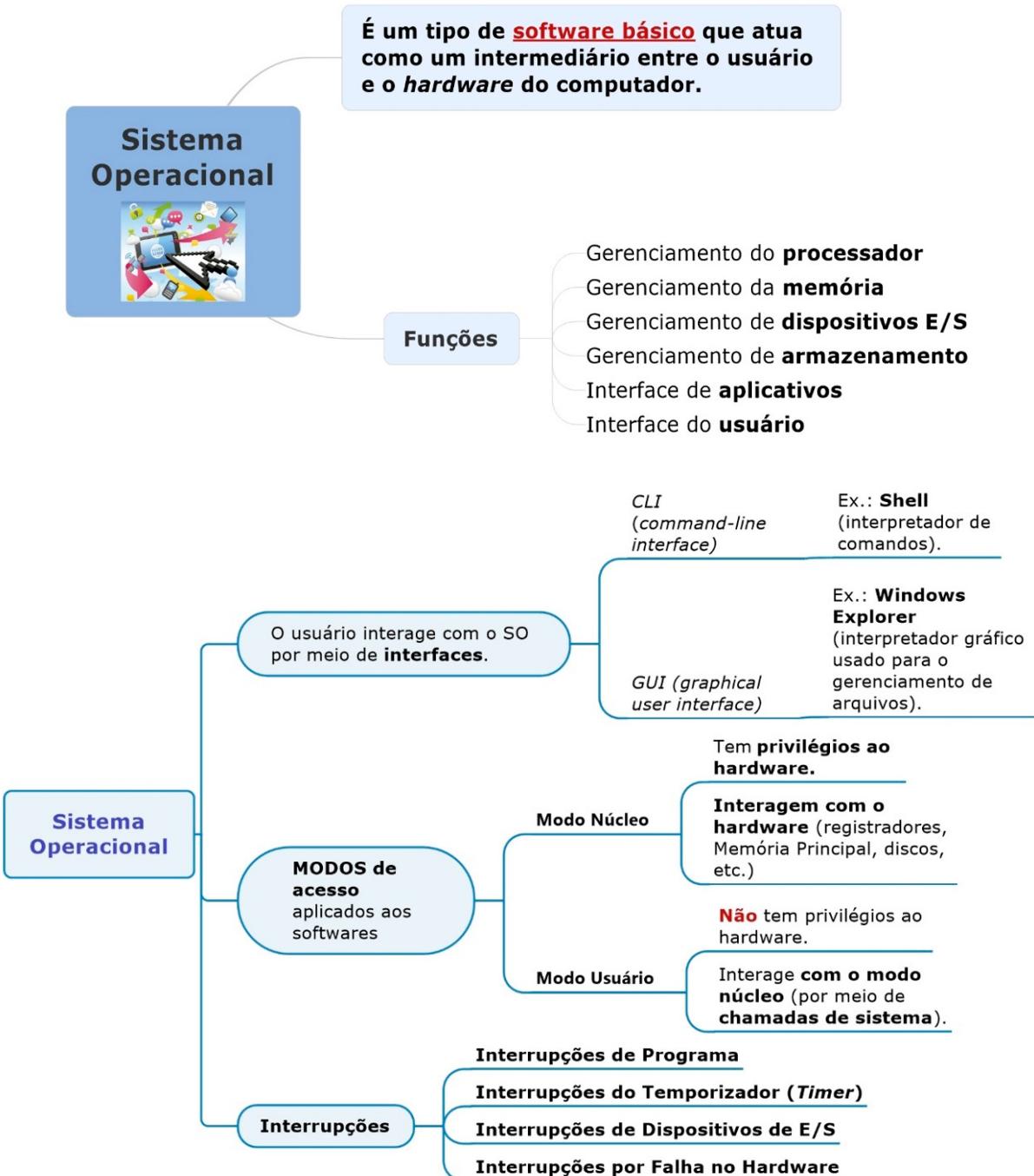
**016.** (FCC/TRT - 24ª REGIÃO (MS)/TÉCNICO JUDICIÁRIO/TECNOLOGIA DA INFORMAÇÃO/2017) Ao realizar a instalação do Active Directory Domain Services (AD DS) no Windows Server 2008, um Técnico incluiu a opção Global Catalog Server com o intuito de

- a) aumentar a velocidade de busca de Nomes de Domínio (DNS) por meio do recurso de cache do catálogo global.
- b) manter uma listagem dos servidores de autenticação da internet disponíveis para uso no domínio local.
- c) fornecer a autenticação do nome principal do usuário para o controlador de domínio no processo de logon.
- d) listar os identificadores de autenticidade de servidores em nuvem acessíveis para o domínio local.
- e) listar os servidores de Nomes de Domínio raiz para facilitar o processo de busca de recursos na internet.



**Letra c.**

## RESUMO



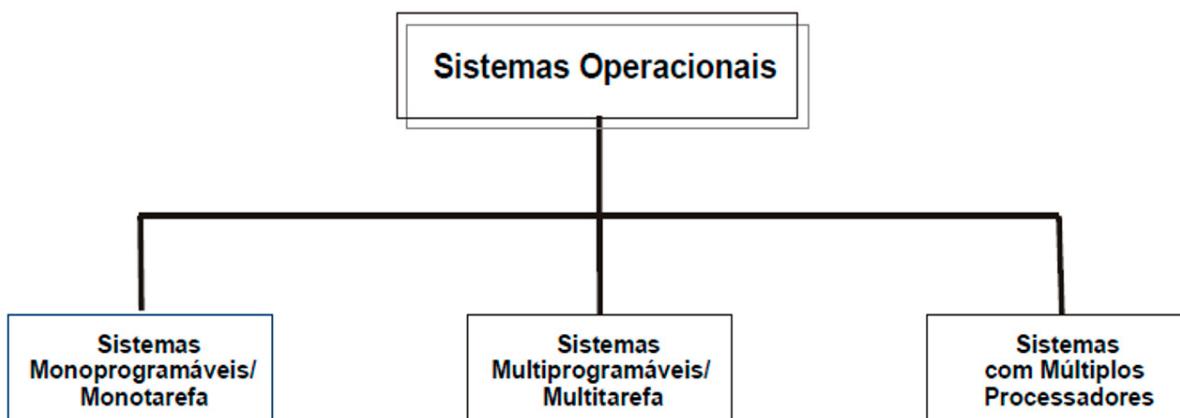
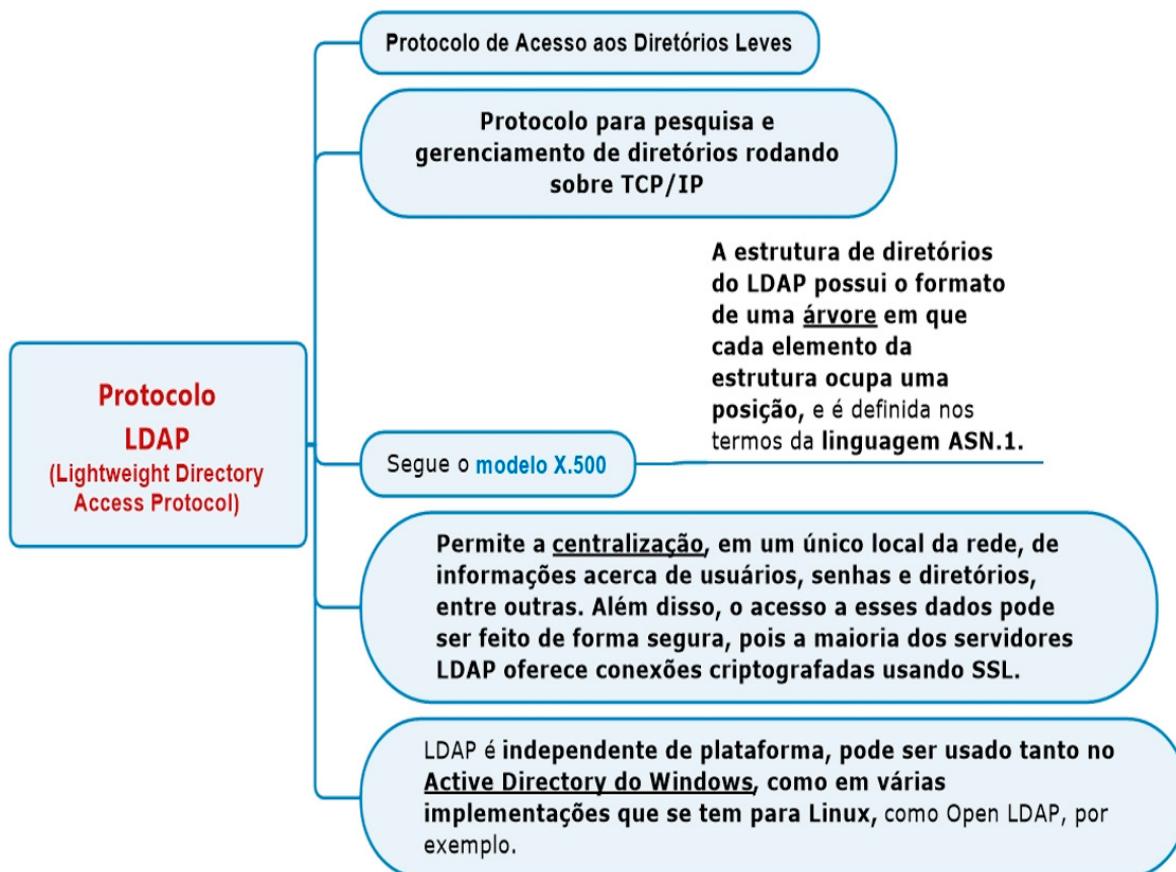


Figura -Tipos de Sistemas Operacionais

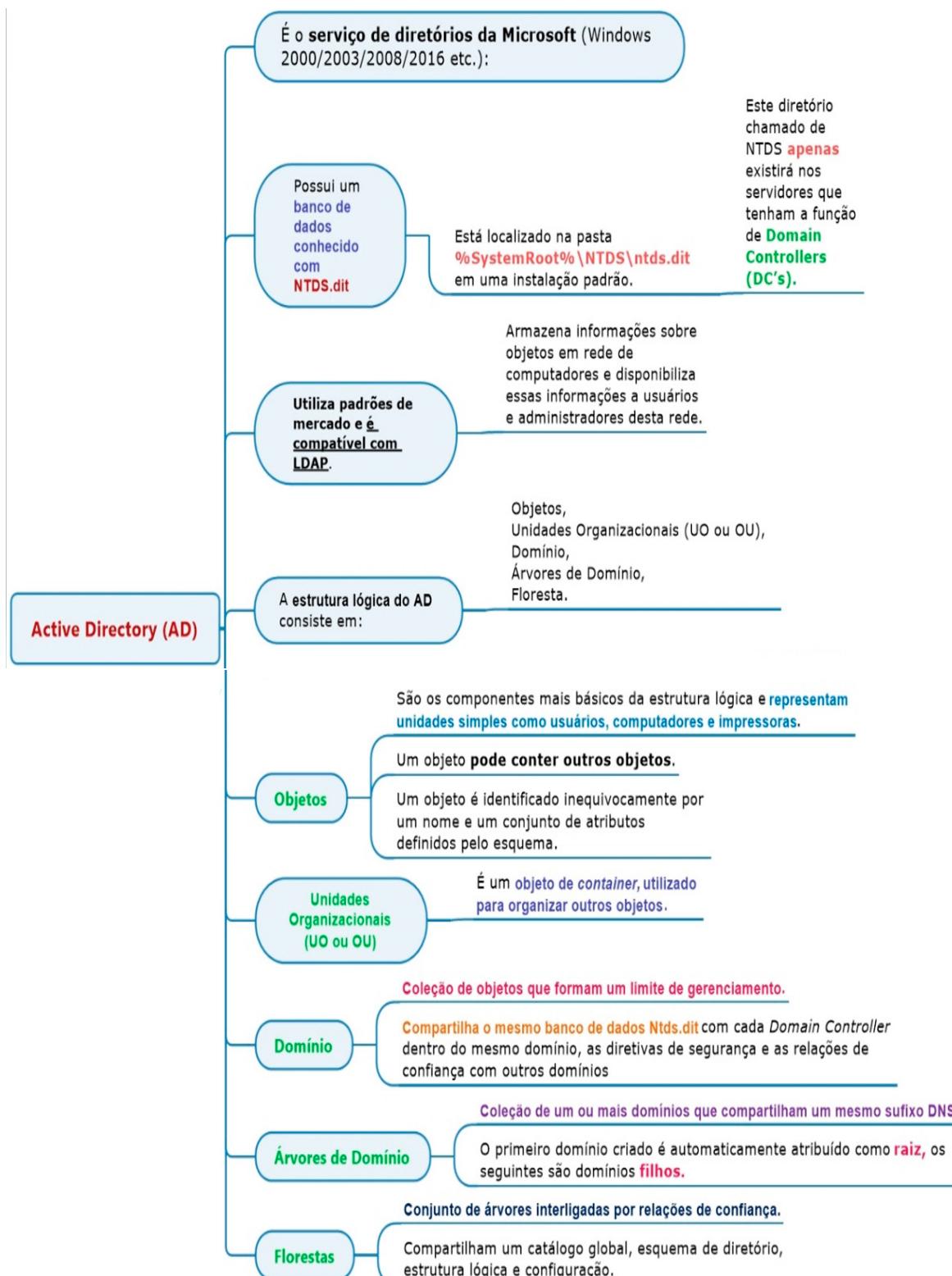
## Tipos de Sistemas Operacionais

- Sistemas Monoprogramáveis/Monotarefa
- Sistemas Multiprogramáveis/Multitarefa
  - Sistemas Batch
  - Sistemas de Tempo Compartilhado
  - Sistemas de Tempo Real
- Sistemas com Múltiplos Processadores
- Sistemas Fortemente Acoplados
  - Sistemas Simétricos
  - Sistemas Assimétricos
- Sistemas Fracamente Acoplados
  - Sistemas Operacionais de Rede
  - Sistemas Operacionais Distribuídos

## LDAP



## Active Directory



## QUESTÕES COMENTADAS NA AULA

**001.** (FCC/TRT- 16ª REGIÃO (MA)/ANALISTA JUDICIÁRIO - TECNOLOGIA DA INFORMAÇÃO/2014) Um Sistema Operacional (SO) realiza o gerenciamento <sup>I</sup> que inclui o fornecimento do sistema de arquivos para a representação de arquivos e diretórios e o gerenciamento do espaço em dispositivos com grande capacidade de armazenamento de dados.

<sup>II</sup>, que são a unidade básica de trabalho do SO. Isso inclui a sua criação, sua exclusão e o fornecimento de mecanismos para a sua comunicação e sincronização.

<sup>III</sup>, controlando que partes estão sendo usadas e por quem. Além disso, é responsável pela alocação e liberação dinâmica de seu espaço.

As lacunas I, II e III são, correta e respectivamente, preenchidas por:

- a) de armazenamento - de processos - de memória
- b) em memória secundária - de serviços - em memória principal
- c) de arquivos - de barramentos - de discos
- d) de discos – de threads- de cache
- e) de I/O - de tempos de CPU - de RAM

**002.** (CESPE/STF/ANALISTA JUDICIÁRIO DE TI/2013) No que se refere aos sistemas operacionais, julgue os itens subsecutivos.

No modo de operação do processador denominado modo usuário, instruções privilegiadas não podem ser executadas. Se houver tentativa de execução nesse caso, o hardware automaticamente gerará a interrupção e acionará o sistema operacional.

**003.** (CESGRANRIO/PETROBRAS/ANALISTA DE SISTEMAS JÚNIOR – INFRA/2012) O mecanismo pelo qual programas dos usuários solicitam serviços ao núcleo do sistema operacional é denominado

- a) biblioteca do sistema
- b) chamada do sistema
- c) editor de ligação
- d) shell de comandos
- e) ligação dinâmica

**004.** (ESAF/MDIC/ANALISTA DE COMÉRCIO EXTERIOR/2012) Em sistemas multiprogramáveis, são estados do processo:

- A. Execução (executing). Disponível (available). Espera (log).
- b) Execução (running). Pronto (ready). Espera (wait).
- c) Direção (driving). Ponto (float). Espera (waste).
- d) Execução (running). Inserção (insert). Avaliação (control).
- e) Pronta execução (ready running). Compartilhamento (sharing). Espera (wait).

**005.** (FGV/SEFAZ-RJ/AUDITOR-FISCAL DA RECEITA ESTADUAL - PROVA 1/2011) No contexto dos sistemas operacionais Windows, os mais modernos operam com base em um recurso, baseado em dois aspectos, descritos a seguir:

I – É fornecido um tempo às aplicações em execução, sendo o controle da CPU feito pelo próprio sistema operacional. Dessa forma, independentemente do fato de uma dada aplicação ter ou não terminado o que estava fazendo, esgotado seu time slice, o sistema operacional retoma o controle da CPU e o repassa para o próximo da fila.

II – Cada aplicação “roda” em um espaço próprio, de modo que, em caso de problemas, a aplicação em pane é finalizada, permanecendo as demais ativas em execução.

Esse recurso é conhecido por

- a) Multitarefa Preemptiva.
- b) Segmentação Paginada.
- c) Escalonamento Round Robin.
- d) Multiprocessamento Particionado.
- e) Multiprogramação com Swapping.

**006.** (FCC/TRE-RN/ANALISTA JUDICIÁRIO/ANALISTA DE SISTEMAS/2011) Nos sistemas operacionais, múltiplas execuções que ocorrem no mesmo ambiente do processo com um grande grau de independência uma da outra é uma característica do modelo de processo que contempla o conceito de

- a) bus.
- b) switch.
- c) thread.
- d) disk array.
- e) split-cylinder.

**007.** (CESPE/STF-TÉCNICO JUDICIÁRIO DE TI/2013) No que se refere aos sistemas operacionais, julgue os itens subsequentes.

No contexto do sistema operacional, processo é uma abstração que reúne uma série de atributos, entre os quais estão o espaço de endereçamento, as permissões de acesso, as quotas e o fluxo de execução. Fluxo de execução é uma thread e multithreading é a associação de vários fluxos de execução a um único processo.

**008.** (CESPE/STF-TÉCNICO JUDICIÁRIO DE TI/2013) No que se refere aos sistemas operacionais, julgue os itens subsequentes.

Embora a técnica de paginação auxilie a multitarefa, permitindo a alocação de vários processos em áreas diferentes de memória, ela não elimina o problema de que tal alocação deva ser contígua.

**009.** (CESGRANRIO/BNDES/ANALISTA DE SUPORTE/2008) O uso de memória secundária (de menor velocidade, comparada com a memória principal) como memória virtual acaba ocasionando perda de desempenho. Neste contexto, supondo que um computador da linha PC, monoprocessado, admite expansibilidade de seus componentes, a solução que atualmente oferece melhor relação custo-benefício para atenuar o problema é:

- a) adicionar um disco rígido de maior capacidade.
- b) adicionar memória RAM.
- c) aumentar o tamanho do arquivo de troca (swap) do sistema operacional.
- d) substituir o processador por um mais veloz.
- e) trocar o computador todo por um mais moderno, pois não há solução para o problema.

**010.** (CESGRANRIO/CASA DA MOEDA/ANALISTA DE NÍVEL SUPERIOR - BANCO DE DADOS/2009) Determinado sistema operacional apresenta a seguinte lista de segmentos de memória disponíveis, em sequência: 10 KB, 20 KB, 40 KB, 80 KB, 160 KB.

Nesse momento, um processo solicita alocação de memória para 16 KB. Considerando-se o uso do algoritmo first fit, que segmento será utilizado na alocação?

- a) 10 KB.
- b) 20 KB.
- c) 40 KB.
- d) 80 KB.
- e) 160 KB.

**011.** (FCC/TRF-5A REGIÃO/ANALISTA/2003) O sistema de arquivos utilizado na formatação dos discos e reconhecido por TODAS as versões do sistema operacional Windows é denominado:

- a) EXT2;
- b) FAT 16;
- c) FAT 32;
- d) NTFS;
- e) HPFS.

**012.** (CESPE/TCU/2009) Com relação aos sistemas operacionais Linux e Windows, julgue os itens a seguir.

[O NTFS, como o UFS, é um sistema de arquivos hierarquizado com arquivos e diretórios, em que cada arquivo tem um proprietário, mas não pertence a um grupo. O acesso não é regulado por um conjunto de bits de permissão, mas por listas de acesso, implementando um modelo discricionário.]

**013.** (CESGRANRIO/TRANSPETRO/ANALISTA DE SISTEMAS JÚNIOR – SOFTWARE/2011) Os Sistemas Operacionais estão sujeitos a um fenômeno denominado deadlock. Para que uma situação de deadlock seja criada, as seguintes condições devem acontecer simultaneamente:

- a)** exclusão mútua (mutual exclusion), monopolização de recursos (hold and wait), não preempção (no preemption) e espera circular (circular wait).
- b)** exclusão mútua (mutual exclusion), transferência excessiva de páginas (thrashing), superposição de processos (process overlapping) e espera circular (circular wait).
- c)** transferência excessiva de páginas (thrashing), superposição de processos (process overlapping), monopolização de recursos (hold and wait) e não preempção (no preemption).
- d)** exclusão mútua (mutual exclusion), monopolização de recursos (hold and wait), superposição de processos (process overlapping) e falha de escalonamento (scheduling fail)
- e)** transferência excessiva de páginas (thrashing), não preempção (no preemption), espera circular (circular wait) e falha de escalonamento (scheduling fail).

**014.** (FCC/PREFEITURA DE SÃO LUÍS-MA/AUDITOR-FISCAL DE TRIBUTOS I/TECNOLOGIA DA INFORMAÇÃO (TI)/2018) Um Auditor que cuida do Active Directory (AD) de uma rede realizou uma ação que contraria as melhores práticas para proteger o AD. Esta ação criou oportunidade para que um invasor se inserisse no domínio e conseguisse senhas e privilégios de usuários importantes. Uma ação realizada pelo Auditor que pode ter sido a causa do problema é a

- a)** ativação do usuário administrador local nas estações.
- b)** definição do mínimo privilégio possível aos usuários do AD.
- c)** redução dos usuários dos Grupos Administradores do domínio.
- d)** realização de auditorias importantes (logon, alterações no AD, pastas etc.).
- e)** realização de monitoramento de logs referentes ao AD.

**015.** (FCC/TRT - 6ª REGIÃO (PE)/ANALISTA JUDICIÁRIO/TECNOLOGIA DA INFORMAÇÃO/2018) No Active Directory do Windows Server 2012, o serviço que provê as funcionalidades para a criação de domínios e repositórios de dados para o armazenamento de informações sobre objetos na rede é o AD

- a)** DS.
- b)** LDS.
- c)** CS.
- d)** RMS.
- e)** FS.

**016.** (FCC/TRT - 24ª REGIÃO (MS)/TÉCNICO JUDICIÁRIO/TECNOLOGIA DA INFORMAÇÃO/2017) Ao realizar a instalação do Active Directory Domain Services (AD DS) no Windows Server 2008, um Técnico incluiu a opção Global Catalog Server com o intuito de

- a) aumentar a velocidade de busca de Nomes de Domínio (DNS) por meio do recurso de cache do catálogo global.
- b) manter uma listagem dos servidores de autenticação da internet disponíveis para uso no domínio local.
- c) fornecer a autenticação do nome principal do usuário para o controlador de domínio no processo de logon.
- d) listar os identificadores de autenticidade de servidores em nuvem acessíveis para o domínio local.
- e) listar os servidores de Nomes de Domínio raiz para facilitar o processo de busca de recursos na internet.

## QUESTÕES DE CONCURSO

### Sistemas Operacionais

**017.** (CESPE/POLÍCIA FEDERAL/PERITO CRIMINAL FEDERAL - ÁREA 3/2018) A respeito de sistemas operacionais e de aplicativos de edição de textos e planilhas, julgue o item a seguir. A técnica de swapping consiste em transferir temporariamente um processo da memória para o disco do computador e depois carregá-lo novamente em memória.



**Troca (Swapping):** consiste na troca dos programas (não usados) entre a RAM e o disco rígido. Em outras palavras, é uma técnica que permite a um computador executar programas e manipular arquivos de dados maiores que a memória principal. O sistema operacional copia o máximo de dados possível na memória principal e deixa o restante no disco. Quando o sistema operacional precisa de dados do disco, ele troca uma parte dos dados (chamados de página ou segmento) na memória principal por uma parte dos dados no disco

**Certo.**

**018.** (CESPE/ABIN/SUPORTE A REDE DE DADOS/2010) No contexto de sistemas operacionais, semáforos são tipos de variáveis que podem ser verificadas e alteradas em instruções atômicas, ou seja, sem possibilidades de interrupções. Esse tipo de variável é empregado em tarefas como o compartilhamento de recursos entre processos.



Um **semáforo** é uma variável atômica, ou seja, a mudança de estado tem que ser completa e sem interrupções. Esse tipo de variável (semáforo) é empregado em tarefas envolvendo por exemplo o compartilhamento de recursos entre processos.

**Certo.**

**019.** (CESPE/STF/TÉCNICO JUDICIÁRIO/APOIO ESPECIALIZADO/TECNOLOGIA DA INFORMAÇÃO/2013) Em um algoritmo de escalonamento FIFO, os processos são executados na mesma ordem que chegam à fila. Quando um processo do tipo *cpu-bound* está na frente da fila, todos os processos devem esperá-lo terminar seu ciclo de processador.



O algoritmo de escalonamento **FIFO** (*first-in, first-out*, ou primeiro a entrar, primeiro a sair) em que o primeiro processo que requer a CPU é o primeiro para o qual a CPU é alocada é **não preemptivo**.

O **tamanho dos ciclos de CPU e de E/S** varia de processo para processo. Dessa forma, podemos identificar dois tipos de processos. Uns que requisitam mais E/S e têm ciclos de CPU

muito pequenos e outros que gastam mais tempo executando instruções e têm ciclos de E/S pequenos:

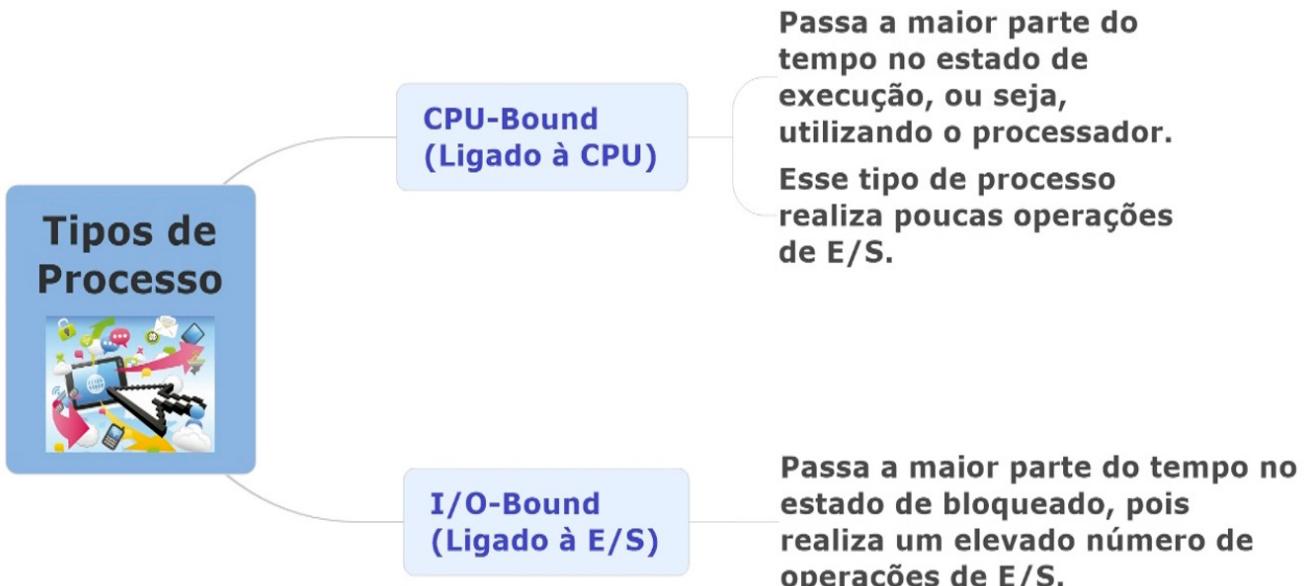


Figura. Tipos de Processos. Fonte: Quintão (2020)

Conforme visto, o processo *CPU-bound* passa a maior parte do tempo utilizando o processador, assim, raramente é bloqueado. Dessa forma, os demais processos devem esperar acabar seu ciclo de uso do processador (**quantum**).

**Certo.**

**020.** (CESPE/TRE-PI/TÉCNICO JUDICIÁRIO/OPERAÇÃO DE COMPUTADORES/2016) A respeito das características do algoritmo de escalonamento SPF (shortest process first), assinale a opção correta.

- Os processos são executados na ordem em que chegam à fila de espera e executados até o final, sem nenhum evento preemptivo.
- No SPF, um processo recém-chegado e em espera, cujo tempo estimado de execução completa seja menor, provoca a preempção de um processo em execução que apresente tempo estimado de execução completa maior.
- O SPF favorece processos longos em detrimento dos mais curtos. Estes, ao chegarem à fila de espera, são obrigados a aguardar a conclusão dos processos longos que já estiverem em andamento, para, então, entrar em execução.
- Os processos são despachados na ordem em que são colocados em espera e recebem uma quantidade limitada de tempo do processador para execução; além disso, são interrompidos caso sua execução não se conclua dentro do intervalo de tempo delimitado.
- O escalonador seleciona o processo que estiver à espera e possuir o menor tempo de execução estimado e o coloca em execução até a sua conclusão.



O problema do **escalonamento** é definido da seguinte forma: dados os vários processos que devem ser executados por um computador, deve-se definir qual a ordem em que os processos serão executados.

O algoritmo de escalonamento SPF (*shortest process first*), também conhecido por "**Menor Job Primeiro**" (ou SJF – *Shortest-Job-First*), é um algoritmo que lida com os processos de acordo com a extensão dos seus ciclos de CPU. Quando a CPU está disponível, ela é alocada ao processo que possui o próximo ciclo de CPU menor.

**Letra e.**

**021.** (CESPE/TRE-PI/TÉCNICO JUDICIÁRIO - OPERAÇÃO DE COMPUTADORES/2016) O componente central de um sistema operacional, que determina o local da memória onde deverá ser colocado o código de um novo processo chamado para ser executado por um processo pai, lido de um arquivo previamente armazenado em um dispositivo de entrada e saída, que, por sua vez, está conectado à rede local, é denominado

- a) gerenciador de sistema de arquivos.
- b) gerenciador de comunicação interprocessos.
- c) gerenciador de memória.
- d) escalonador de processos.
- e) gerenciador de entrada e saída.



Entre os componentes centrais do sistema operacional estão:

Gerenciador de sistema de arquivos

Organiza coleções nomeadas de dados em dispositivos de armazenamento e fornece uma interface para acessar os dados nesses dispositivos.

Gerenciador de comunicação interprocessos (IPC)

Permite que os processos se comuniquem uns com os outros.

Gerenciador de memória

Determina quando e como a memória é alocada aos processos e o que fazer quando a memória principal estiver cheia.

É esse o componente que irá determinar o local da memória onde deverá ser colocado o código de um novo processo chamado para ser executado por um processo pai, lido de um arquivo previamente armazenado em um dispositivo de entrada e saída, que, por sua vez, está conectado à rede local é denominado gerenciador de memória.

Escalonador de processos

Realiza funções como criação e deleção de processos, escalonamento de processos, mecanismos de comunicação e sincronização entre processos, entre outros.

Gerenciador de Entrada/Saída

Atende às solicitações de entrada/saída para dispositivos de hardware respectivamente.

Essa questão teve o gabarito alterado de A para C, no gabarito definitivo!

GABARITO PRELIMINAR	GABARITO DEFINITIVO
A	C

**Letra c.**

**022.** (CESPE/2016/TRE-PI/TÉCNICO JUDICIÁRIO - OPERAÇÃO DE COMPUTADORES) Assinale a opção correta acerca da estratégia de gerenciamento de memória de busca antecipada.

- a) O sistema carrega parte de um programa, ou de dados, da memória principal que ainda não foi referenciada.
- b) O sistema determina previamente que um programa ou dados sejam carregados no primeiro espaço disponível da memória.
- c) O sistema determina previamente que programas ou dados sejam carregados no espaço de memória em que melhor couberem.
- d) O sistema determina que, ao se verificar a alta ocupação da memória, parte do conteúdo carregado seja encontrada e removida da memória para dar lugar a novos carregamentos.
- e) O sistema posiciona a próxima porção do programa ou de dados na memória principal quando um programa em execução os referencia.



A **política de busca de páginas** determina quando uma página deve ser carregada para a memória. Basicamente, tem-se 2 estratégias: paginação por demanda e paginação antecipada.

- **Na paginação por demanda (demand paging)**, as páginas dos processos são transferidas da memória secundária para a principal apenas **quando referenciadas**.
- **Na paginação antecipada (anticipatory paging ou prepaging)**, o sistema **carrega para a memória principal, além da página referenciada, outras páginas que podem ou não ser necessárias** ao processo ao longo do seu processamento. Se imaginarmos que o programa está armazenado sequencialmente no disco, existe uma grande economia de tempo em levar um conjunto de páginas da memória secundária, ao contrário de carregar uma de cada vez. No entanto, caso o processo não precise das páginas carregadas antecipadamente, o sistema terá perdido tempo e ocupado memória principal desnecessariamente.

Fonte: Machado, F. B; Maia, L. P. Arquitetura de sistemas operacionais. Rio de Janeiro:LTC, 2007.

**Letra a.**

## Active Directory (AD)

**023.** (CESPE/ABIN/OFICIAL TÉCNICO DE INTELIGÊNCIA - ÁREA 8/2018) Julgue o próximo item, que trata de administração de serviços de diretório, de serviço de monitoramento e gerenciamento de redes e de serviços de rede.

No Active Directory, apesar da exigência de uma política de senha única por domínio, a alteração dessa política não exige a interrupção do serviço para ser efetivada.



No Active Directory, desde o Windows Server 2008, já é possível fazer a definição de mais de uma política de senha para o mesmo domínio. Como a alteração dessa política afetará todos os usuários relacionados ao domínio será necessária a interrupção do serviço para que seja efetivada.

**Errado.**

**024.** (CESPE/TRE-BA/TÉCNICO JUDICIÁRIO/OPERAÇÃO DE COMPUTADORES/2017) Na estrutura de confinamento hierárquica do serviço de diretório Active Directory, o componente que funciona como um limite de segurança para uma organização e define o escopo de autoridade para os administradores é denominado

- a) esquema.
- b) floresta.
- c) domínio.
- d) unidade organizacional.
- e) catálogo global.



Conforme destaca [https://docs.microsoft.com/pt-br/previous-versions/windows/server/hh831484\(v=ws.11\)?redirectedfrom=MSD](https://docs.microsoft.com/pt-br/previous-versions/windows/server/hh831484(v=ws.11)?redirectedfrom=MSD):

- **a floresta funciona como um limite de segurança para uma organização e define o escopo de autoridade para administradores.**
- Por padrão, uma floresta contém um único domínio, que é conhecido como o domínio raiz da floresta.
- Domínios adicionais podem ser criados na floresta para fornecer particionamento de dados do AD DS, o que permite que as organizações repliquem dados somente onde for necessário. Isso possibilita que o AD DS seja dimensionado globalmente em uma rede que tem largura de banda disponível limitada. Um domínio Active Directory também suporta várias outras funções principais relacionadas à administração, incluindo identidade de usuário em âmbito de rede, autenticação e relações de confiança.

**Obs.:** Uma floresta é um conjunto de árvores interligadas por relações de confiança. Compartilham um catálogo global, esquema de diretório, estrutura lógica e configuração.

**Letra b.**

**025.** (CESPE/TCE-PA/AUDITOR DE CONTROLE EXTERNO - ÁREA INFORMÁTICA/ANALISTA DE SEGURANÇA/2016) Acerca dos diretórios de serviços LDAP e AD (Active Directory), julgue o item que se segue.

O AD, além de armazenar, em seu banco de dados, objetos como usuários, membros de grupos e relações de confiança, disponibiliza alguns serviços, como, por exemplo, a autenticação de usuários.



O **Active Directory (AD)** é o **serviço de diretórios do Windows** (a partir da versão 2000). Ele identifica todos os recursos disponíveis em uma rede, mantendo suas informações, como: contas de usuários, grupos, políticas de segurança etc. em um banco de dados. Os recursos (impressoras, computadores, dentre outros) ficam disponíveis para usuários e aplicações.

Dentre as funções do AD merecem destaque:

- fornecer autenticação do nome principal do usuário;
- replicações entre os controladores de domínio;
- pesquisa de objetos na base de dados;
- interface de programação para acesso aos objetos do diretório; etc.

**Certo.**

**026.** (CESPE/MEC/2015) Julgue o item que se segue, relativo a Active Directory, IIS e Terminal Service.

Um formato conhecido como um Active Directory com menos recursos é o AD LDS ou Active Directory Lightweight Directory Services.



Veja a seguir alguns serviços do AD:

<b>AD DS (Domain Services)</b>	Armazena informações sobre usuários, computadores, dispositivos etc.
<b>AD LDS (Lightweight Directory Services)</b>	Provê praticamente a mesma funcionalidade do AD DS, mas não requer o desenvolvimento de domínios ou DCs (é mais "light").
<b>AD CS (Certificate Services)</b>	Para criação e gerenciamento de certificados de chaves públicas.

<b>AD FS (Federation Services)</b>	Criação de identidade de acesso que opera através de múltiplas plataformas (Windows ou não).
<b>AD RMS (Rights Management Services)</b>	Serviços para habilitar a criação de soluções com proteção de informação.

**Certo.**

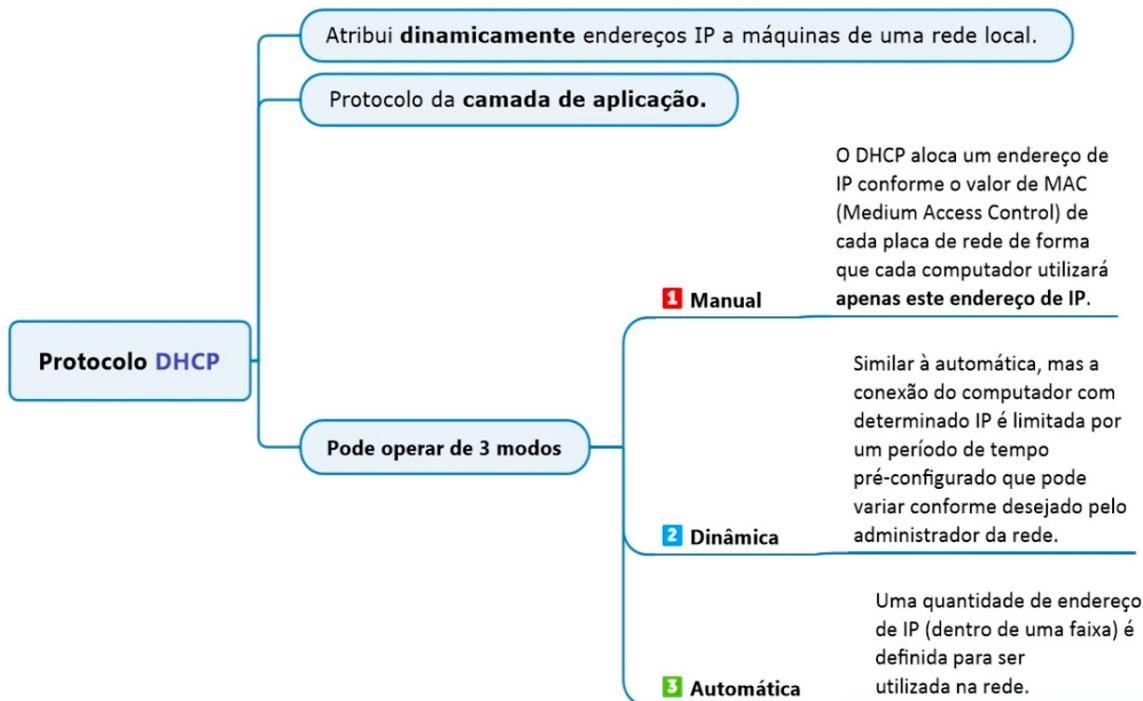
**027.** (UNIRIO/UNIRIO/2014) Active Directory está relacionado aos itens a seguir, EXCETO:

- a) Catálogo global.
- b) implementação de serviço de diretório no protocolo DHCP.
- c) Distribuição de Software Automática.
- d) Gerenciamento centralizado.
- e) Replicação automática.



**Active Directory** é o serviço de diretórios da Microsoft (Windows 2000/2003/2008/2016 etc.).

O protocolo a ser utilizado para o AD é o **LDAP (Lightweight Directory Access Protocol)**. O DHCP (*Dynamic Host Configuration Protocol* ou Protocolo de Configuração Dinâmica de Cliente) é um protocolo que atribui dinamicamente endereços IP a máquinas de uma rede local.



**Letra b.**

**028.** (CONSULPLAN/2013/TRE-MG/ANALISTA JUDICIÁRIO - ANÁLISE DE SISTEMAS) "Active Directory (AD – Diretório Ativo, em Português) representa uma nova forma de pensar a rede, não se limita a uma ferramenta de administração e possui alguns elementos que o formam."

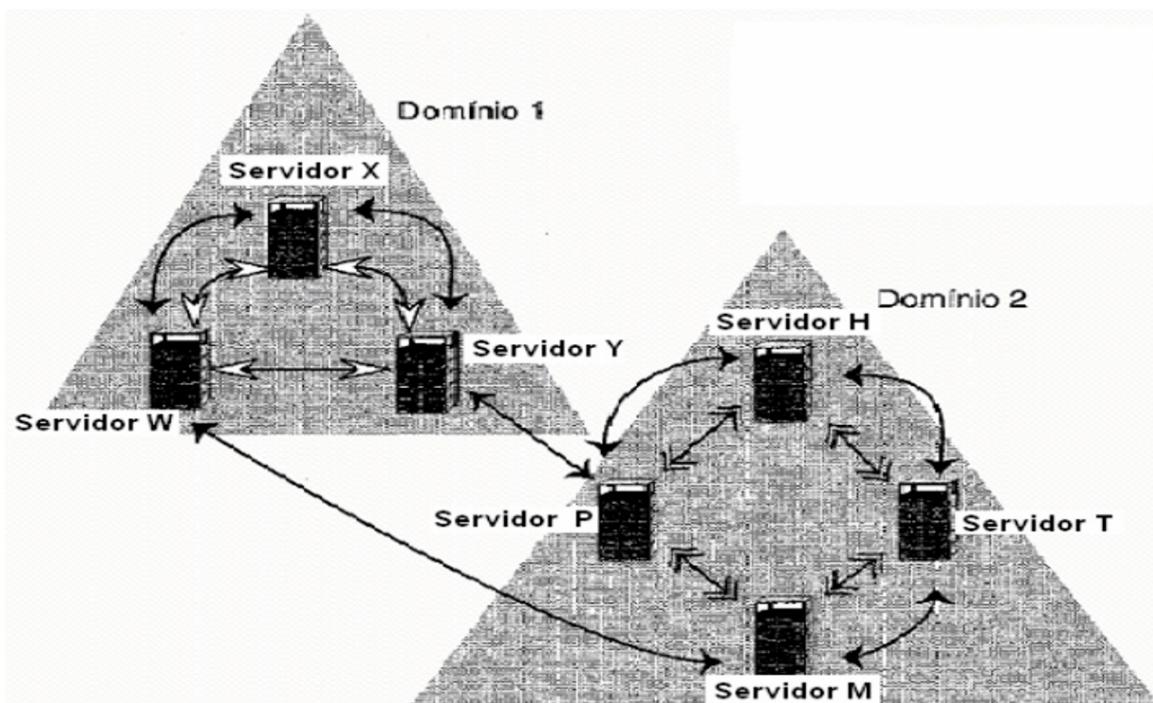
(Thompson, 2010). Que elemento do AD define o tipo de transparência que existirá entre os diferentes segmentos da rede com AD?

- a) Árvore.
- b) Domínio.
- c) Confiança
- d) Unidade organizacional.
- e) Controlador de domínio.



Dentre os elementos citados, a **relação de confiança** define o tipo de transparência que existirá entre os diferentes segmentos da rede com Active Directory (AD – Diretório Ativo, em Português).

Com a utilização de **domínios**, podemos fazer com que nossa rede reflita a estrutura de uma empresa. Quando utilizamos vários domínios temos o conceito de **relação de confiança**. A relação de confiança permite que os usuários de ambos os domínios acessem os recursos localizados nesses domínios.



**Exemplo de uma relação de confiança. (Fonte Windows Server 2003 A Bíblia)**

Fonte: [http://www.bcc.ufba.br/wp-content/uploads/2013/2006/Implantacao\\_e\\_estudo\\_de\\_um\\_sistema\\_utilizando\\_o\\_active\\_directory.pdf](http://www.bcc.ufba.br/wp-content/uploads/2013/2006/Implantacao_e_estudo_de_um_sistema_utilizando_o_active_directory.pdf)

**Letra c.**

**029.** (CESPE/ANAC/ANALISTA ADMINISTRATIVO – ÁREA 5/2012) Acerca dos conceitos, configurações, administração e gerenciamento de serviços de redes, julgue os itens subsequentes:

tes. No active directory, o conceito de floresta é utilizado para descrever um conjunto de objetos hierarquicamente organizados.



**Floresta:** Conjunto de uma ou mais árvores.

**Árvore:** Conjunto de um ou mais domínios.

**Domínio** permite que os recursos sejam agrupados logicamente, de acordo com os critérios de cada local.

**Errado.**

---

**030.** (FCC/2011/TRT-14/TÉCNICO JUDICIÁRIO TI/ADAPTADA) Os elementos que definem as permissões de controle de acesso sobre os objetos no Active Directory incluem apenas descritores de segurança e autenticação do usuário.



Os **elementos que definem as permissões de controle de acesso sobre os objetos** no Active Directory incluem descritores de segurança, herança de objetos e autenticação do usuário.

**Errado.**

---

**031.** (CONSULPLAN/COFEN/2011) Qual dos componentes a seguir NÃO faz parte da estrutura lógica do Active Directory no Windows Server?

- a) Objects.
- b) Organizational Units.
- c) Domain Forests.
- d) Domains.
- e) Forests.



**a) Objetos:** são as contas de usuário, impressoras, computadores etc. Cada **objeto** tem um conjunto de propriedades que também é armazenado no AD.

**b) Unidades Organizacionais (UOs ou OUs em Inglês):** no AD, são os objetos do tipo contêineres, que podem conter objetos usuários, grupos, computadores, impressoras, aplicativos, diretrivas de segurança e compartilhamento de arquivos. Uma UO pode ainda conter outra UO.

**c) Floresta:** **conjunto de uma ou mais árvores.** Assim, as florestas são de árvores e não de domínios! Portanto, essa assertiva é a resposta da questão.

**d) Domínios:** agrupamentos lógicos de contas e recursos. Um **domínio** provê área de atuação e limites administrativos e de segurança. Permite que os recursos sejam agrupados **logicamente**, de acordo com os critérios de cada local.

e) **Florestas:** conjuntos de árvores interligadas por relações de confiança. O uso de florestas é bastante comum em grupos de empresas, em que cada uma das empresas do grupo mantém uma autonomia de identidade em relação as outras.

Dessa forma, *Domain Forests NÃO* faz parte da estrutura lógica do Active Directory no Windows Server.

**Letra c.**

---

**032.** (CESPE/2008/SERPRO) O Active Directory, incluso no sistema operacional Microsoft Windows Server 2003, tem suporte ao LDAP (lightweight directory access protocol).



O **Active Directory** é o serviço de diretórios da Microsoft (Windows 2000/2003/etc.):

- Utiliza padrões de mercado e é compatível com LDAP.
- Autenticação utiliza **Kerberos**.
  - Estrutura hierárquica permite benefícios na organização dos dados.
- Base de dados pode ser distribuída entre vários servidores
  - Performance
  - Tolerância a falhas
- O Active Directory está localizado nas máquinas controladoras de domínio.
- O repositório AD armazena diversos objetos e recursos: Contas de usuários, Grupos, Computadores, Impressoras.
- Cada objeto tem um conjunto de propriedades que também é armazenado no AD.
- Um domínio provê área de atuação e limites administrativos e de segurança.
- Permite que os recursos sejam agrupados logicamente, de acordo com os critérios de cada local.
- **Árvore:** Conjunto de um ou mais domínios.
- **Floresta:** Conjunto de uma ou mais árvores.

**Certo.**

---

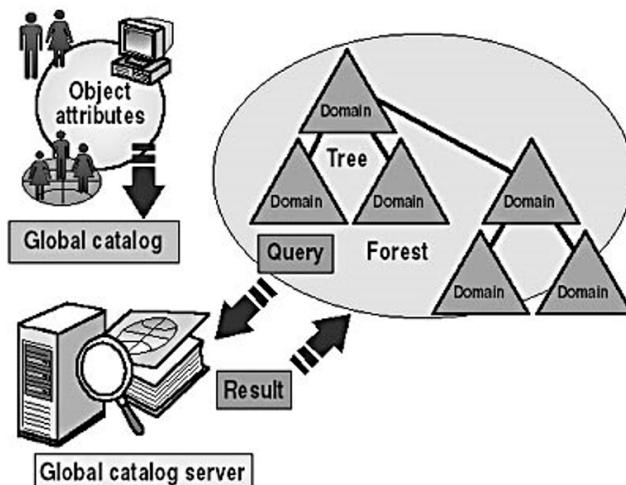
**033.** (FGV/2008/SENADO) O Windows Server 2003 utiliza um serviço de diretório, denominado *Active Directory*, que emprega um banco de dados onde ficam armazenados todos os recursos de uma rede e ele os torna acessíveis a todos os usuários e aplicativos dessa rede. Dentre seus componentes, um representa um depósito de informações que armazena um subconjunto dos atributos de todos os objetos existentes no *Active Directory*, tendo a função de agilizar a realização de queries. Nele existe a informação necessária para que se saiba a localização de qualquer objeto existente no *Active Directory*. Esse componente é denominado:

- a) Main Library.**
- b) Data Schema.**

- c) Global Catalog.
- d) Domain Controller.
- e) Organizational Unit.



- a) **Errado.** Main Library não existe.
- b) **Errado.** Data Schema destaca como os dados serão armazenados.
- c) **Certo.** Global Catalog é a resposta da questão. Trata-se de um depósito de informações que armazena um subconjunto dos atributos de todos os objetos existentes no Active Directory, tendo a função de agilizar a realização de queries. Item **CERTO**.



- d) **Errado.** Domain controller é a máquina que roda o Active Directory em si.
- e) **Errado.** Organizational Unit é uma forma de organização lógica de seus objetos.

**Letra c.**

**034.** (VUNESP/CREA SP/AGENTE ADMINISTRATIVO II - ANALISTA DE SUPORTE/2008) CONSIDERE UM COMPUTADOR SERVIDOR COM SISTEMA OPERACIONAL Windows 2003 Server. Assinale a alternativa que contém o serviço de rede necessário para a instalação do Active Directory nesse computador.

- a) DHCP
- b) DNS
- c) FTP
- d) SMTP
- e) WINS



O **DNS** é pré-requisito para a instalação do Active Directory nesse computador.

**Letra b.**

## LDAP

**035.** (CESPE/ABIN/OFICIAL TÉCNICO DE INTELIGÊNCIA - ÁREA 9/2018) Acerca de OAuth e LDAP (Lightweight Directory Access Protocol), julgue o item seguinte.

LDAP é um protocolo de diretórios que provê repositórios de informações de recursos de sistemas e serviços dentro de um ambiente centralizado e estritamente relacionado ao servidor. Por questão de limitação do padrão x.500, do qual foi originado, o LDAP não suporta funções de segurança e de acesso de cliente.



**LDAP (Lightweight Directory Access Protocol)** é o protocolo principal para acesso ao Active Directory e um protocolo padrão para pesquisa e gerenciamento de diretórios rodando sobre TCP/IP. Um diretório LDAP segue o modelo X.500, que é uma estrutura hierárquica de nós chamada de *Directory Information Tree* (DIT), cada um consistindo de um conjunto de atributos com seus respectivos valores. **LDAP suporta funções de segurança e de acesso de cliente**. Como exemplo, tem a operação StartTLS – a partir da versão 3 do LDAP, que protege a conexão com criptografia (através do protocolo TLS).

**Errado.**

**036.** (CESPE/2008/CBM-DF/ENGENHEIRO DE REDES DE COMUNICAÇÃO) Julgue os itens seguintes, acerca de sistemas operacionais, seus diversos tipos e características. O LDAP (lightweight directory access protocol) é um protocolo útil quando se trata de gerenciamento de autenticação em sistemas Linux, pois permite a centralização, em um único local da rede, de informações acerca de usuários, senhas e diretórios, entre outras. Além disso, o acesso a esses dados pode ser feito de forma segura, pois a maioria dos servidores LDAP oferece conexões criptografadas usando SSL (secure socket layer).



O LDAP (*lightweight directory access protocol*) é independente de plataforma, pode ser usado tanto no AD do Windows, como em várias implementações que se tem para Linux, como Open LDAP, por exemplo. Além disso, LDAP pode oferecer conexões criptografadas para trafegar as informações de forma segura.

**Certo.**

**037.** (CESPE/2008/MPE-RR) Com relação a qualidade de serviço, aplicações de voz e imagem sobre redes e serviços de diretório LDAP, julgue os itens seguintes. O protocolo LDAP, assim como o protocolo DAP (Directory Access Protocol), tem o objetivo de padronizar a comunicação e prover acesso a serviços de diretórios distribuídos. Os clientes são os DUA (Directory User Agent) e os DSA (Directory System Agents).



O DSA (Directory System Agents) roda no SERVIDOR, o restante da questão está ok!

**Errado.**

**038.** (FCC/TRT - 15<sup>a</sup> REGIÃO (SP)/TÉCNICO JUDICIÁRIO - TECNOLOGIA DA INFORMAÇÃO/2015)

Quanto ao LDAP:

- a)** é baseado no padrão X.500, sendo o adequado substituto para banco de dados relacional, file system ou DNS.
- b)** é o protocolo principal para acesso ao Active Directory e um protocolo padrão para acesso a diretórios administrados por meio do TCP/IP.
- c)** aplicações LDAP podem ser agrupadas em até 2 categorias que são: apps para localizar usuários nos bancos de dados; e apps que gerenciam estes usuários.
- d)** na estrutura de um diretório LDAP as informações são armazenadas no padrão objeto-relacional.
- e)** apesar de ser mono-plataforma, não é comercializado como open source.



**a) Errado.** LDAP (*Lightweight Directory Access Protocol*) é baseado no padrão X.500, mas não é o substituto para banco de dados relacional, file system ou DNS.

Limitações:

- Não substitui Banco de Dados Relacionais (heavy updates, processamento transacional, geração de relatórios, SQL, two phase commit).
- Não é um *file system*, dificultando o armazenamento de BLOB (binary large objects).
- Não serve como base para uma solução DNS, não funciona "connectionless" como o DNS.

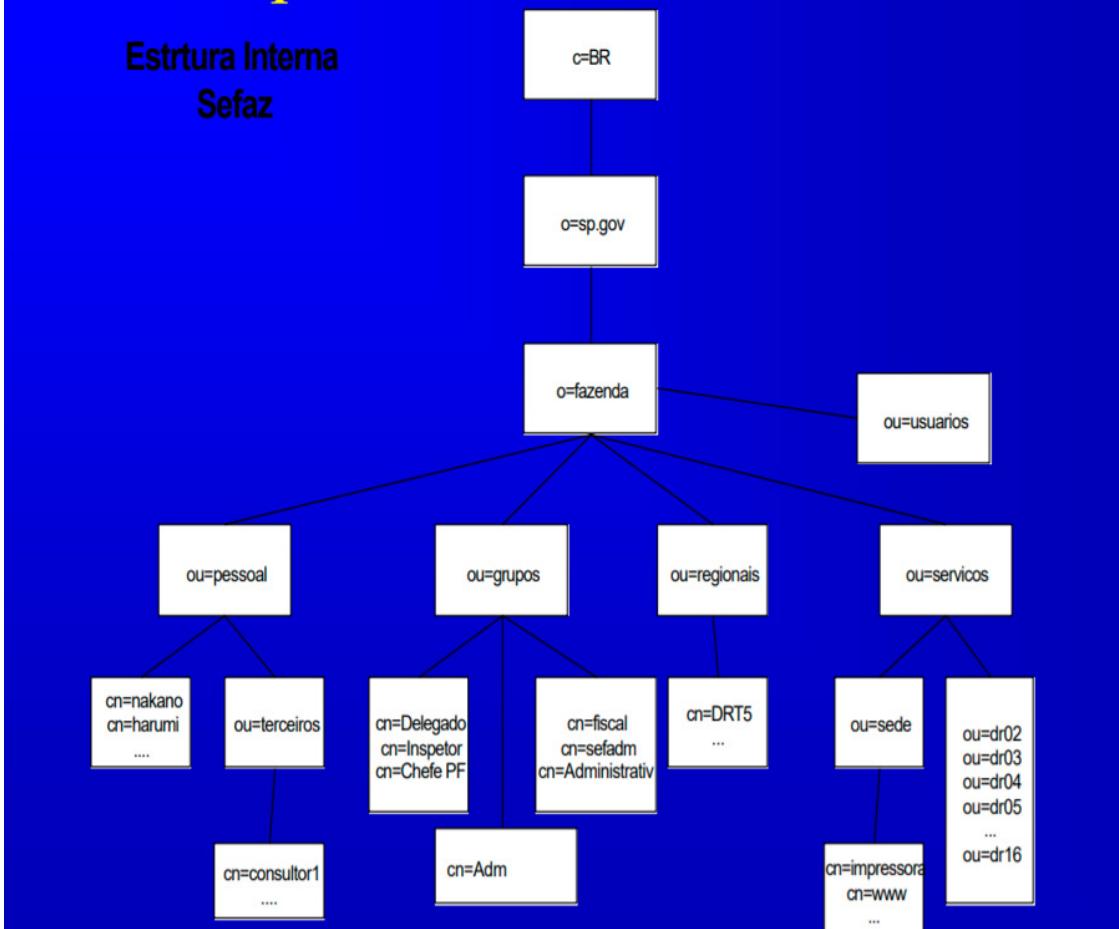
**b) Correto.** LDAP (*Lightweight Directory Access Protocol*) é o protocolo principal para acesso ao Active Directory e um protocolo padrão para pesquisa e gerenciamento de diretórios rodando sobre TCP/IP, criado como uma alternativa ao protocolo *Directory Access Protocol* (DAP), que era mais difícil de utilizar, e era derivado da família de padrões X.500 da ITU-T.

**c) Errado.** Aplicações LDAP podem ser agrupadas em até 3 categorias que são: apps para localizar usuários e recursos na rede, apps que gerenciam estes usuários e recursos, apps de autenticação e segurança.

**d) Errado.** Um diretório LDAP segue o modelo X.500, que é uma estrutura hierárquica de nós chamada de *Directory Information Tree* (DIT), cada um consistindo de um conjunto de atributos com seus respectivos valores.

# Exemplo de uma árvore LDAP

Estrutura Interna  
Sefaz



e) **Errado.** São multiplataformas (Multi-plataforma - aplicações LDAP-aware). Opções de fornecedores de solução: comercial, open source.

Veja mais: [https://memoria.rnp.br/\\_arquivo/sci/2000/ldap.pdf](https://memoria.rnp.br/_arquivo/sci/2000/ldap.pdf)

**Letra b.**

**039.** (FCC/TCE-AP/ANALISTA DE CONTROLE EXTERNO - CONTROLE EXTERNO TECNOLOGIA DA INFORMAÇÃO/2012)

Protocolo que organiza as informações como uma árvore e permite localizar pessoas, recursos e serviços, entre outros. Descrito na RFC 2251, é uma versão simplificada do serviço X.500 do OSI. Trata-se do

- a) LCP.
- b) IMAP.
- c) LDAP.
- d) DNS.
- e) ICMP.



O **LDAP** (*Lightweight Directory Access Protocol*) está diretamente relacionado com o X.500. Ele foi originalmente desenhado como um protocolo simples para *desktops* com a função de gateway para os servidores de X.500.

**Letra c.**

---

**040.** (FCC/TRT 14ª/ANALISTA JUDICIÁRIO/TECNOLOGIA DA INFORMAÇÃO/2011) O sistema LDAP é derivado de um sistema de diretórios chamado

- a) Frame Relay.
- b) ATM.
- c) DoS.
- d) X.500.
- e) DNS.



O **sistema LDAP** é derivado de um sistema de diretórios chamado X.500.

**Nota:** X.500: "pesado" principalmente por requerer que o cliente e servidor utilizassem o protocolo OSI para comunicação. Por sua vez, o LDAP usa mensagens com baixa sobrecarga que são mapeadas diretamente para a camada de TCP (porta padrão 389 e 636 sobre SSL) do TCP/IP.

**Letra d.**

**041.** (FCC/MPE-RN/ANALISTA DE TI/2010) Podem ser consideradas características do LDAP: organização de servidores de forma hierárquica; substituição dos bancos de dados relacionais; multiplataforma; padrão aberto; grande escalabilidade.



A banca costuma tratar o **LDAP** (*Lightweight Directory Access Protocol*) como um serviço de diretórios, mas, na verdade ele é um protocolo de acesso ao serviço de diretórios.

A **função** do LDAP é organizar os recursos de rede de forma hierárquica, como uma árvore de diretório. LDAP é uma versão simplificada do padrão para sistema de diretórios chamado **X.500**.

São características relacionadas ao **LDAP**:

- organização de servidores de forma hierárquica;
- multiplataforma (pode ser usado tanto no Active Directory do Windows, como em várias implementações que se tem para Linux, como Open LDAP, por exemplo);
- baseado em padrão aberto;

- grande escalabilidade (é possível replicar servidores LDAP (com backup ou balanceamento de carga) e incluir novos servidores, à medida que cresce a estrutura da organização);
- oferta maior velocidade de consulta que um banco de dados relacional;
- disponibiliza esquemas padronizados para dados e consolidação de informações, é replicável e distribuível. A substituição dos bancos de dados relacionais não está relacionada ao LDAP. Normalmente um cliente conecta-se ao servidor LDAP, através da porta padrão 389 (TCP).

**Errado.**

---

## GABARITO

---

1. a                    37. E  
2. C                    38. b  
3. b                    39. c  
4. b                    40. d  
5. a                    41. E  
6. c  
7. C  
8. E  
9. b  
10. b  
11. b  
12. C  
13. a  
14. a  
15. a  
16. c  
17. C  
18. C  
19. C  
20. e  
21. c  
22. a  
23. E  
24. b  
25. C  
26. C  
27. b  
28. c  
29. E  
30. E  
31. c  
32. C  
33. c  
34. b  
35. E  
36. C

## REFERÊNCIAS

BALIEIRO, R. **Sistemas operacionais**. Rio de Janeiro: Estácio de Sá, 2015.

Estácio. **Sistemas Operacionais**. Disponível em: <<http://estacio.webaula.com.br/cursos/gon291/aula1.html>>. Acesso em: nov. 2020.

MACHADO, F. B.; MAIA, L. P. **Arquitetura de Sistemas Operacionais**. 4.ed., Rio de Janeiro: LTC, 2007.

OLIVEIRA, R. S., CARISSIMI, A. S., TOSCANI, S. S. **Sistemas Operacionais**. Porto Alegre: Instituto de Informática da UFRGS: Editora Sagra Luzzatto, 2001.

QUINTÃO, P. L. **Notas de aula**, 2020.

SHAW, A. C. **Sistemas e Software de Tempo Real**. Porto Alegre: Bookman, 2003.

SILBERSCHATZ, A., GAGNE, G., GALVIN, P. B. **Sistemas Operacionais – Conceitos e Aplicações**. Rio de Janeiro: Campus, 2001.

SILBERSCHATZ, A., GALVIN, P. B. **Sistemas Operacionais: Conceitos**. Rio de Janeiro: Campus, 2000.

SILBERSCHATZ, Abraham. **Sistemas operacionais com Java**. Rio de Janeiro: Elsevier Brasil, 2008.

TANENBAUM, A. S., WOODHULL. **Sistemas Operacionais: Projeto e Implementação**. 2<sup>a</sup>. Ed. Porto Alegre: Bookman, 2000.

TANENBAUM, A. S. **Sistemas operacionais modernos**. 3. ed. Rio de Janeiro: Campus, 2010.

ZANINE, A. **Apostila de SISTEMAS OPERACIONAIS I**. Tubarão, 2000.

## **Patrícia Quintão**



Mestre em Engenharia de Sistemas e computação pela COPPE/UFRJ, Especialista em Gerência de Informática e Bacharel em Informática pela UFV. Atualmente é professora no Gran Cursos Online; Analista Legislativo (Área de Governança de TI), na Assembleia Legislativa de MG; Escritora e Personal & Professional Coach. Atua como professora de Cursinhos e Faculdades, na área de Tecnologia da Informação, desde 2008. É membro: da Sociedade Brasileira de Coaching, do PMI, da ISACA, da Comissão de Estudo de Técnicas de Segurança (CE-21:027.00) da ABNT, responsável pela elaboração das normas brasileiras sobre gestão da Segurança da Informação.

Autora dos livros: Informática FCC - Questões comentadas e organizadas por assunto, 3<sup>a</sup>. edição e 1001 questões comentadas de informática (Cespe/UnB), 2<sup>a</sup>. edição, pela Editora Gen/Método.

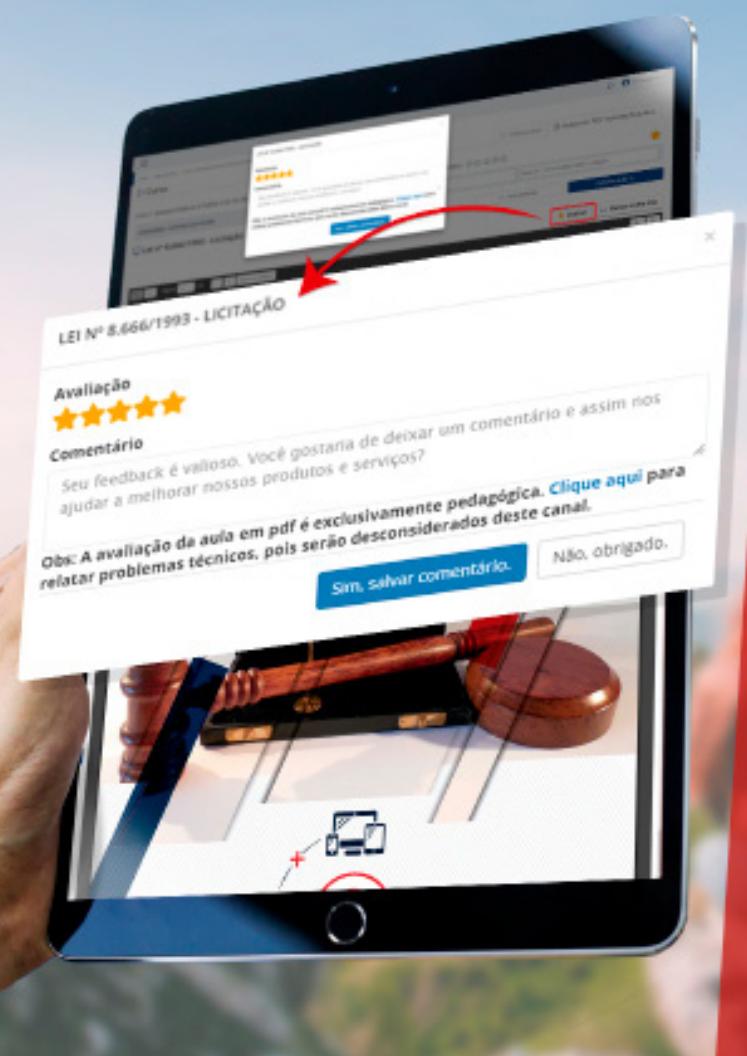
Foi aprovada nos seguintes concursos: Analista Legislativo, na especialidade de Administração de Rede, na Assembleia Legislativa do Estado de MG; Professora titular do Departamento de Ciência da Computação do Instituto Federal de Educação, Ciência e Tecnologia; Professora substituta do DCC da UFJF; Analista de TI/Suporte, PRODABEL; Analista do Ministério Público MG; Analista de Sistemas, DATAPREV, Segurança da Informação; Analista de Sistemas, INFRAERO; Analista - TIC, PRODEMGE; Analista de Sistemas, Prefeitura de Juiz de Fora; Analista de Sistemas, SERPRO; Analista Judiciário (Informática), TRF 2<sup>a</sup> Região RJ/ES, etc.

@coachpatriciaquintao

/profapatriciaquintao

@plquintao

t.me/coachpatriciaquintao



# **NÃO SE ESQUEÇA DE AVALIAR ESTA AULA!**

**SUA OPINIÃO É MUITO IMPORTANTE  
PARA MELHORARMOS AINDA MAIS  
NOSSOS MATERIAIS.**

**ESPERAMOS QUE TENHA GOSTADO  
DESTA AULA!**

**PARA AVALIAR, BASTA CLICAR EM LER  
A AULA E, DEPOIS, EM AVALIAR AULA.**

**AVALIAR**