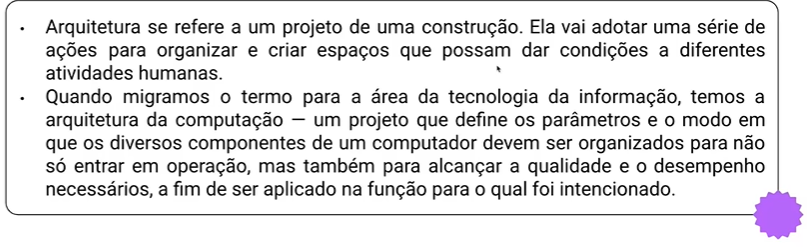
**Arquitetura de Computadores:**



O objetivo principal dela é propor um passeio por diferentes áreas da Computação: vamos saber mais sobre: a história da computação, como os dados são representados na memória, ter uma ideia sobre algoritmos, linguagens de programação, estruturas de dados, sistemas operacionais, bancos de dados, engenharia de software, redes de computadores, segurança da informação, computação gráfica, teoria da computação etc.

Computador:

**Um processador, ou CPU** (Unidade Central de Processamento), é o componente responsável por executar instruções e realizar cálculos em um computador. Ele interpreta comandos e coordena as operações dos demais componentes do sistema. A família Core i, da Intel, é uma linha de processadores com diversas gerações, que variam em desempenho e funcionalidades, atendendo diferentes necessidades de uso, desde tarefas básicas até aplicações exigentes como jogos e edição de vídeo.

**A memória de acesso aleatório (RAM)** é um tipo de memória volátil usada para armazenar temporariamente dados e instruções que o processador precisa acessar rapidamente. Ela permite leitura e gravação de informações de forma aleatória, sem seguir uma ordem sequencial, garantindo alta velocidade de acesso. A RAM é essencial para o desempenho do sistema, pois armazena programas em execução e dados em uso, sendo liberada ao desligar o computador.

**O HD (Disco Rígido) e o SSD (Unidade de Estado Sólido)** são dispositivos de armazenamento de dados. O HD usa discos magnéticos giratórios para gravar e acessar informações, sendo mais acessível, porém mais lento e suscetível a danos mecânicos. Já o SSD utiliza memória flash para armazenar dados, oferecendo maior velocidade, resistência e eficiência energética, mas com custo mais elevado por gigabyte. Ambos são essenciais para armazenar sistemas, programas e arquivos.

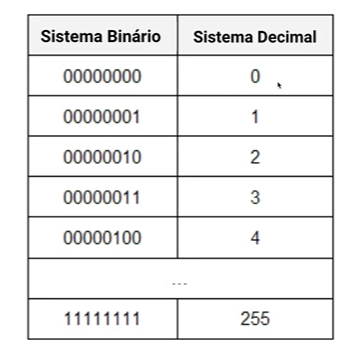
**Componentes físicos**, ou **hardware**, são as **partes tangíveis** de um computador, responsáveis por sua operação. Incluem o processador (CPU), memória RAM, dispositivos de armazenamento (HD ou SSD), placa-mãe, fonte de alimentação, periféricos (teclado, mouse) e placas adicionais, como a de vídeo. Eles trabalham juntos para executar tarefas e suportar o funcionamento dos sistemas e programas instalados.

Já o **software compreende os programas que instruem o hardware** a realizar tarefas específicas, como sistemas operacionais, aplicativos, jogos e navegadores de internet.

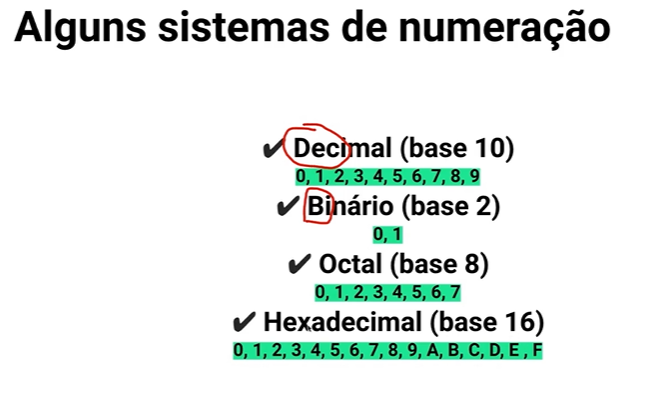
**Sistemas de Numeração:**

Bit: “0” – aberto/desligado ou “1” – fechado/ligado

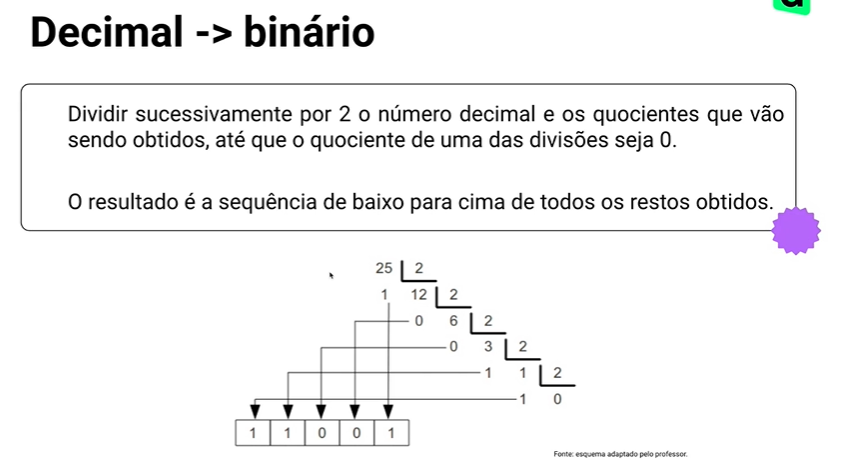
Byte: 8xbit. Ex: A =10010010 / a= 11001100

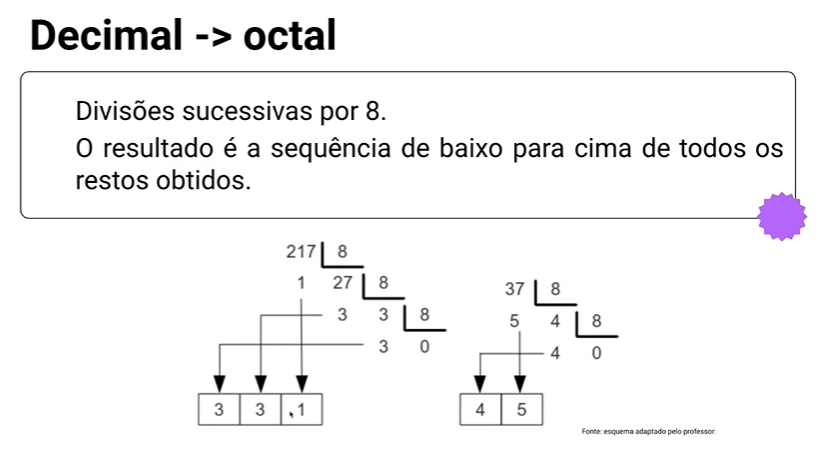


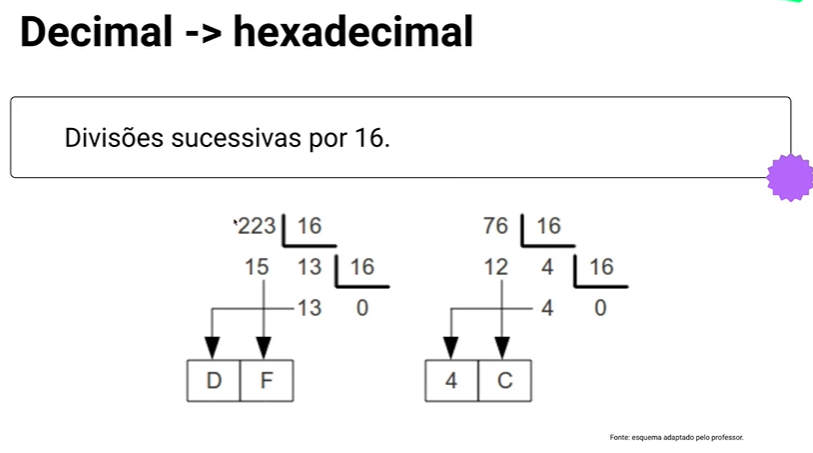


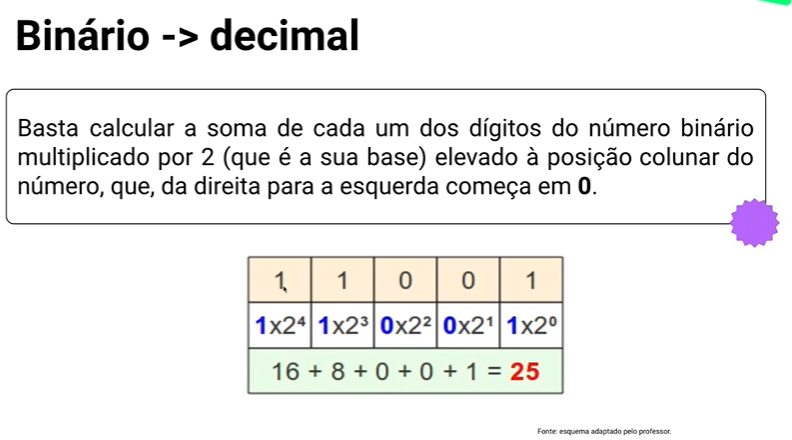


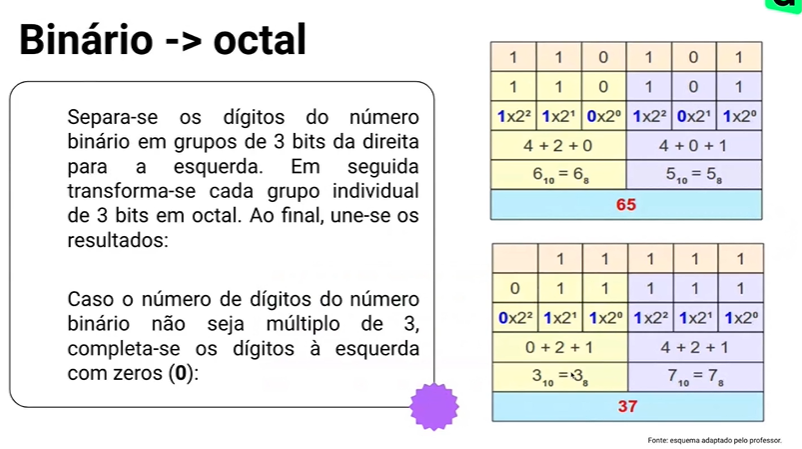
**Conversões:**

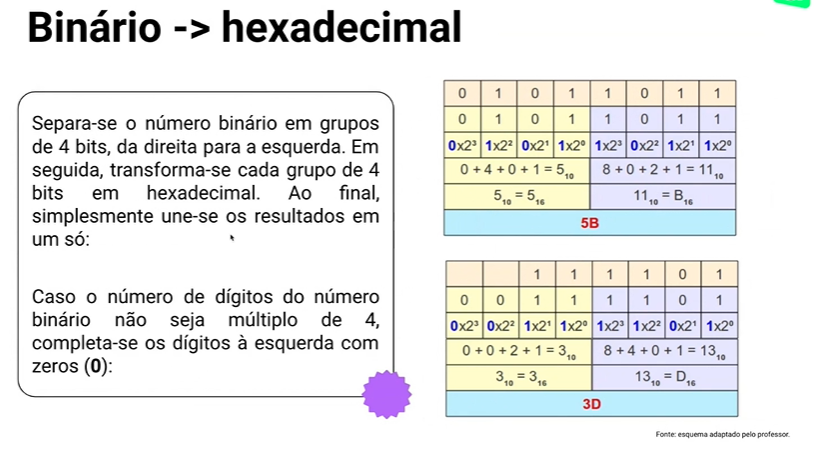


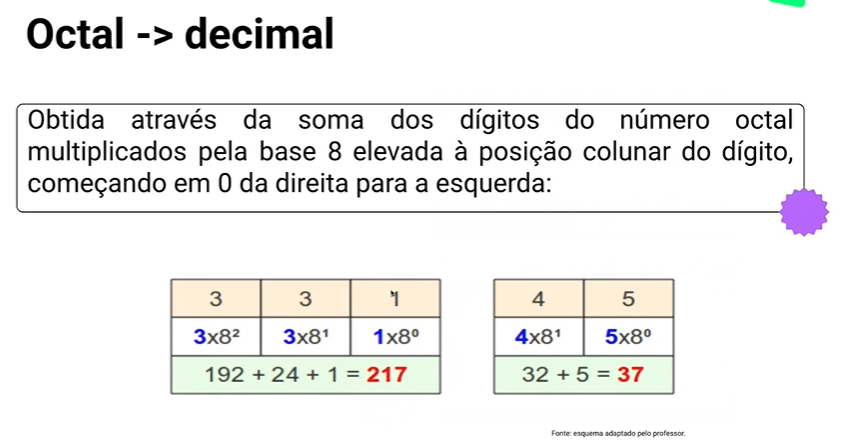


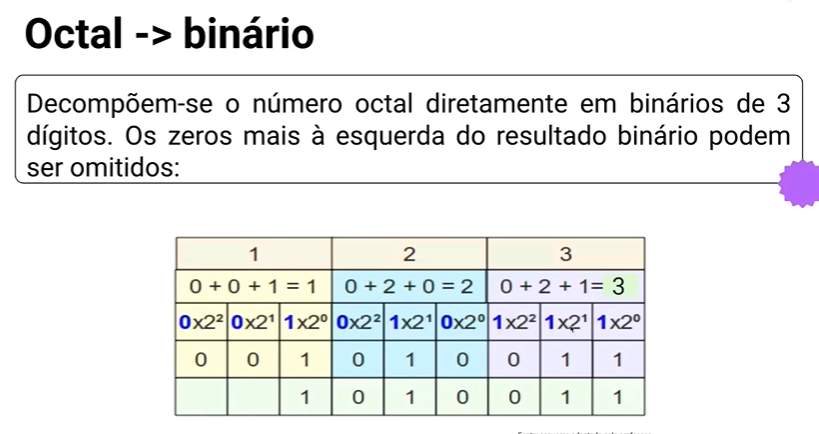


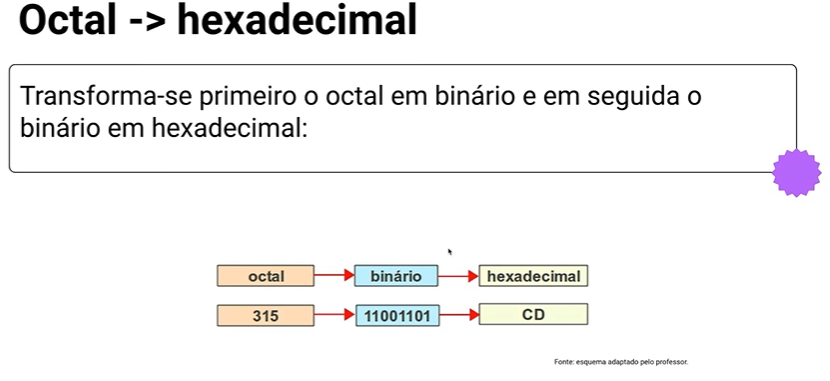


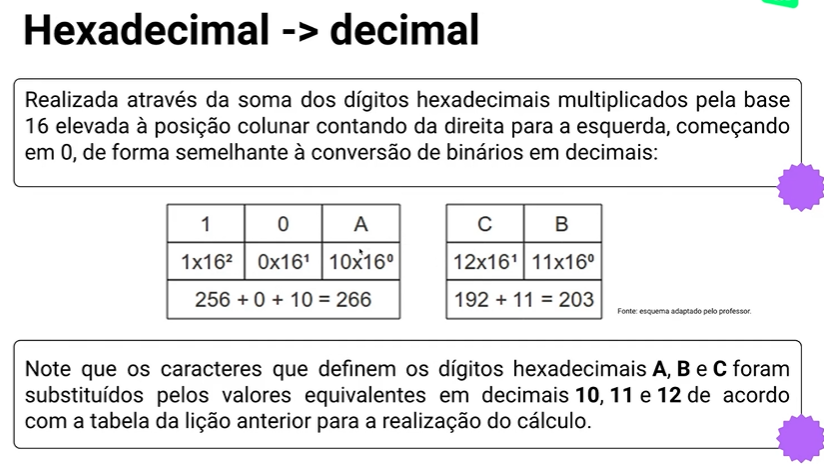


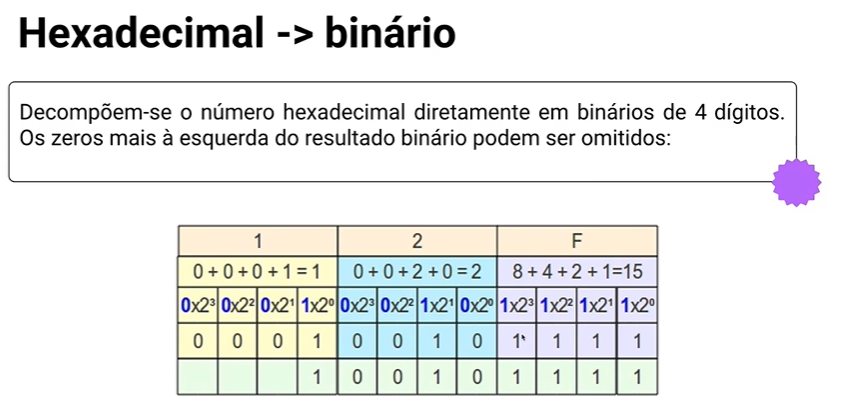


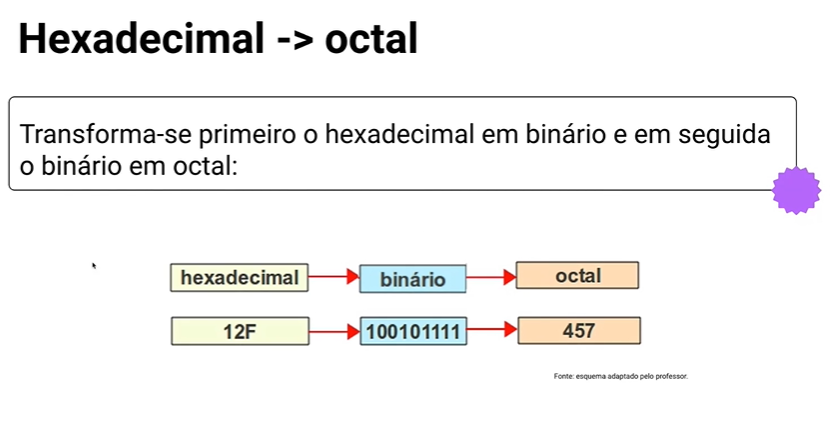






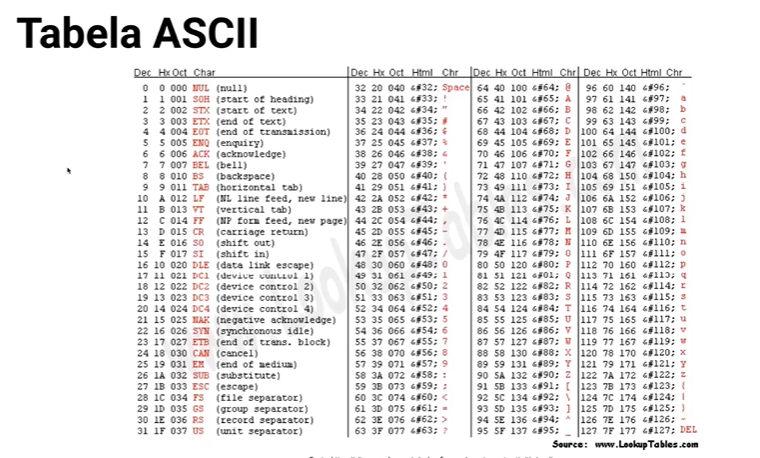






**Resumo e dicas:**



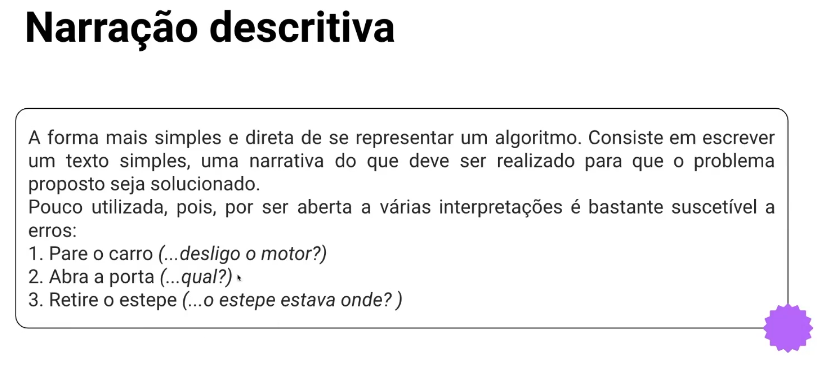


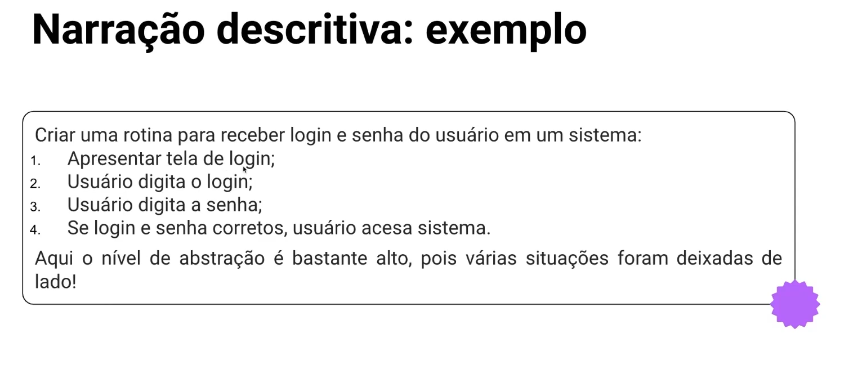
**Algoritmo:**

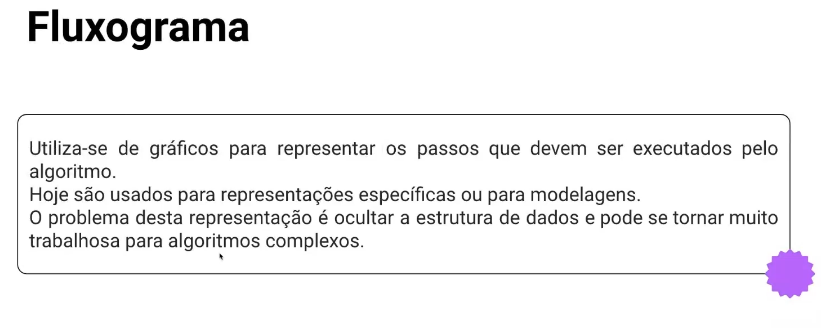
Sequencia de instrução ou comandos realizados de maneira sistematica com o objetivo de resolver um problema ou executar uma tarefa.

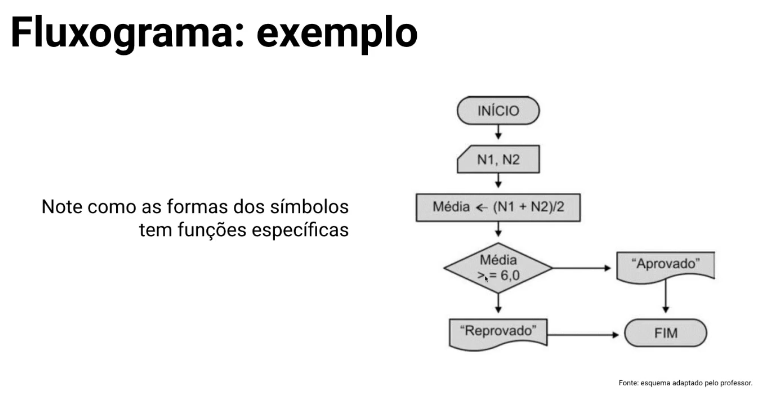
Possui entrada, processamento e saida.

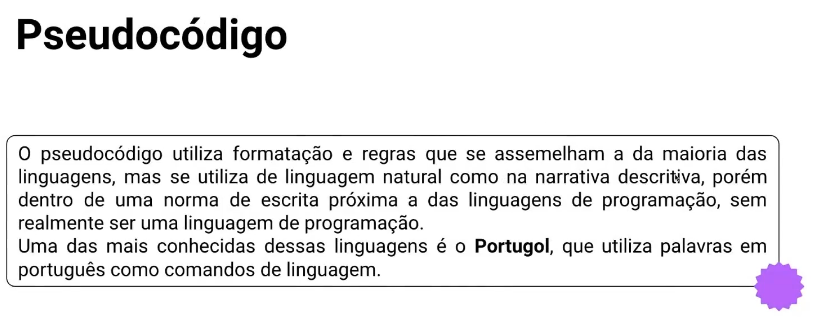
Como representar?

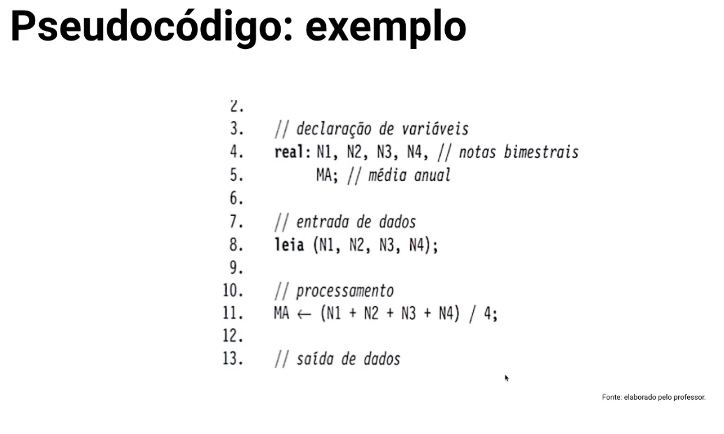






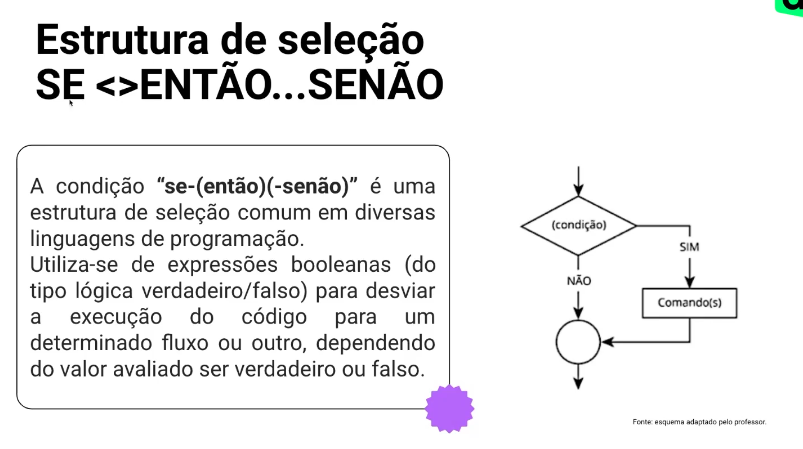




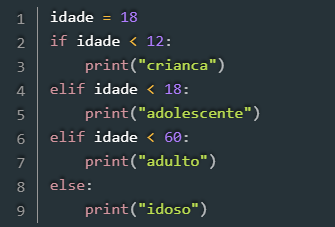


Estruturas:

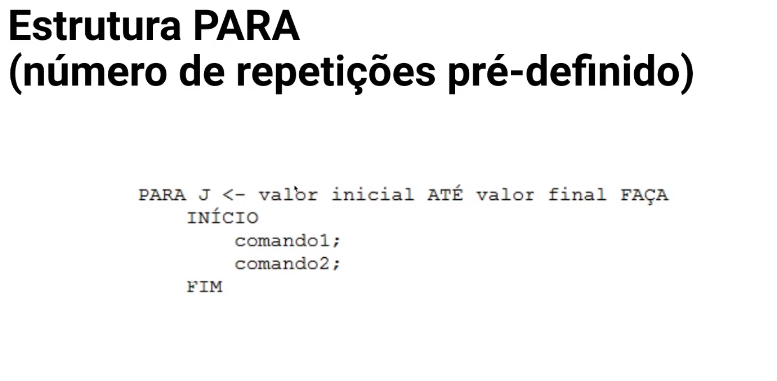
* SE:



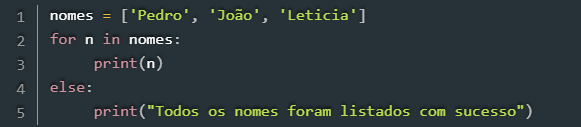
Em Python:

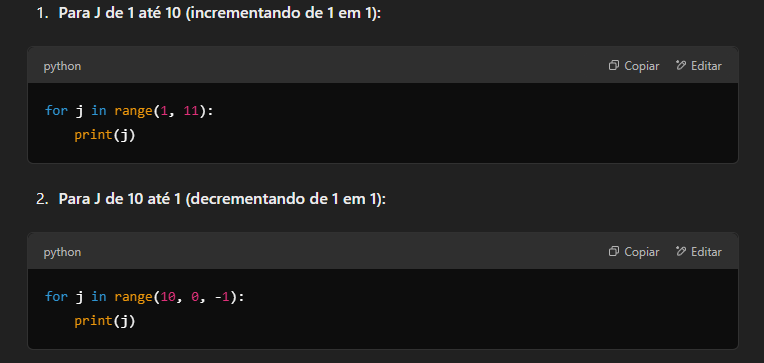


* PARA:

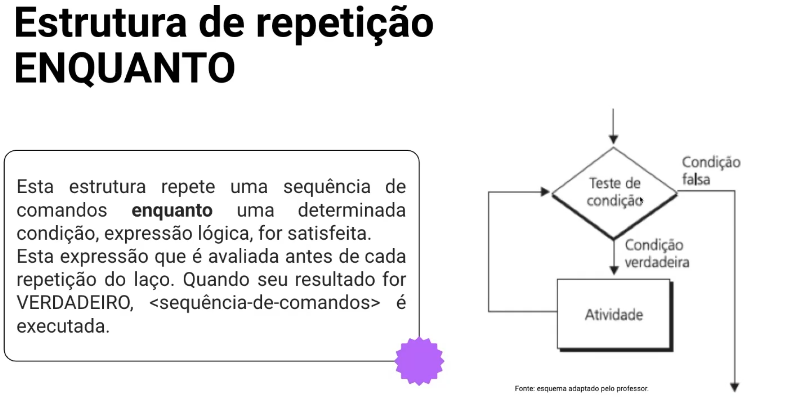


Em Python:

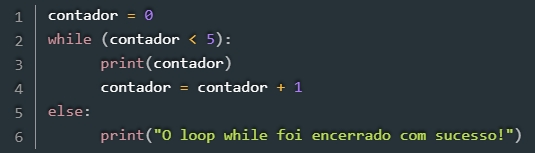


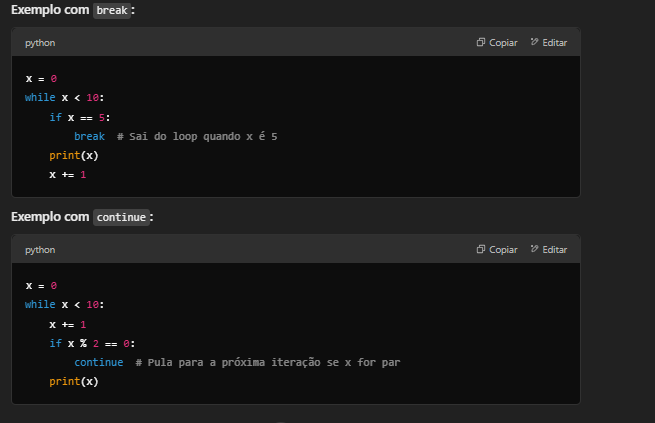


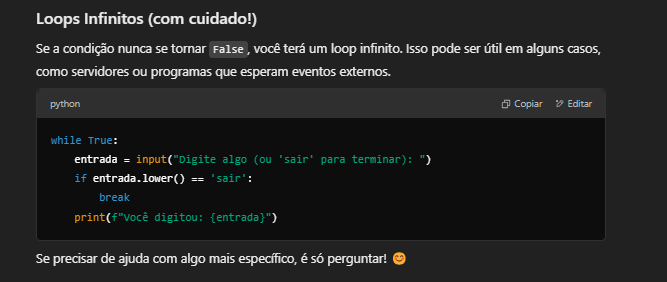
* Enquanto:



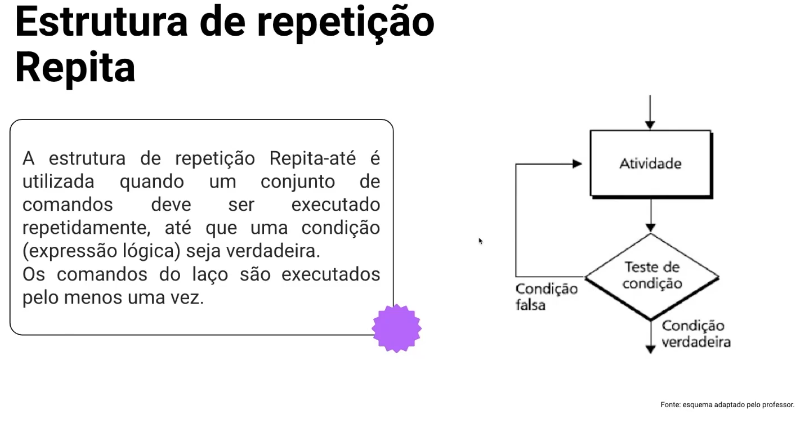
Em Python:



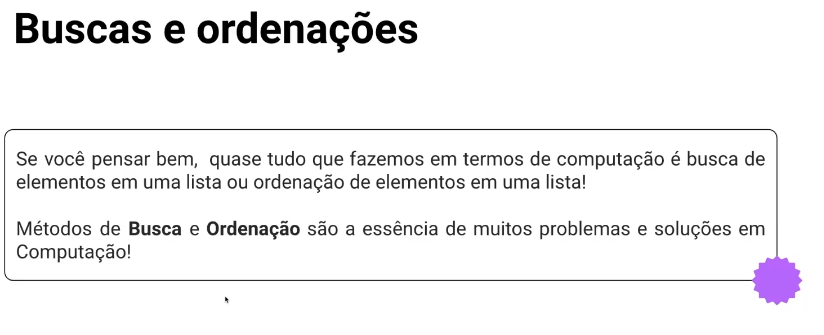


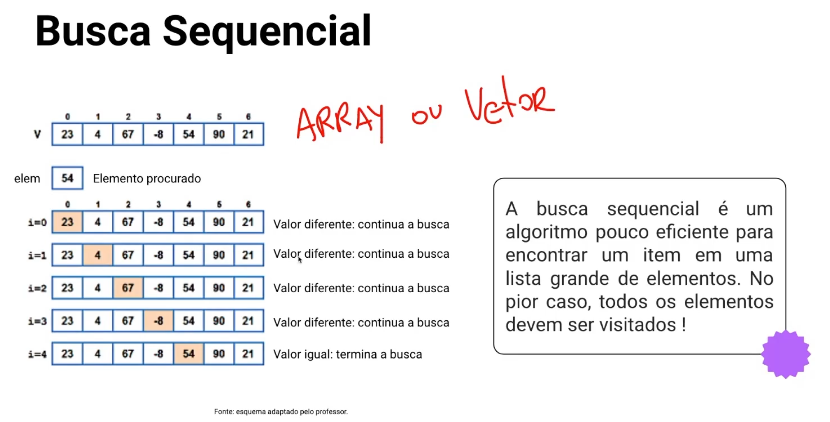


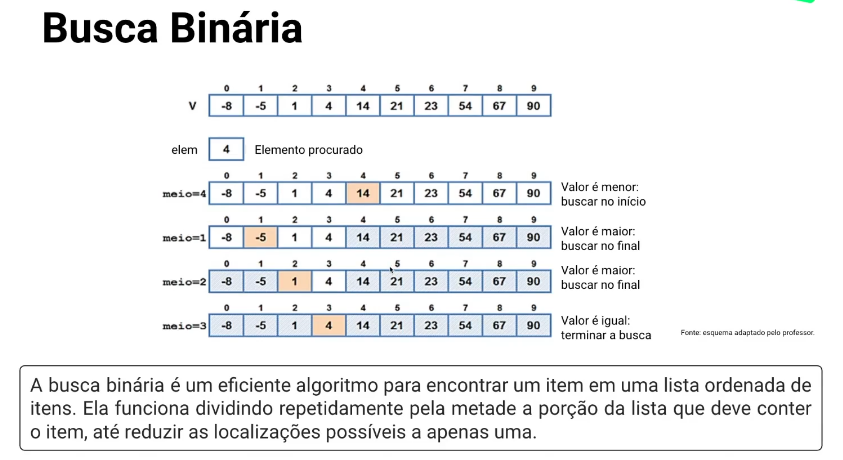
* Repita:

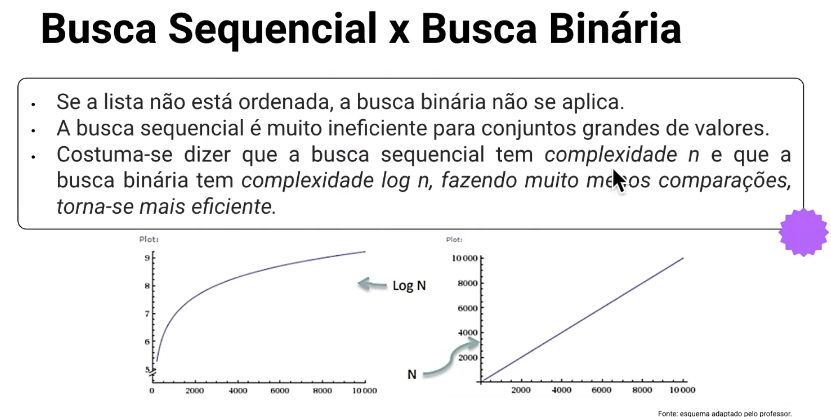


Eficiência dos algoritmos:



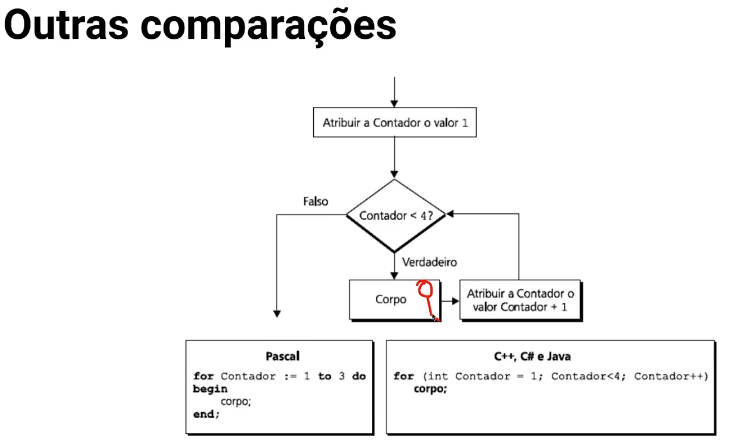


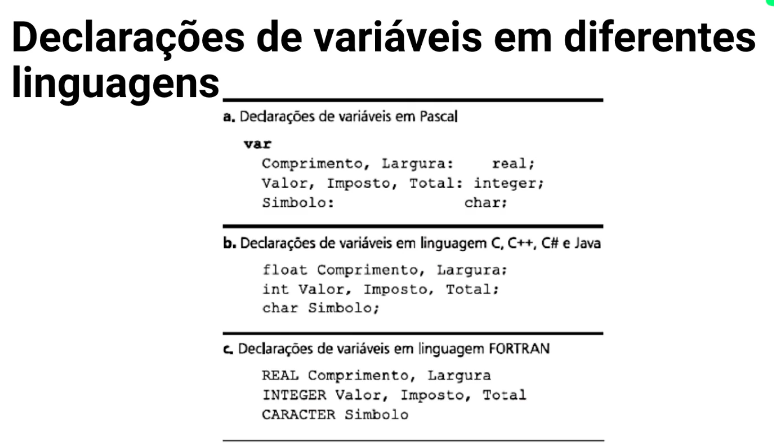




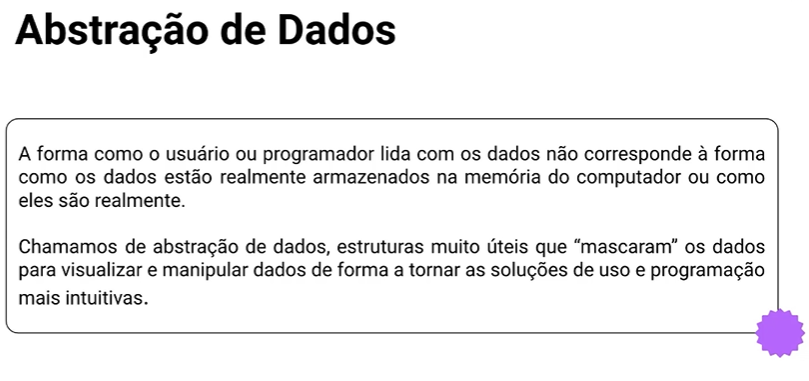
**Linguagens de Programação:**

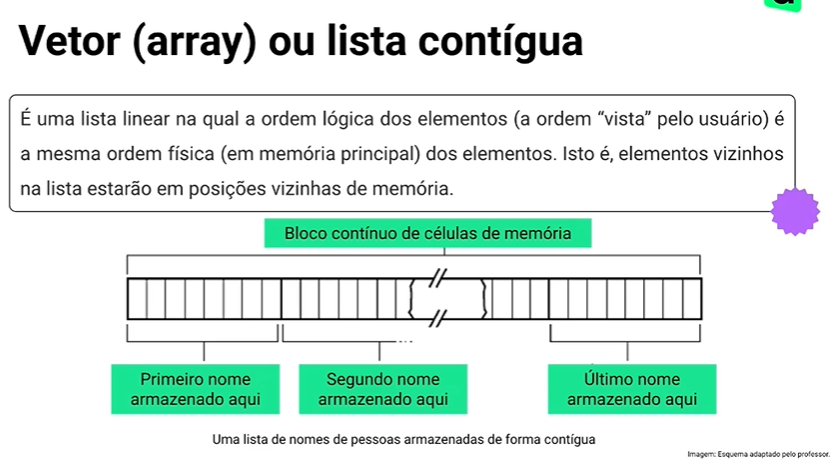




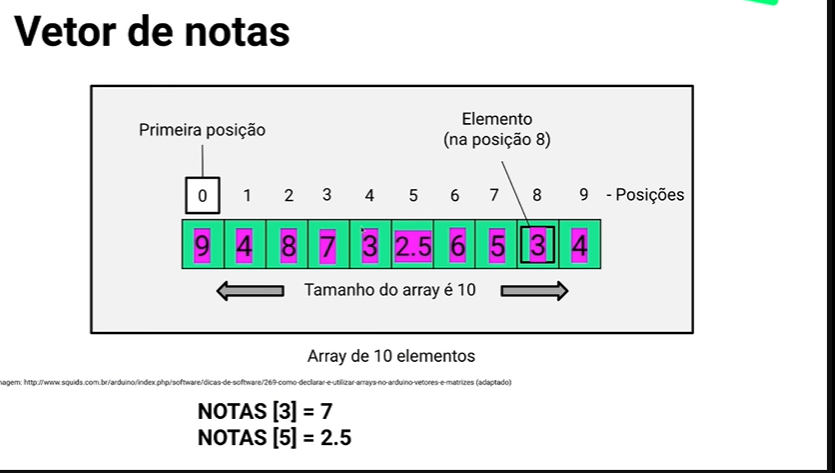


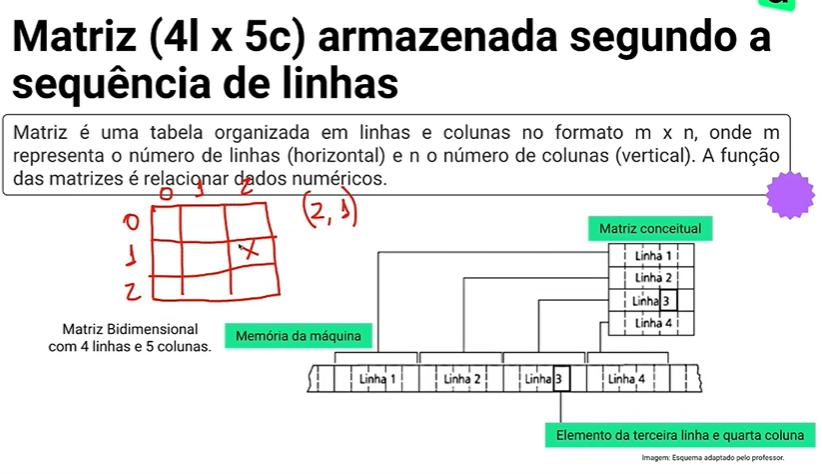
**Vetores:**



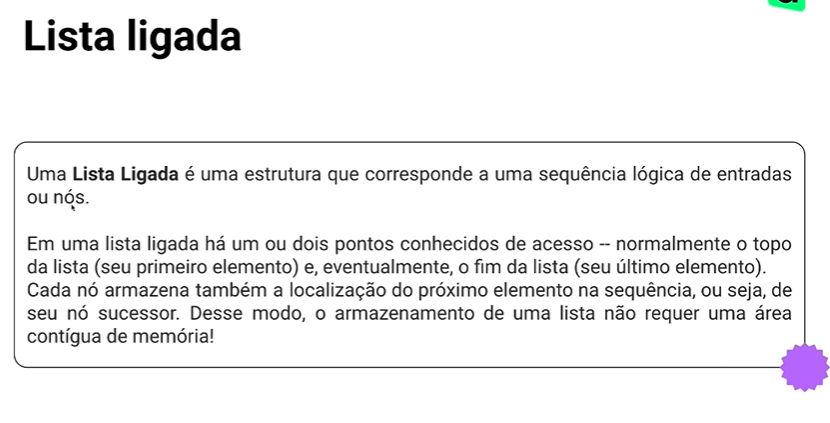


EX:



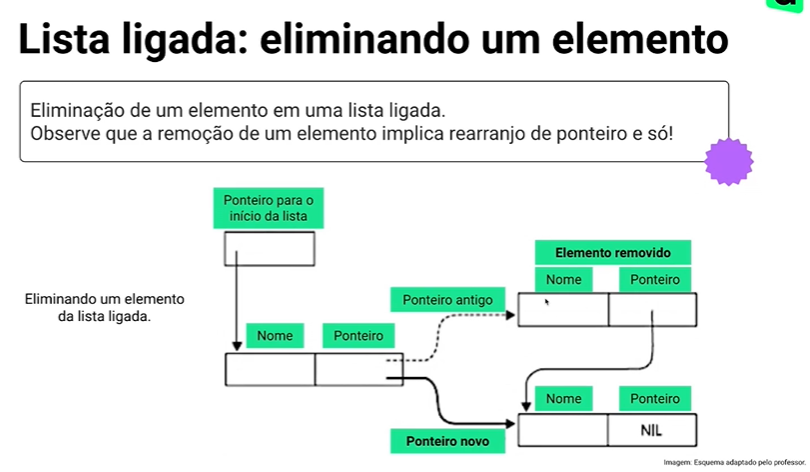
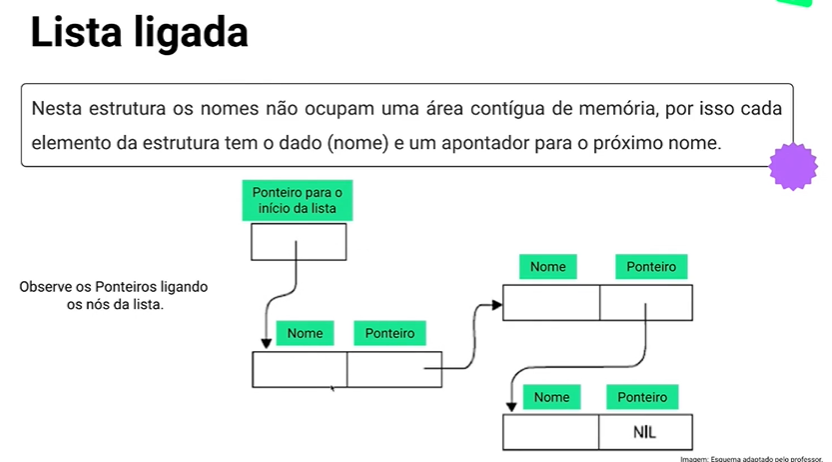


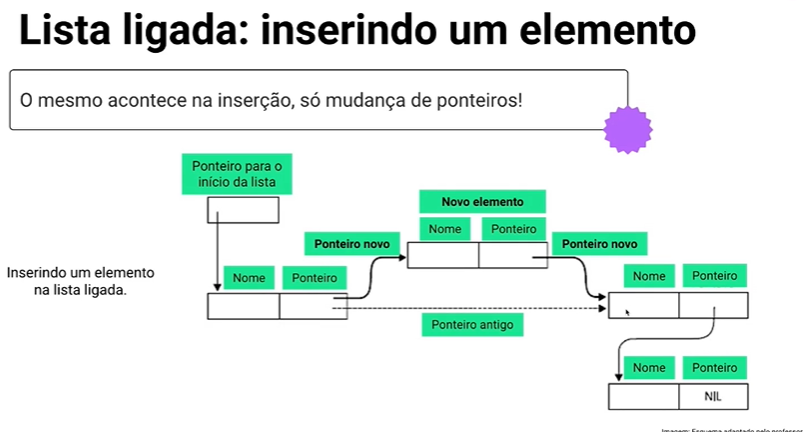
**Listas ligadas:**



Listas ligadas são estruturas de dados que consistem em nós interligados por ponteiros. Cada nó contém um valor e um ponteiro que aponta para o próximo nó na sequência. A flexibilidade dos ponteiros permite fácil inserção e remoção de elementos sem a necessidade de realocação de memória, como ocorre em vetores.

A principal diferença entre listas ligadas e vetores reside na forma como os elementos são armazenados e acessados. As listas ligadas permitem inserção e remoção eficientes de elementos em qualquer posição, enquanto os vetores exigem realocação de memória ao inserir ou remover elementos em posições diferentes do final.

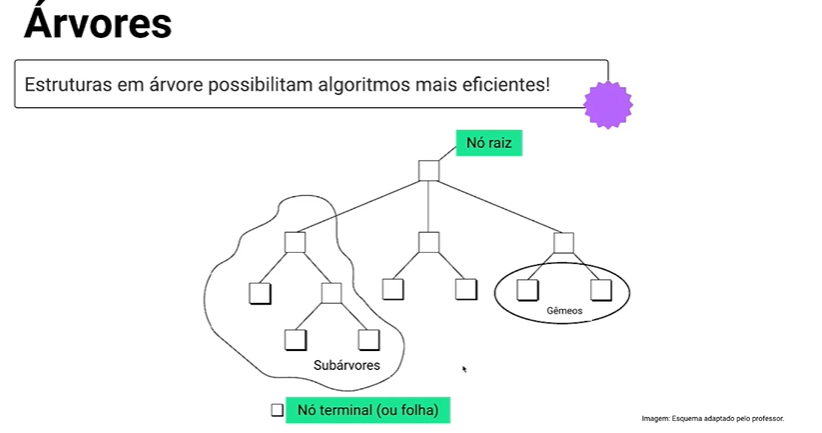




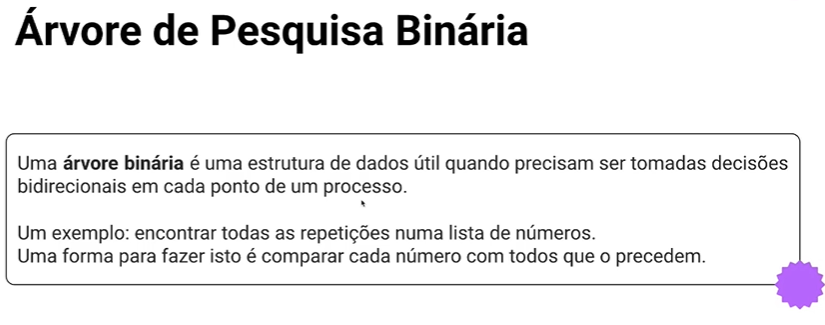
**Árvores:**

Em computação, uma árvore é uma estrutura de dados hierárquica que se assemelha a uma árvore invertida, com a raiz no topo e ramificações para baixo, semelhante à estrutura de pastas em um sistema de arquivos.

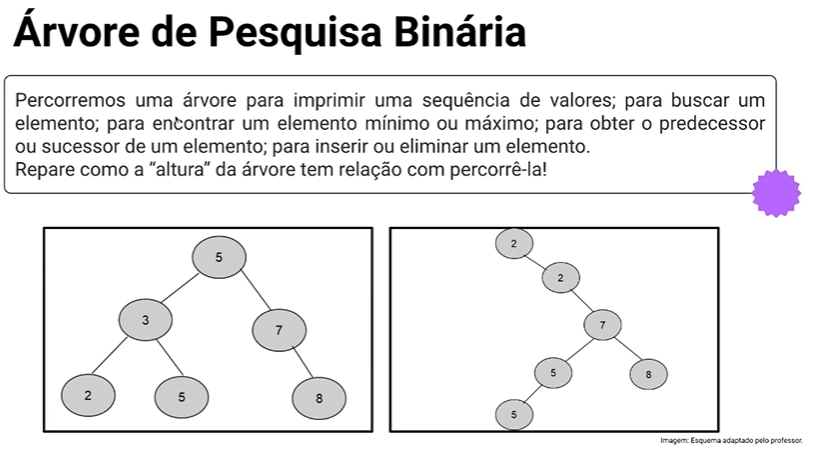
A estrutura hierárquica das árvores permite algoritmos eficientes, como algoritmos de busca, que se beneficiam da hierarquia para encontrar elementos de forma otimizada.



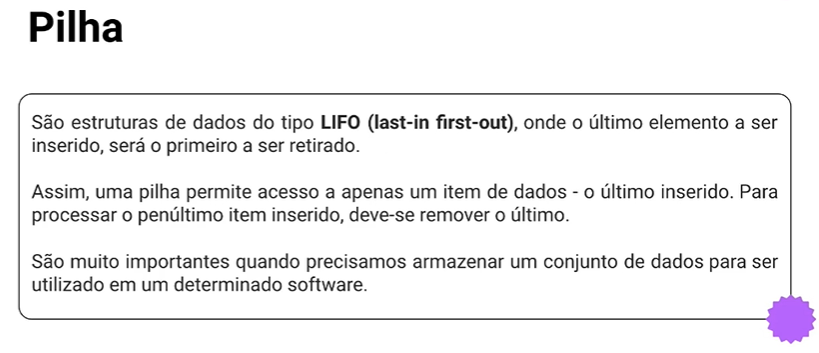
Árvores de pesquisa binária são uma implementação específica de árvores que permite tomar decisões bidirecionais, como em uma busca binária, onde cada nó possui no máximo dois filhos.



O percurso em árvore binária é usado em grafos para representar decisões em um caminho, como em aplicativos de GPS, onde a raiz de uma subárvore é impressa entre os valores de suas subárvores esquerda e direita.



**Pilha:**



**Fila:**

