



RESUMO DEFINITIVO 2021

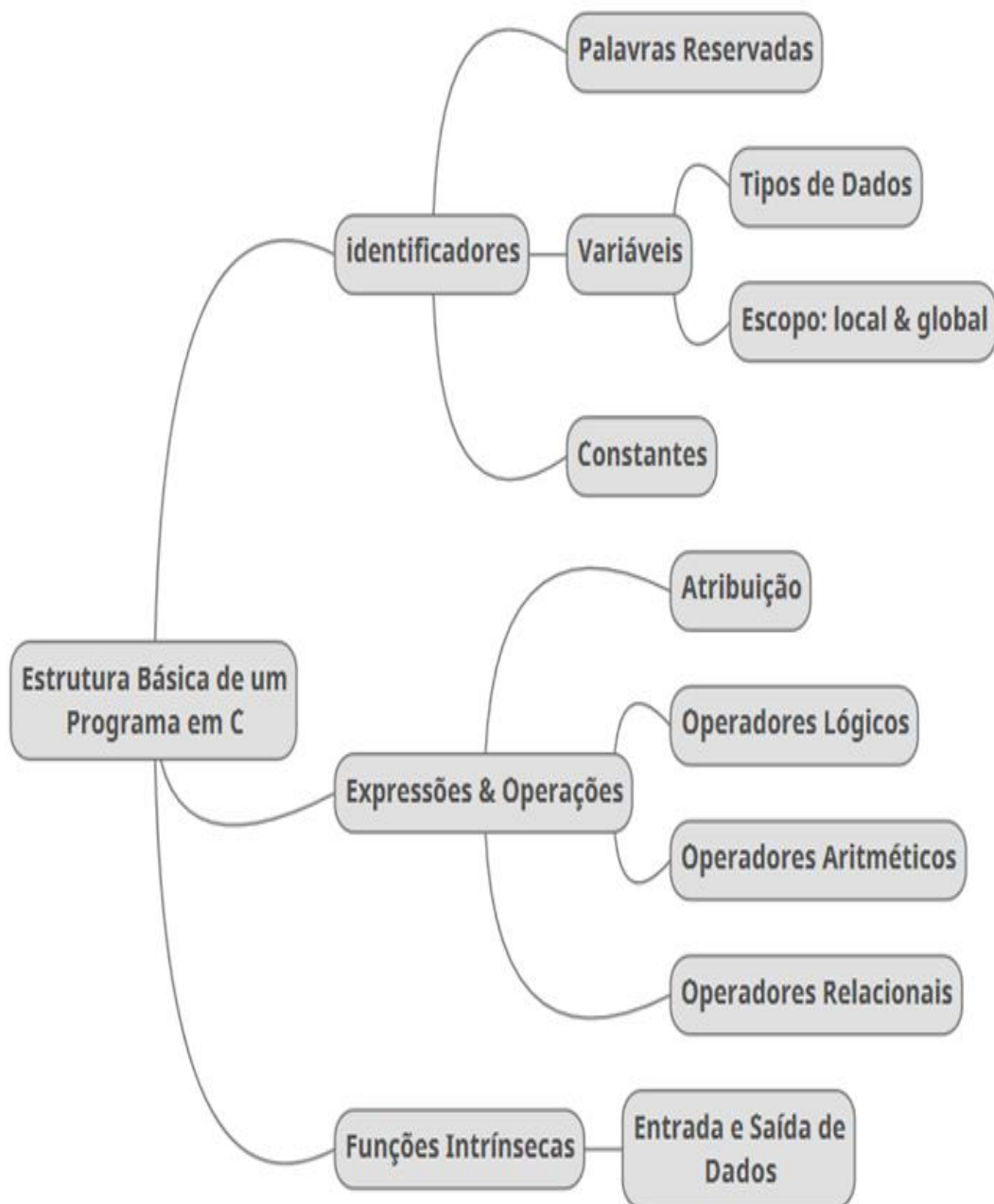
**Lógica de Programação
e
Estruturas de Dados
em Linguagem C**

GUIA DE ESTUDOS

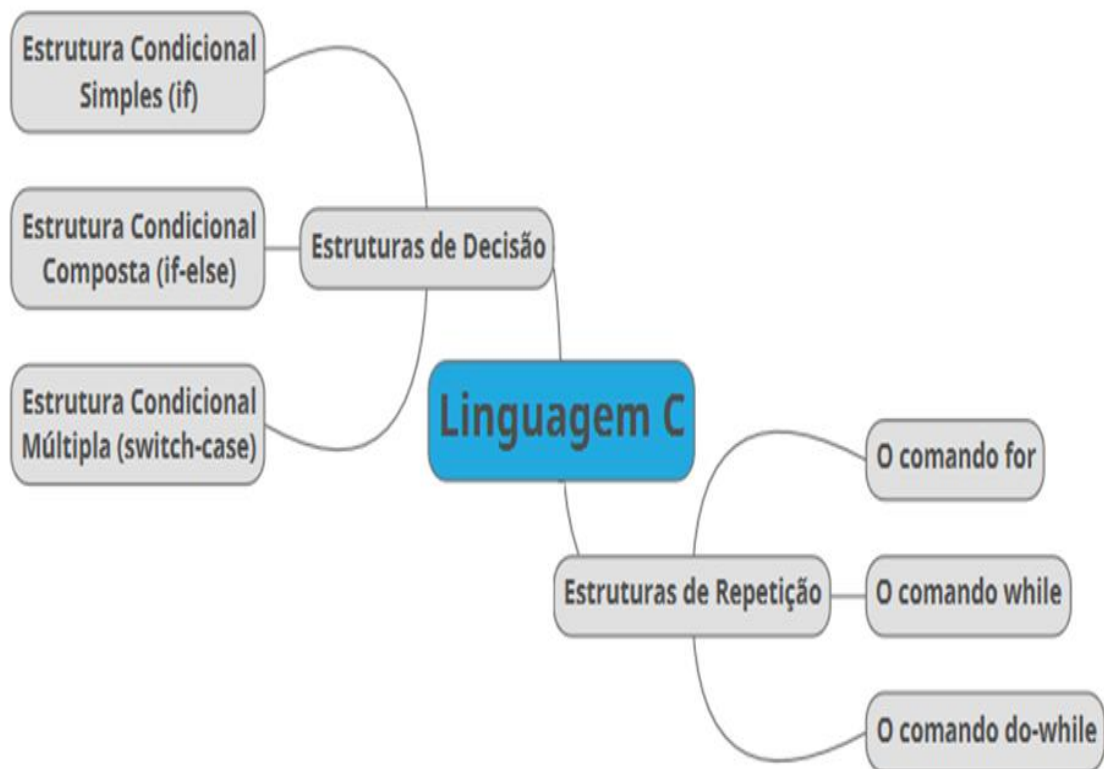
Lógica de Programação e Estruturas de Dados

1. Antes de mais nada: entenda os grandes grupos de conceitos a serem compreendidos.

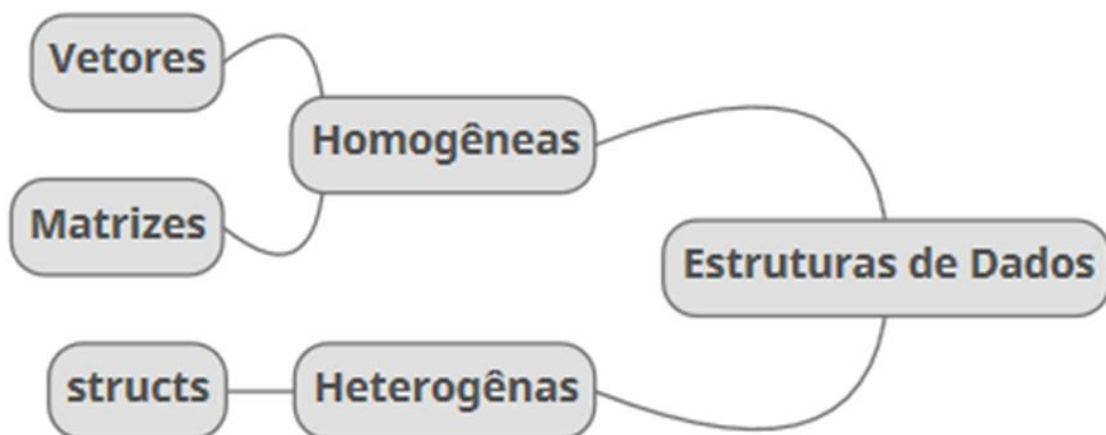
a. Fundamentos elementares:



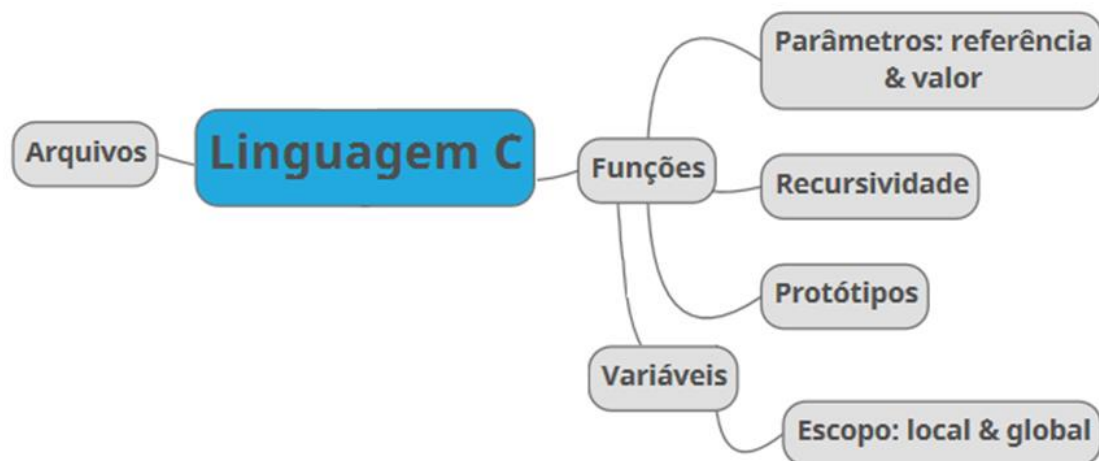
b. Controle de fluxo:



c. Estruturas de dados básicas:



d. Funções e Arquivos:



2. Como analisar as questões e se preparar para resolver exercícios?

- a) Realizar a leitura com atenção.
- b) Grifar os pontos importantes, palavras chave (“palavrinhas mágicas”):
 - a. Algoritmo
 - b. Código fonte
 - c. Teste de mesa
 - d. Variáveis
 - e. Arranjo (*Array*)
 - f. Vetor
 - g. Matriz
 - h. Arranjo
 - i. Rotinas e sub-rotinas
 - j. Procedimento
 - k. Função
- c) Entender o que está sendo pedido, “traduzir”, interpretar o enunciado.

3. Quais temas podem ter relação com lógica de programação?

- a. Pseudocódigo e linguagens de programação
 - i. Elimina ambiguidades
 - ii. Sintaxe (estrutura “ortográfica”)
 - iii. Semântica (lógica funcional)
 - iv. Erros sintáticos x semânticos
- b. Conceitos básicos
 - i. Variáveis
 - ii. Constantes
 - iii. Expressões
 - iv. Atribuição
 - v. Entrada e saída de dados
 - vi. Funções
- c. Estruturas condicionais
 - i. Simples (`if`)
 - ii. Composta (`if-else`)
 - iii. Aninhadas
 - iv. Múltipla (`switch-case`)
- d. Estruturas de repetição
 - i. Por contagem (`for`)
 - ii. Por condição (`while` e `do-while`)
 - iii. Aninhadas
- e. Estruturas de dados homogêneas e heterogêneas
 - i. Registros (`structs`)
 - ii. Vetores (geralmente acompanhada de uma estrutura de repetição do tipo `for`)
 - iii. Matrizes (geralmente acompanhada de um `for` aninhado a outro `for`)
- f. Recursividade
 - i. Caso base – solução trivial, por definição
 - ii. Resolve o problema maior a partir da solução de um problema menor
 - iii. Dividir para conquistar
- g. Ponteiros
 - i. Alocação dinâmica de memória

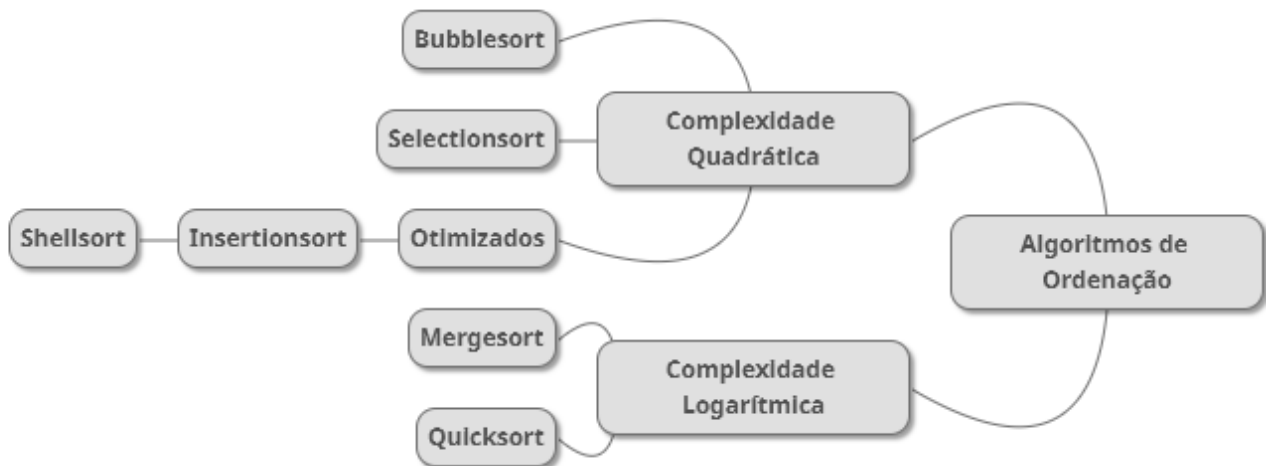
ii. Passagem de parâmetros por referência

h. TESTES DE MESA!!! IMPORTANTÍSSIMO!!!

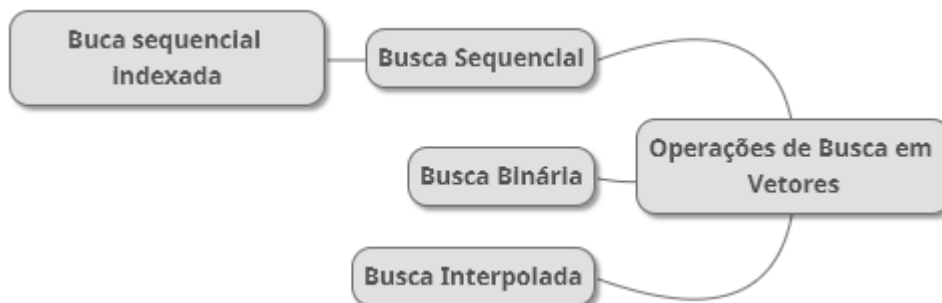
- i. Aqui, trata-se de executar o algoritmo como se fosse a própria máquina. Quando a máquina solicita interações com o usuário, você também deve assumir o papel do respectivo usuário.

4. Caso já tenha uma boa noção sobre programação, é hora de evoluir para as estruturas de dados avançadas:

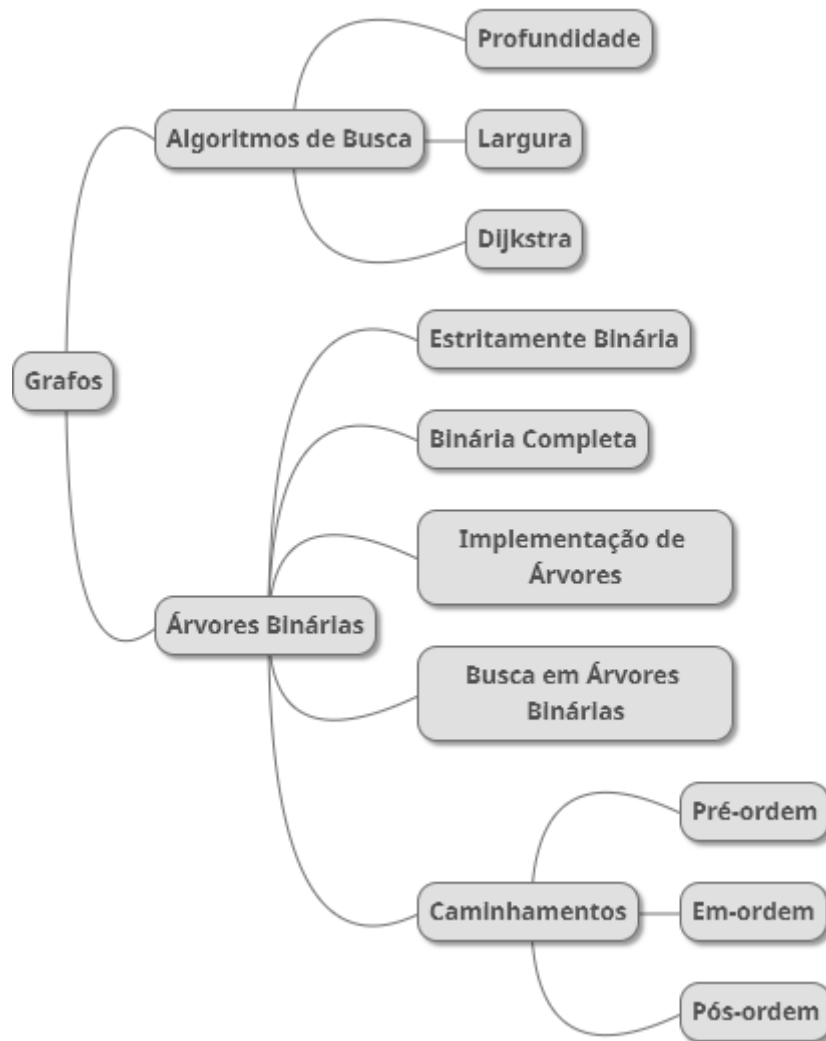
- a. Listas, pilhas e filas
b. Algoritmos de ordenação:



- c. Algoritmos de busca:



d. Árvores e Grafos:



5. Procure por listas de exercícios, em livros ou na internet, que sigam a seguinte estrutura geral:

- a. Dá-se um código fonte / pseudocódigo
 - i. Pede-se teste de mesa (resultado), ou;
 - ii. Pergunta-se se há erro no código, oi;
 - iii. Pede-se para completar o código (em caso de códigos incompletos, no enunciado).
- b. Dá-se um problema algorítmico
 - i. Pede-se o algoritmo (código) que resolve o problema