

# Seminário I - Escopo para a apresentação

## Estruturas de Dados Baseadas em Árvores

### Introdução (Parte 1 - Aluno 1)

**Tema:** Introdução a Árvores e Tipos de Árvores no Livro

**Conteúdo:**

#### 1. Definição de Árvores

- Árvores como estruturas hierárquicas (Capítulo B.5, página 1169).
- Terminologia: raiz, folhas, nós internos, altura, profundidade.

#### 2. Tipos de Árvores no Livro

- **Árvores Binárias** (Seção 10.3, página 265).
- **Árvores Binárias de Busca (BST)** (Capítulo 12, páginas 312-330).
  - Exemplo: Busca em BST (Figura 12.2).
- **Árvores vermelho-preto** (Capítulo 13, páginas 331-360).
  - Exemplo: Balanceamento após inserção (Figura 13.4).
- **B-Trees** (Capítulo 18, páginas 497-519).
  - Exemplo: Inserção em B-Tree (Figura 18.7).

#### 3. Aplicações Práticas

- BSTs em sistemas de banco de dados.
  - B-Trees em sistemas de arquivos.
- 

### Heaps (Parte 2 - Aluno 2)

**Tema:** Heaps e suas Propriedades

**Conteúdo:**

#### 1. Definição e Características

- Heap como árvore binária completa (Capítulo 6, página 161).
- Propriedades de max-heap e min-heap.

#### 2. Operações Básicas

- **Build-Max-Heap** (Seção 6.3, página 167).
  - Exemplo: Transformação em max-heap (Figura 6.3).
- **Heapify** (Seção 6.2, página 164).
  - Exemplo: Correção de violação (Figura 6.2).

#### 3. Aplicações

- Heapsort (Seção 6.4, página 170).
  - Exemplo prático: Ordenação de um array (Figura 6.4).
- 

## Filas de Prioridade (Parte 3 - Aluno 3)

**Tema:** Filas de Prioridade Baseadas em Heaps

**Conteúdo:**

### 1. Definição e Operações

- Filas de prioridade como abstração (Seção 6.5, página 172).
- **Insert** e **Extract-Max** (páginas 173-174).

### 2. Implementação com Heaps

- Aumentar prioridade (Seção 6.5, página 175).
- Exemplo: Atualização de prioridade em um hospital.

### 3. Aplicações

- Algoritmo de Dijkstra (Seção 22.3, página 620).
  - Sistemas de escalonamento (OS).
- 

## Comparação e Conclusão (Parte 4 - Aluno 4)

**Tema:** Análise Comparativa e Resumo

**Conteúdo:**

### 1. Comparação de Desempenho

- BST vs. Heaps vs. B-Trees (Tabela resumo).
- Complexidade: Inserção, Busca, Remoção.

### 2. Cenários de Uso

- Heaps para filas de prioridade.
- B-Trees para grandes volumes de dados.

### 3. Conclusão

- Síntese das vantagens de cada estrutura.
  - Perguntas para discussão em sala.
- 

## Conteúdos para a Reunião

- Divisão das tarefas
- Definição do template de apresentação (Slide)

- Questionamentos, sugestões e mais dicas.

## **Divisão de Tarefas:**

1. **Aluno 1:** Pesquisar e apresentar os conceitos básicos e tipos de árvores.
2. **Aluno 2:** Focar em Heaps (construção, operações e exemplos).
3. **Aluno 3:** Explorar filas de prioridade e aplicações práticas.
4. **Aluno 4:** Comparar estruturas e sintetizar o conteúdo.

## **Dicas:**

- Use figuras do livro para ilustrar
- Inclua pseudocódigo de forma dinâmica no slide
- Relacione com problemas reais