

2ºTrabalho de Estrutura de Dados
Docente: Jean Bertrand
Discente: Yan dos Santos Teixeira
Diferença dos Percursos nas Árvores ABB e AVL

Introdução

Neste projeto, foram implementadas três formas de percorrer as árvores binárias: em ordem, pré-ordem e pós-ordem, tanto para a Árvore Binária de Busca (ABB) quanto para a Árvore AVL. Estes percursos foram utilizados para observar e comparar a estrutura e o balanceamento das árvores após a inserção de dados.

Definição dos Percursos

Em Ordem (in-order):

Visita a subárvore esquerda, depois o nó atual, e por fim a subárvore direita.

Resultado em uma ABB: os valores saem em ordem crescente.

Pré-Ordem (pre-order):

Visita o nó atual, depois a subárvore esquerda e, por fim, a subárvore direita.

Útil para reconstrução da árvore ou exportação de estrutura.

Pós-Ordem (post-order):

Visita a subárvore esquerda, depois à direita, e por fim o nó atual.

Comum em algoritmos de liberação (free) ou avaliação de expressões.

Implementação no Código

As três funções de percurso foram implementadas nos dois arquivos (abb.c e avl.c) e chamadas a partir do main.c, utilizando a mesma lógica recursiva para ambos os tipos de árvore:

```
printf("ABB em ordem: ");
```

```
emOrdem(raizABB);
```

```
printf("\n");
```

```
printf("AVL em ordem: ");
```

```
emOrdem(raizAVL);
```

```
printf("\n");
```

Diferença Prática Observada

Em uma ABB, a estrutura da árvore depende da ordem de inserção. Se os dados forem inseridos em ordem crescente ou decrescente, a árvore pode se tornar desbalanceada, fazendo com que os percursos reflitam essa estrutura assimétrica.

Na AVL, a árvore se mantém automaticamente balanceada após cada inserção. Isso garante que os percursos percorrem uma estrutura mais equilibrada, com subárvores esquerda e direita com alturas mais próximas.

Exemplo Ilustrativo (dados inseridos: 10, 20, 30, 40, 50)

Percurso em Ordem:

ABB: 10 20 30 40 50 (correto, mas árvore está desbalanceada para a direita)

AVL: 10 20 30 40 50 (igual, porém a estrutura está balanceada)

Pré-Ordem:

ABB: 10 20 30 40 50 (mostra claramente a estrutura linear da árvore)

AVL: 30 20 10 40 50 (estrutura balanceada com rotações)

Pós-Ordem:

ABB: 50 40 30 20 10

AVL: 10 20 50 40 30

Conclusão

Os percursos em si são iguais em lógica para ABB e AVL, mas os resultados e estruturas percorridas diferem. Essa diferença demonstra na prática como o balanceamento automático da AVL impacta diretamente a profundidade, simetria e desempenho das operações, especialmente em grandes volumes de dados.