UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ DEPARTAMENTO ACADÊMICO DE INFORMÁTICA

# IMPLEMENTAÇÃO DE ÁRVORE BINÁRIA DE BUSCA PARA CRIAÇÃO DE UM DICIONÁRIO

Guilherme Aristides Marcos

GUILHERME ARISTIDES MARCOS

# IMPLEMENTAÇÃO DE ÁRVORE BINÁRIA DE BUSCA PARA CRIAÇÃO DE UM DICIONÁRIO

Projeto prático apresentado na disciplina de Estrutura de Dados II (CSF30), do Departamento Acadêmico de Informática da Universidade Tecnológica Federal do Paraná, *campus* Curitiba, como requisito de composição da nota final.

## INTRODUÇÃO

A Árvore Binária de Busca (ABB), é uma estrutura de dados que, com um conjunto potencialmente grande, permite realizar as operações de inserção, remoção e busca de elementos de forma “eficiente” no caso médio, que possui, em notação Big-O, a complexidade O(log n).

As principais vantagens do uso da ABB, são a alta velocidade de busca e também a possibilidade de percorrer a árvore de várias maneiras, ordenando-a de certa forma. Partindo do princípio da árvore de “separar” seus nós de acordo com as respectivas chaves, ou seja, a igualdade entre as chaves em análise. Isso a torna especialmente eficiente para conjuntos de dados grandes e desordenados, já que a inserção na árvore, se dá seguindo algumas regras.

## DESCRIÇÃO DO PROBLEMA

Tendo em vista que existem inúmeras estruturas de dados, cada uma com seus respectivos objetivos, a árvore binária de busca foi escolhida porque chamou a atenção ao lidar com grandes bases de dados, e principalmente, ao fazer isso de maneira eficiente. Foi decidido que seria interessante aplicar os conceitos da disciplina abordando-a.

A ideia do projeto consiste na criação de um dicionário utilizando os conceitos de árvore. Operações básicas de inserção e busca, ademais foi utilizado o percurso in ordem. Considerando isto, a árvore escolhida superou as expectativas, uma vez que além de atingir o objetivo também otimizou-o. Tornou-se possível, por exemplo, realizar buscas e ler uma base grande de dados muito rapidamente, além de exportar a árvore gerada, em ordem alfabética.

## ABORDAGEM DOS CONCEITOS VISTOS

Os conceitos vistos na disciplina de Estrutura de Dados 2 foram essenciais para o desenvolvimento do projeto sobre a árvore binária de busca. Os fundamentos de árvores contribuíram na escolha do tema do projeto, visto que, sabendo as árvores já estudadas na disciplina, procurava-se algo incomum, porém com semelhanças ao citado em sala de aula e que confirma toda a teoria vista. Dito isto, explica-se a decisão pela árvore ABB utilizando strings ao invés de números inteiros, como chaves., sendo assim a struct base da árvore ficou com os dois nós e também com duas strings (palavra e significado). Com essa alteração, foram necessários diversos ajustes em relação ao código da ABB trabalhada em aula, posto isso, o projeto corroborou para um aprofundamento na compreensão não só da árvore que foi aplicada, mas de árvores binárias e as estruturas de árvores como um todo.

## TESTES REALIZADOS

Para melhor compreensão dos testes, foi gravado um vídeo efetuando cada operação do programa (busca/inserção/leitura/exportação). Segue o link:

* + https://youtu.be/olzDkXuW\_bI

## SOLUÇÕES ENCONTRADAS

A ideia inicial era aplicação de um dicionário, por meio de uma árvore que comportasse duas cadeias de caracteres, que se referem a palavras/termos e também aos seus respectivos significados. Depois de pesquisas, a árvore ABB e tabelas Hash aparentavam conciliar-se com o tema, e após a escolha pela árvore ABB, então começou a tentativa de implementação, iniciando com conceitos já conhecidos e buscando uma melhora através do estudo de novos conceitos, tanto de árvores como da linguagem em si.

Várias dificuldades surgiram ao longo do desenvolvimento do trabalho, mais especificamente, na função de inserção e na leitura e obtenção da base de dados.

Na inserção, houve um problema em manter a raiz intacta, para resolver esse problema, a raiz foi definida como um nó global da árvore, entretanto outro problema surgiu, a função de verificação se a palavra já estava no dicionário, retornava sempre verdadeiro, não possibilitando o processo de inserção, para resolver esse problema se mostrou necessário resetar o nó temporário que auxilia na inserção na árvore.

Em seguida, na operação de leitura da base de dados, a dificuldade foi em relação a passar as informações lá presentes, respeitando o padrão lá imposto. Com isso após análise do arquivo que estava no formato ‘.json’ foi notado a presença de um carácter chave, no caso as aspas (“), com isso a leitura ocorreu em função desse caráter. Outro fator notado foi a presença de palavras/termos com iniciais em letras minúsculas e os significados com letra maiúscula, sendo assim esse foi outro fator a se considerar para saber em quais das duas strings de cada nó guardar a informação.

Algumas outras adversidades apareceram, como por exemplo a base de dados no idioma inglês, já que a ideia inicial era um dicionário na língua portuguesa, também foi realizada a tentativa de deixar a raiz como um nó não global, porém o programa apresentou uma série de problemas, então optou-se pela solução inicial.