

## Relatório Árvore PATRICIA

Rafael Rodrigues Monteiro  
Universidade do Estado de Minas Gerais  
Engenharia da Computação, 3º período 2022  
rafael.1697550@discente.uemg.br

**PALAVRAS-CHAVE:** árvore; patrícia; digital;

### 1. INTRODUÇÃO

O objetivo principal do trabalho é relatar o conhecimento adquirido durante as pesquisas sobre a árvore digital patrícia. O presente relatório visa abordar a definição de uma árvore patrícia, suas formas de implementação e suas aplicações.

### 2. DEFINIÇÃO DE UMA ÁRVORE DIGITAL PATRÍCIA

O algoritmo PATRICIA é um método de busca em árvores criado em 1968 por Donald R. Morrison. Ele é uma representação compacta de uma árvore TRIE, diferentemente das TRIEs, ela não armazena informações nos nodos internos, apenas contadores e ponteiros para cada sub-árvore descendente. O nome do algoritmo vem de *Practical Algorithm To Retrieve Information Coded in Alphanumeric*. O método é extremamente eficaz e útil quando utilizado para representar chaves de tamanho variável e longas, como títulos e frases. A motivação inicial que Donald teve para a criação do código foi a otimização de busca em arquivos de bibliotecas.

### 3. PROPRIEDADES

#### 3.1 - Espaço

Diferentemente do algoritmo TRIE, a árvore PATRICIA não contém nós com apenas um filho. Cada um de seus nós possui dois filhos ou é um nó folha. Com isso o seu número de nós internos não ultrapassa o numero de nós. Cada nó é capaz de armazenar uma string, ao contrário da árvore TRIE que armazena apenas um caractere por nó.

#### 3.2 – Prefixos

A árvore PATRICIA faz uso de uma operação conhecida como correspondência por prefixos, tal operação é responsável por retornar uma lista com todas as strings que possuem determinado prefixo.

## 4. FORMAS DE IMPLEMENTAÇÃO

### 4.1 – Busca

Realizar uma busca em uma árvore PATRICIA é similar a uma busca em um algoritmo TRIE, com a diferença de que ao chegar em um nó, é comparado apenas um caractere, indo contra a comparação de substrings inteiras que acontece no algoritmo TRIE. No pior caso, a complexidade do algoritmo em relação a notação *Big O* é de  $O(|s|)$ , sendo *s* é a palavra procurada.

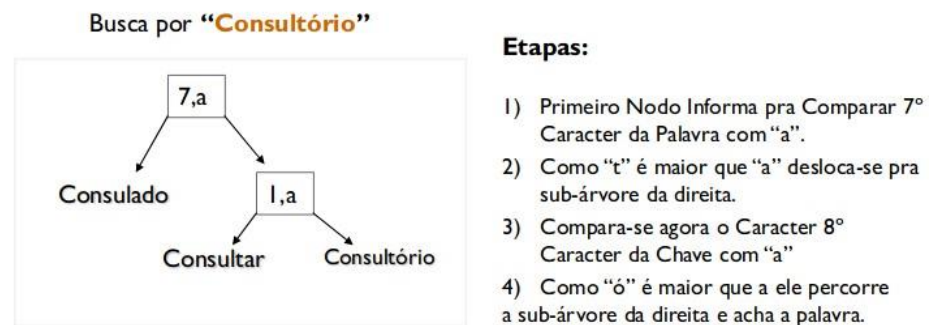


Figura 1: Busca na árvore

### 4.2 Inserção

A inserção em uma árvore PATRICIA inicialmente segue os passos da função de busca até o momento em que a busca é encerrada, pois a string procurada não é encontrada na árvore. Se a busca se encerrar em uma aresta, um novo nó é criado e irá armazenar a posição do caractere que distingue

Se a busca é encerrada em uma aresta, um novo nó é criado nessa aresta. Esse nó armazena a posição do caractere que distingue a chave destino daquela aresta e a chave que se deseja inserir, e tem como filhos o nó que estava na extremidade seguinte da aresta e um novo nó com a parte restante da nova chave. Se a busca for encerrada em um nó, então um nó filho é criado e o restante da nova chave é usado como rótulo para aresta entre os dois.

Ambos os casos de inserção possuem a complexidade do algoritmo em relação a notação *Big O* de  $O(|s| + |E|)$ , sendo *s* a palavra a ser inserida na árvore e *E* o números de caracteres suportados pelo alfabeto da árvore.

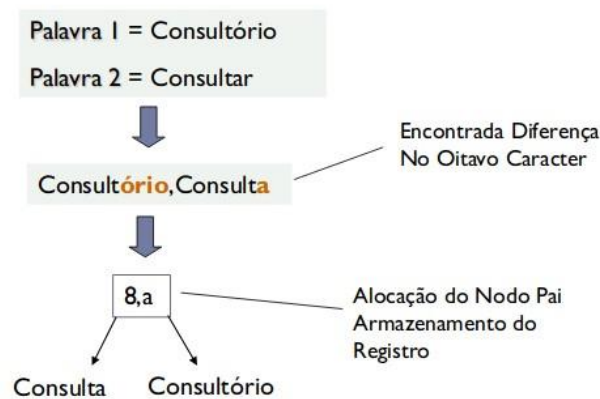


Figura 2: Inserção na árvore

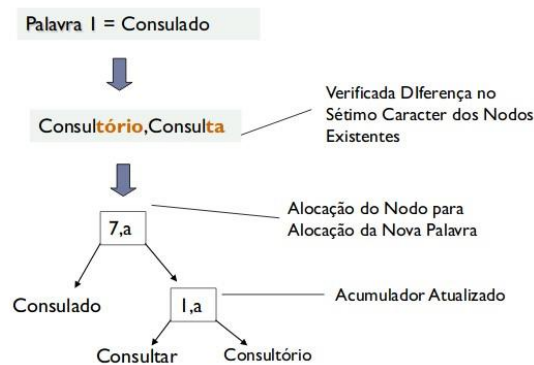


Figura 3: Inserção na árvore

### 4.3 Remoção

A remoção em uma árvore PATRÍCIA ocorre de forma oposta ao algoritmo de inserção, primeiro localizamos a folha correspondente a string e a removemos da árvore. Em seguida, como o pai terá apenas um filho, os nós pai e irmão do nó removido são agrupados formando um único nó. Em termos de complexidade seguindo a notação *Big O* temos que:

A complexidade de tempo depende diretamente do tempo para remover 2 nós da árvore, se essa remoção for considerada linear, então a complexidade de tempo da operação é  $O(|s|)$ , onde  $s$  é a string que será removida, se essa remoção tiver complexidade  $O(N)$ , então a complexidade de tempo da operação é  $O(|s| + N)$ , onde  $N$  é o tamanho total de todas as strings armazenadas na árvore.

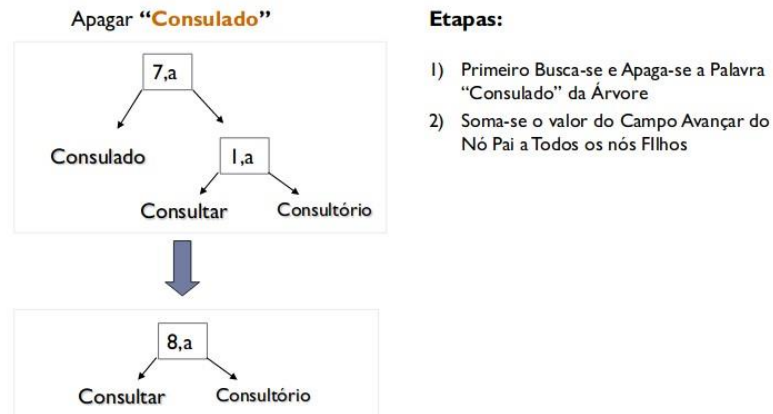


Figura 4: Remoção na árvore

## 5. APLICAÇÕES DE UMA ÁRVORE DIGITAL PATRÍCIA

As árvores TRIEs quando implementadas na indexação de textos, geram um grande o número de chaves, impossibilitando que esses dados possam ser armazenados na memória principal. As árvores PATRÍCIA contornam esse problema devido ao seu nível de compactação, fazendo com que as árvores tenham uma altura menor e consequentemente diminuindo o tempo de acesso.

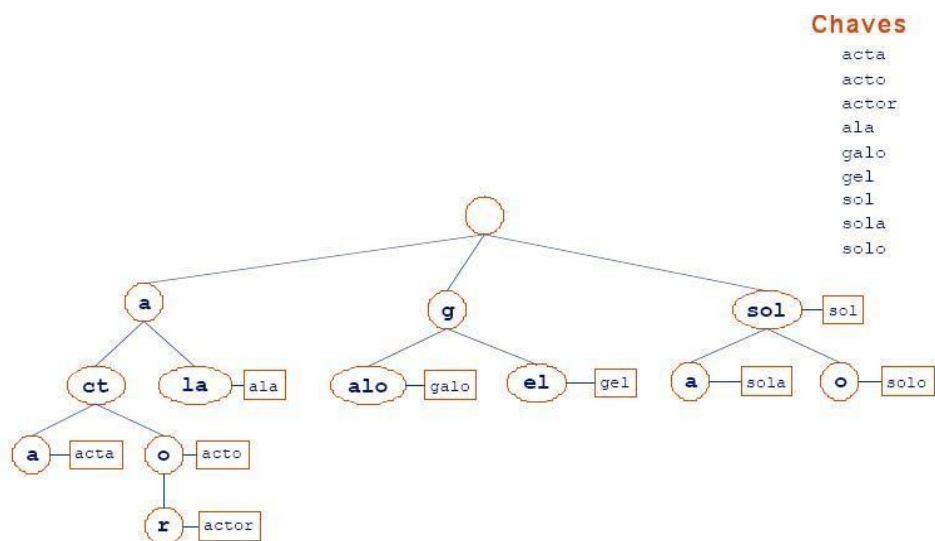


Figura 5: Demonstração de árvore uma PATRÍCIA

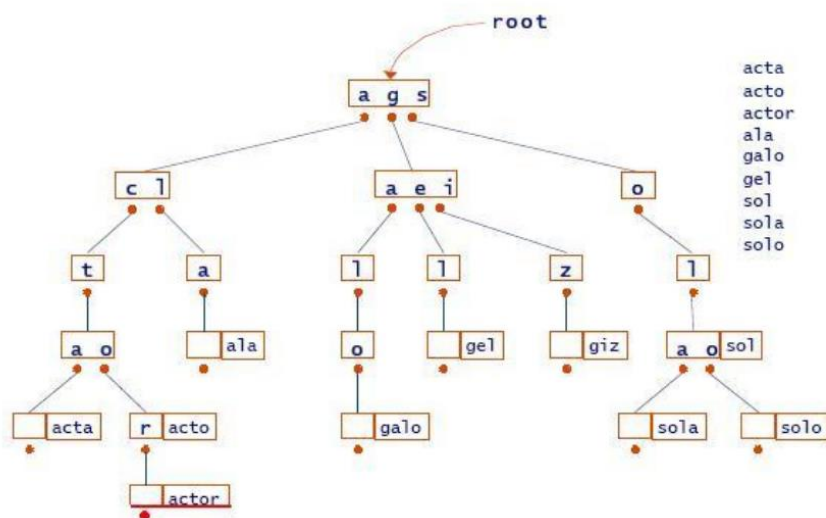


Figura 6: Demonstração de uma árvore TRIE

Logo, indexação de textos é uma das aplicações de árvores PATRÍCIA, uma de suas aplicações mais comuns é o corretor ortográfico, nesse tipo de programa (corretor ortográfico) as palavras são comparadas com um dicionário armazenado em arquivo, e se não são encontradas indica-se as opções para correção.

O algoritmo PATRÍCIA também é utilizado em aplicações de autopreenchimento de textos. Podemos observar essas aplicações desde browsers a serviços de e-mail. A cada caractere digitado, são apresentadas as opções de preenchimento, e quando só existir um caminho possível a ser seguido na árvore ocorre o preenchimento automático.

Dentre outras aplicações que também utilizam o algoritmo podemos citar:

- Banco de dados;
- Busca em redes P2P;
- Ganho de eficiência com documentos XML

## 6. CONCLUSÃO

Portanto, através deste relatório foi possível conhecer a definição de uma árvore TRIE e um pouco sobre sua criação. 3 funções principais também foram implementadas em forma de pseudocódigo com suas devidas descrições afim de melhorar o entendimento.

Concluimos também que uma árvore PATRICIA é mais eficiente que uma árvore TRIE devido ao seu nível de compactação. Portanto o algoritmo utiliza menos memória e processamento.

## 7. REFERÊNCIAS

Wikipédia. **Árvore Patrícia**. Disponível em:

<[https://pt.wikipedia.org/wiki/%C3%81rvore\\_Patrici](https://pt.wikipedia.org/wiki/%C3%81rvore_Patrici)

a> Acessado em 21 de junho de 2022.

Bueno, Marcio. **Árvores Trie e Patrícia**. Disponível em:

<[https://marciobueno.com/arquivos/ensino/ed2/ED2\\_06\\_Trie\\_Patricia.p](https://marciobueno.com/arquivos/ensino/ed2/ED2_06_Trie_Patricia.pdf)

df> Acessado em 21 de junho de 2022.

MORRISON, DONALD R. **PATRICiA--Practical Algorithm to Retrieve Information Coded in Alphanumeric**. Res. Rep. SC-RR-67-734, Sandia Corp., Albuquerque, N. Mex., Oct. 1967.