

## Árvore 2-3

Luigi Wagner  
Rafael Alessandro  
Rafael Falcão

Ciências da Computação – Disciplina: Estruturas de Dados – 1  
Instituto de Informática  
Universidade Federal de Goiás

6 de novembro de 2017

# Sumário

- 1 Introdução
- 2 Códigos
- 3 Terceiro Tópico
- 4 Questionário
- 5 Referências Bibliográficas

# Introdução

Uma Árvore 2-3 é uma árvore onde cada nó com filho (nó interno) tem também 2 filhos (2-node) e 1 elemento de dados (chave) ou 3 filhos (3-nodes) e 2 elementos de dados (chaves). Os nós externos a árvore (nós-folha) não tem filhos e possuem um ou dois elementos de dados (chaves).

# Introdução

## Propriedades

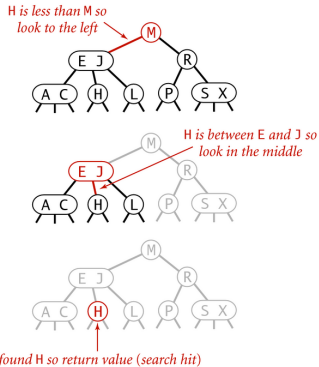
As principais propriedades de uma Árvore 2-3 são:

- Cada nó interno tem dois filhos (2-node) se tem uma chave, ou três filhos (3-node) se tem duas chaves;
- Cada nó não-folha tem 2 ou 3 filhos. Se tem 2 filhos tem 1 item de dados e se tem 3 filhos tem 2 itens de dados;
- Todos os dados são ordenados;
- Todas as folhas estão no mesmo nível;
- Cada nó folha tem 1 ou 2 campos.

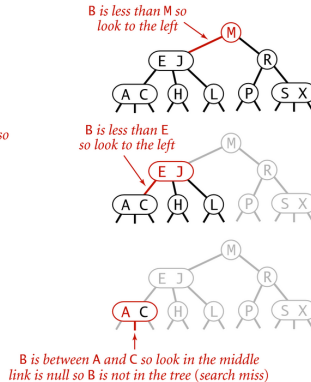
Como, por exemplo:

# Introdução

## successful search for H



## unsuccessful search for B



Search hit (left) and search miss (right) in a 2-3 tree

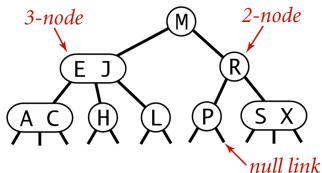
Figura: <https://www.ime.usp.br/~pf/estruturas-de-dados/aulas/figuressw/Chapter3/TTsearch.png>

# Introdução

## Componentes ou Anatomia

Uma Árvore 2-3 é composta por:

- 1 Componente 1;
- 2 Componente 2;
- 3 Componente 3.



Anatomy of a 2-3 search tree

Figura: <https://www.ime.usp.br/~pf/estruturas-de-dados/aulas/figuressw/Chapter3/TTanatomy.png>

# Introdução

## Bloco

Um bloco pode conter figuras, tabelas, listas, etc.

# Sumário

- 1 Introdução
- 2 Códigos**
- 3 Terceiro Tópico
- 4 Questionário
- 5 Referências Bibliográficas



## Programas

Agora que já vimos como funciona e os princípios básicos das [Árvores 2-3](#), daremos uma olhada nas estruturas e algoritmos para a manipulação e armazenamento das supracitadas [Árvores 2-3](#).

- 1 Estrutura básica ([Struct](#));
- 2 Alocação da [Árvore 2-3](#).

# Código

## Estrutura de uma Árvore 2-3

```
1 typedef struct _23tree_link {  
2     struct 23tree_node *node; //internal node  
3     int item;  
4 } 23tree_link;
```

# Código

## Alocação de uma Árvore 2-3

```
1 23tree *23tree_alloc() {
2     23tree *t;
3     23tree_node *r;
4
5     t = malloc(sizeof(23tree));
6     t->n = 0;
7     t->min_item = NULL;
8     t->stack = malloc(STACK_SIZE * sizeof(23tree_node *));
9     r = t->root = malloc(sizeof(23tree_node));
10    r->key1 = r->key2 = NULL;
11    r->link_kind = LEAF;
12    r->left.item = r->middle.item = r->right.item;
13
14    return t;
15 }
```

# Sumário

- 1 Introdução
- 2 Códigos
- 3 Terceiro Tópico**
- 4 Questionário
- 5 Referências Bibliográficas

# Terceiro Tópico

## Tabelas

Qualquer recursos disponível no LaTeX e seus pacotes complementares podem ser utilizados, desde que ao final a equipe gere um arquivo compactado (extensão .zip) contendo toda a “*pasta*” com os *slides* desenvolvidos.

A pasta gerada também deverá conter um arquivo .pdf com os *slides* utilizados.

# Terceiro Tópico

## Exemplo 01

Exemplos devem utilizar um bloco para serem apresentados, como o que está acontecendo neste momento, ou seja, um bloco está sendo utilizado e seu título é Exemplo 01.

Você também pode utilizar [cores](#) para dar destaque no texto.

# Sumário

- 1 Introdução
- 2 Códigos
- 3 Terceiro Tópico
- 4 Questionário**
- 5 Referências Bibliográficas

# Questionário

Deve haver, ao final, uma seção dedicada à apresentação de um questionário com pelo menos 05 (cinco) questões, e suas respectivas respostas esperadas.

Cada questão deverá ser um bloco, o mesmo ocorrendo com a resposta esperada.

Veja o exemplo....



# Questionário

## Questão 01

Enunciado da primeira questão, se necessário com figuras, tabelas, e tudo mais.

# Questionário

## Questão 01 – Resposta Esperada

Bloco com a resposta esperada para a primeira questão.

# Sumário

- 1 Introdução
- 2 Códigos
- 3 Terceiro Tópico
- 4 Questionário
- 5 Referências Bibliográficas

# Referências Bibliográficas

- [https://pt.wikipedia.org/wiki/%C3%81rvore\\_2-3](https://pt.wikipedia.org/wiki/%C3%81rvore_2-3)
- COOPER, K. and TORCZON, L. – 2011 : *Chapter 2 – Scanners*;
- GRUNE, et al. – 2012 : *Chapter 2 – Program Text to Tokens – Lexical Analysis*.

# Referências Bibliográficas

Se desejar, pode encerrar com uma imagem, uma citação, etc.



Figura: by Mark Kostabi