



Árvore Balanceada Rubro-Negra

Estrutura de Dados II

Fabiano Rafael, Guilherme Piovesan e Ryan Alves.

Introdução às Árvores Rubro-Negras

- **Definição:**

- As Árvores Rubro-Negras são um tipo de árvore binária de busca balanceada.

- **Propriedades:**

- Cada nó é vermelho ou preto.
- A raiz é sempre preta.
- Todas as folhas (nós nulos) são pretas.
- Se um nó é vermelho, então ambos os filhos são pretos (não pode haver dois nós vermelhos consecutivos).
- Todo caminho de um nó até suas folhas nulas contém o mesmo número de nós pretos.

Operações Básicas

- **Inserção:**

- Nós novos são sempre inseridos como nós vermelhos.
- Após a inserção, o balanceamento é garantido através de rotações e recoloração.

- **Remoção:**

- Similar à remoção em árvores binárias de busca, seguida de ajustes para manter as propriedades da árvore.

- **Rotação e Recoloração:**

- Rotações à esquerda e à direita são usadas para reestruturar a árvore.
- Recoloração é feita para garantir que as propriedades sejam mantidas.

Implementação em Python

- Classes Principais:
- **'TNo'**: Representa um nó da árvore, com atributos para cor, chave e referências aos filhos e ao pai.
- **'TArvore'**: Gerencia a raiz da árvore e contém métodos para inserção, remoção e rotação.

Implementação em Python

- Funções Importantes:

- **‘rotacao_esq’ e ‘rotacao_dir’** : Executam rotações necessárias para manter o balanceamento.
- **Inserção:** A função de inserção adiciona um novo nó vermelho na árvore e, em seguida, ajusta a estrutura com rotações e recolorações para manter suas propriedades de balanceamento e cor.
- **Remoção:** A função de remoção elimina um nó da árvore e, em seguida, ajusta a estrutura com rotações e recolorações para preservar as propriedades de balanceamento e cor da Árvore Rubro-Negra.



Árvore Balanceada Rubro-Negra

Estrutura de Dados II

Fabiano Rafael, Guilherme Piovesan e Ryan Alves.