

Prática 03 - AEDS 2

Henrique Freitas

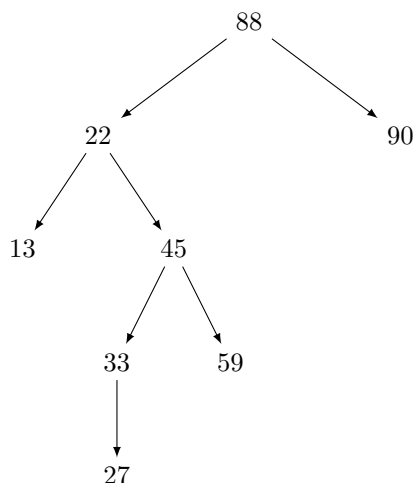
October 2024

Questão 1: Construção de Árvores Binárias de Busca

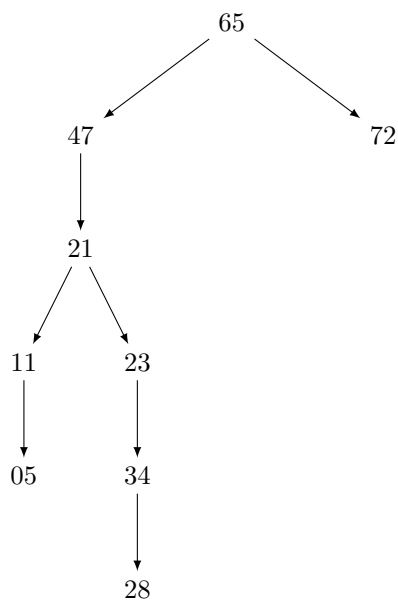
Considerando os conceitos de inserção, remoção, pesquisa e caminhamento em árvores binárias, resolvemos as seguintes questões com base nos conjuntos de dados apresentados:

1. Construa as árvores binárias de busca a partir dos conjuntos abaixo e desenhe a estrutura da árvore após cada inserção de k elementos.

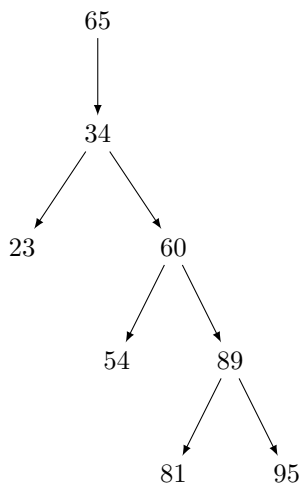
Árvore 1: {88, 22, 45, 33, 22, 90, 27, 59, 13}



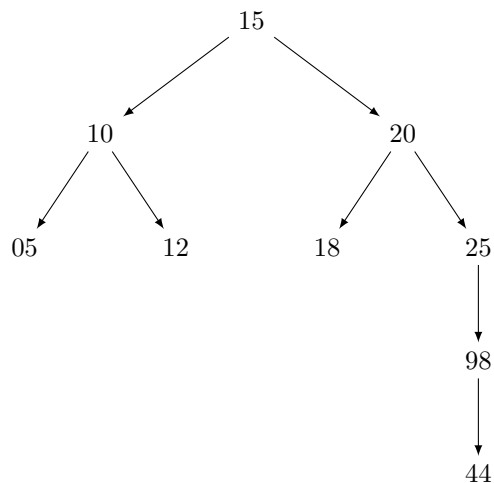
Árvore 2: {65, 47, 21, 11, 72, 23, 05, 34, 28}



Árvore 3: {65, 34, 89, 23, 60, 54, 81, 95, 39}



Árvore 4: {15, 10, 20, 05, 12, 18, 25, 98, 44}

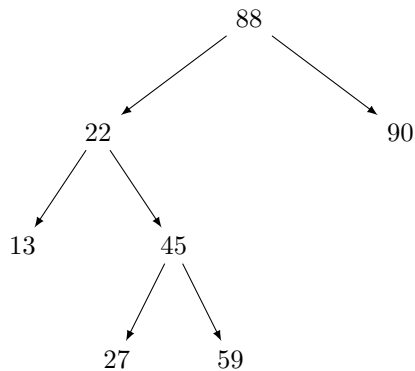


Questão 2: Remoção de Elementos em Árvores Binárias de Busca

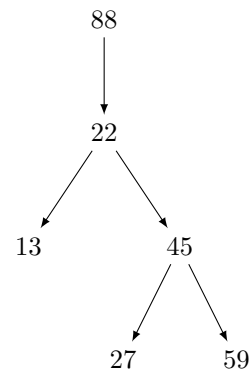
Nesta questão, realizamos a remoção dos elementos indicados e redesenhamos as árvores após cada remoção. Discutimos o impacto estrutural, abordando os diferentes casos de remoção (remoção de folha, remoção de nó com um filho e remoção de nó com dois filhos). Além disso, justificamos a escolha entre o sucessor in-ordem ou o predecessor in-ordem para os casos de remoção de nós com dois filhos.

Árvore 1: {33, 90, 33, 45}

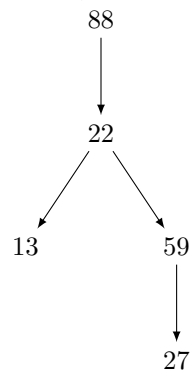
Remoção de 33: O nó 33 tem um filho (27). Ao removermos o nó com um filho, movemos o filho (27) para a posição de 33.



Remoção de 90: O nó 90 é uma folha, então sua remoção não impacta a estrutura da árvore.

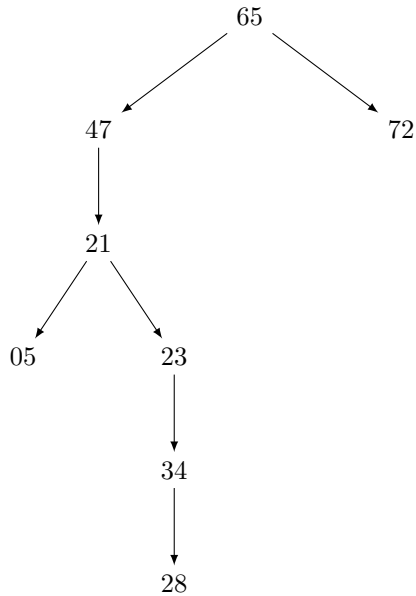


Remoção de 45: O nó 45 tem dois filhos (27 e 59). Para a remoção de um nó com dois filhos, escolhemos o sucessor, que é o nó 59, e substituímos 45 por 59.

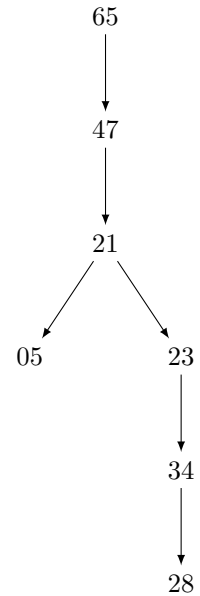


Árvore 2: {11, 72, 65, 23}

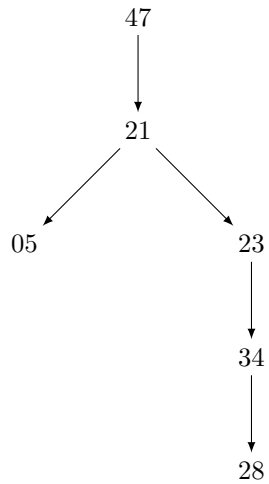
Remoção de 11: O nó 11 é uma folha. Sua remoção não afeta a estrutura da árvore.



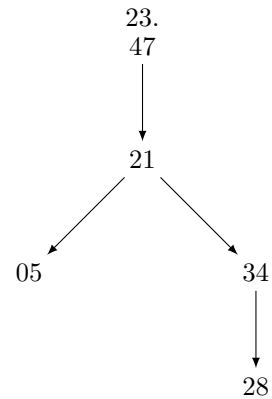
Remoção de 72: O nó 72 é uma folha, então a remoção não afeta a estrutura.



Remoção de 65: O nó 65 tem dois filhos (47 e 72). Escolhemos o sucessor in-ordem (47) para substituir 65.

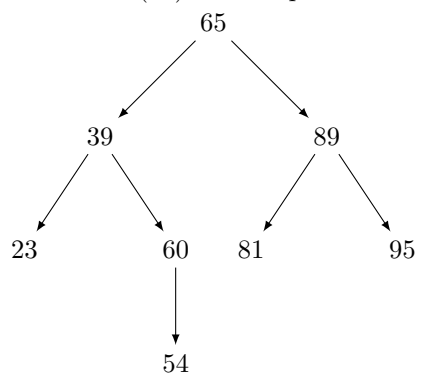


Remoção de 23: O nó 23 tem um filho (34). Removemos o nó e movemos 34 para a posição de

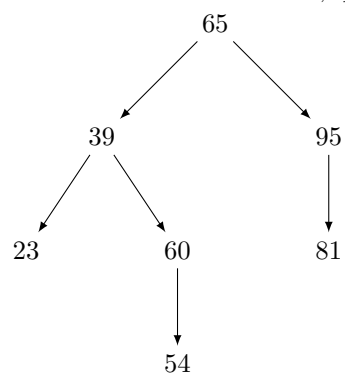


Árvore 3: {34, 89, 81, 95}

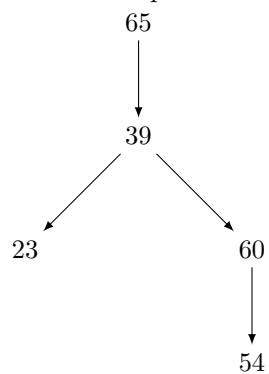
Remoção de 34: O nó 34 tem dois filhos. O sucessor in-ordem (39) é usado para substituir 34.



Remoção de 89: O nó 89 tem dois filhos (81 e 95). Escolhemos o sucessor in-ordem, que é o 95.

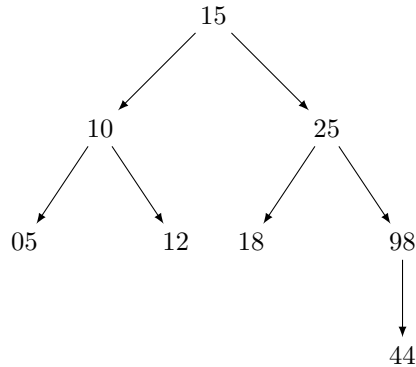


Remoção de 81 e 95: Ambas são folhas e são removidas sem impacto estrutural.

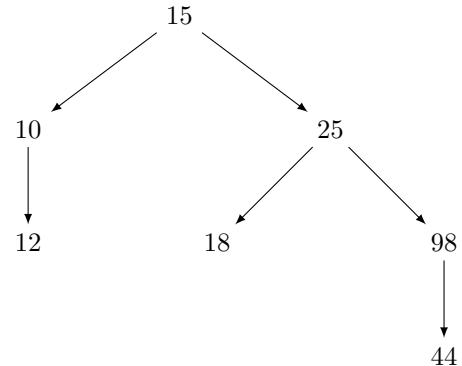


Árvore 4: {20, 05, 18, 44}

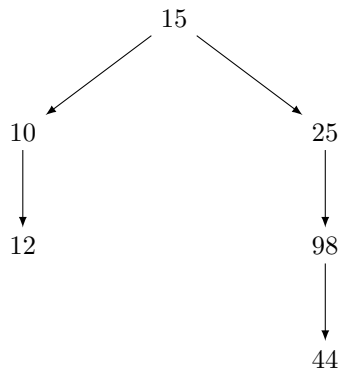
Remoção de 20: O nó 20 tem dois filhos (18 e 25). Para a remoção de um nó com dois filhos, escolhemos o sucessor in-ordem, que é o nó 25, para substituir o 20.



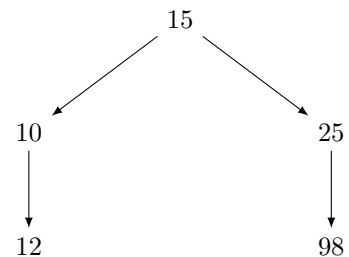
Remoção de 05: O nó 05 é uma folha, então sua remoção não afeta a estrutura da árvore.



Remoção de 18: O nó 18 é uma folha, então sua remoção não afeta a estrutura.



Remoção de 44: O nó 44 é uma folha, então sua remoção também não afeta a estrutura da árvore.

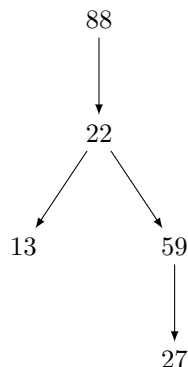


1 Pesquisa em Árvore com Diferentes Tipos de Caminhamentos

Nesta seção, utilizamos quatro tipos de caminhamento para localizar um elemento específico na árvore binária: pré-ordem, inordem, pós-ordem e em largura. Vamos analisar o número de interações e a ordem de visitação dos nós até encontrar o elemento selecionado.

1.1 Árvore de Exemplo

Abaixo está a representação gráfica da árvore binária utilizada para esta análise:



Vamos selecionar o nó **59** para ser localizado em cada um dos percorrimentos.

1.2 Percorrimento Inordem

No percorrimto inordem (ou em ordem), visitamos o nó mais à esquerda, depois a raiz e, em seguida, os nós à direita. A ordem de visitação dos nós na árvore seria:

13 → 22 → 27 → 59 → 88

Para localizar o elemento **59**, o caminho percorrido seria: 13, 22, 27, 59. Foram necessárias **4 interações** até encontrar o elemento.

1.3 Percorrimento Pós-ordem

No percorrimto pós-ordem, primeiro visitamos todos os filhos de um nó e, em seguida, o próprio nó. A ordem de visitação dos nós seria:

13 → 27 → 59 → 22 → 88

O caminho até o elemento **59** seria: 13, 27, 59. Foram necessárias **3 interações** para encontrar o elemento.

1.4 Percorrimento em Largura

O percorrimto em largura visita os nós em cada nível da árvore da esquerda para a direita. A ordem de visitação seria:

88 → 22 → 13 → 59 → 27

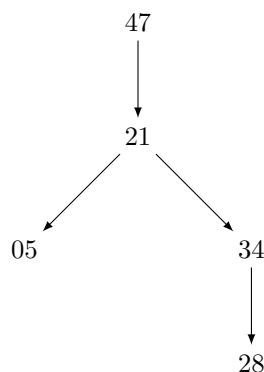
Neste caso, o elemento **59** foi encontrado após **4 interações**, com o caminho: 88, 22, 13, 59.

1.5 Eficiência dos Métodos

Podemos observar que os diferentes tipos de caminhamento apresentam diferentes números de interações. O percorrimto pós-ordem foi o mais eficiente para encontrar o elemento **59**, seguido pelo inordem e, por último, o em largura.

1.6 Percorrimento de Pesquisa na Árvore: Exemplo 2

Utilizaremos a seguinte árvore binária para realizar o estudo dos diferentes tipos de caminhamento. O nó escolhido para ser localizado será o **34**.



1.6.1 Percorrimento Inordem

No percorrimto inordem (ou em ordem), visitamos o nó mais à esquerda, depois a raiz e, em seguida, os nós à direita. A ordem de visitação dos nós na árvore seria:

05 → 21 → 28 → 34 → 47

Para localizar o elemento **34**, o caminho percorrido seria: 05, 21, 28, 34. Foram necessárias **4 interações** até encontrar o elemento.

1.6.2 Percorrimento Pós-ordem

No percorrimto pós-ordem, primeiro visitamos todos os filhos de um nó e, em seguida, o próprio nó. A ordem de visitação dos nós seria:

05 → 28 → 34 → 21 → 47

O caminho até o elemento **34** seria: 05, 28, 34. Foram necessárias **3 interações** para encontrar o elemento.

1.6.3 Percorrimento em Largura

No percorrimto em largura, visitamos os nós em cada nível da árvore da esquerda para a direita. A ordem de visitação seria:

47 → 21 → 05 → 34 → 28

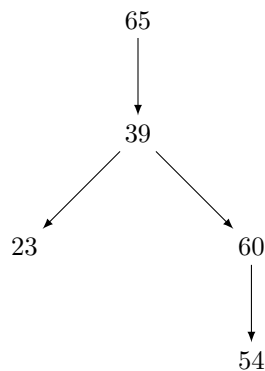
Neste caso, o elemento **34** foi encontrado após **4 interações**, com o caminho: 47, 21, 05, 34.

1.6.4 Eficiência dos Métodos

Analisando os diferentes tipos de caminhamento, observamos que o método pós-ordem é o mais eficiente para localizar o elemento **34**, seguido pelo inordem e, por último, o em largura.

1.7 Percorrimento de Pesquisa na Árvore: Exemplo 3

Utilizaremos a seguinte árvore binária para realizar o estudo dos diferentes tipos de caminhamento. O nó escolhido para ser localizado será o **60**.



1.7.1 Percorrimento Inordem

No percorrimto inordem (ou em ordem), visitamos o nó mais à esquerda, depois a raiz e, em seguida, os nós à direita. A ordem de visitação dos nós na árvore seria:

23 → 39 → 54 → 60 → 65

Para localizar o elemento **60**, o caminho percorrido seria: 23, 39, 54, 60. Foram necessárias **4 interações** até encontrar o elemento.

1.7.2 Percorrimento Pós-ordem

No percorrimto pós-ordem, primeiro visitamos todos os filhos de um nó e, em seguida, o próprio nó. A ordem de visitação dos nós seria:

23 → 54 → 60 → 39 → 65

O caminho até o elemento **60** seria: 23, 54, 60. Foram necessárias **3 interações** para encontrar o elemento.

1.7.3 Percorrimento em Largura

No percorrimto em largura, visitamos os nós em cada nível da árvore da esquerda para a direita. A ordem de visitação seria:

65 → 39 → 23 → 60 → 54

Neste caso, o elemento **60** foi encontrado após **4 interações**, com o caminho: 65, 39, 60.

1.7.4 Eficiência dos Métodos - Exemplo 3

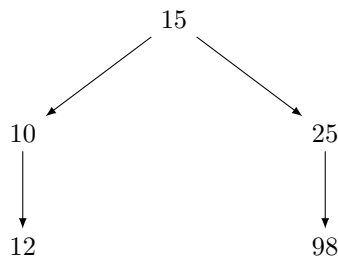
Exemplo 3 (Nó-alvo: 60)

- **Percorrimento Inordem:** Foram necessárias **4 interações**.
- **Percorrimento Pós-ordem:** Foram necessárias **3 interações**.
- **Percorrimento em Largura:** Foram necessárias **4 interações**.

No exemplo 3, o método **pós-ordem** foi o mais eficiente, localizando o nó **60** com apenas **3 interações**. O inordem e o em largura empataram, necessitando de **4 interações**.

1.8 Percorrimento de Pesquisa na Árvore: Exemplo 4

Utilizaremos a seguinte árvore binária para realizar o estudo dos diferentes tipos de caminhamento. O nó escolhido para ser localizado será o **98**.



1.8.1 Percorrimento Inordem

No percorrimento inordem (ou em ordem), visitamos o nó mais à esquerda, depois a raiz e, em seguida, os nós à direita. A ordem de visitação dos nós na árvore seria:

10 → 12 → 15 → 25 → 98

Para localizar o elemento **98**, o caminho percorrido seria: 10, 12, 15, 25, 98. Foram necessárias **5 interações** até encontrar o elemento.

1.8.2 Percorrimento Pós-ordem

No percorrimento pós-ordem, primeiro visitamos todos os filhos de um nó e, em seguida, o próprio nó. A ordem de visitação dos nós seria:

12 → 10 → 98 → 25 → 15

O caminho até o elemento **98** seria: 12, 10, 98. Foram necessárias **3 interações** para encontrar o elemento.

1.8.3 Percorrimento em Largura

No percorrimento em largura, visitamos os nós em cada nível da árvore da esquerda para a direita. A ordem de visitação seria:

15 → 10 → 25 → 12 → 98

Neste caso, o elemento **98** foi encontrado após **4 interações**, com o caminho: 15, 10, 25, 98.

1.8.4 Eficiência dos Métodos - Exemplo 4

Exemplo 4 (Nó-alvo: 98)

- **Percorrimento Inordem:** Foram necessárias **5 interações**.
- **Percorrimento Pós-ordem:** Foram necessárias **3 interações**.
- **Percorrimento em Largura:** Foram necessárias **4 interações**.

No exemplo 4, assim como no exemplo anterior, o método **pós-ordem** também foi o mais eficiente, localizando o nó **98** com **3 interações**. O percorrimento em largura foi intermediário, com **4 interações**, enquanto o inordem precisou de **5 interações**.