

AED2 2025 - AP05 - ARVORES AVL

Instruções:

1. E/S: tanto a entrada quanto a saída de dados devem ser "secas", ou seja, não devem apresentar frases explicativas. Siga o modelo fornecido e apenas complete as partes informadas (veja o exemplo abaixo);
2. Identificadores de variáveis: escolha nomes apropriados;
3. Documentação: inclua cabeçalho, comentários e indentação no programa;
4. Submeta o programa no sistema judge. Turma Integral:
`https://judge.unifesp.br/AED2IA012025`, Turma noturna: `https://judge.unifesp.br/AED2N012025`;
5. O código-fonte pode ser escrito em C ou C++.

Descrição:

Implemente as operações básicas de um algoritmo de pesquisa para árvores do tipo AVL. Seu programa deverá implementar os seguintes procedimentos: (1) inicialização, (2) pesquisa, (3) inserção, (4) remoção, (5) calcular altura do nó e (6) encontrar nós dentro de uma faixa de valores. Para isso, considere:

- Na operação de remoção, quando necessário, dê a preferência para a promoção da menor chave da subárvore à direita, ou seja, o sucessor;
- A altura de um nó x em uma AVL é dada pela distância entre x e o seu descendente mais afastado, ou seja, a altura de x é o número de passos no mais longo caminho que leva x até um nó folha.

Lembre-se que, cada inserção em uma árvore AVL requer que se verifique o fator de balanceamento e, se necessário, uma rotação é realizada. As remoções também podem demandar rotações para manter a árvore balanceada.

Considere as seguintes condições:

1. A complexidade de cada procedimento implementado deve a mesma apresentada em aula;
2. O código-fonte **deve** ser escrito em C/C++ ou Java;
3. **Toda** memória alocada dinamicamente (C/C++) deve ser desalocada;

Solução que violem essas condições **não** serão aceitas.

ENTRADA:

A primeira linha da entrada consiste de uma lista de números inteiros positivos separados por espaços. Esses números devem ser inseridos na árvore. A sequência termina com um número inteiro negativo que não deve ser inserido.

A segunda linha contém outra lista de números inteiros positivos separados por espaços. Assim como na lista da primeira linha, um número inteiro negativo marca o

final da lista. Cada um desses números, com exceção do número negativo, deve-se ser usado como chave de pesquisa na árvore. No caso de uma pesquisa sem sucesso, ou seja, o número não está na árvore, o número pesquisado deverá ser inserido na árvore. Se a pesquisa for bem sucedida, ou seja, o número está na árvore, o número deverá ser removido.

A terceira linha de entrada contém dois valores inteiros positivos. Esses dois valores serão a faixa de dados a serem pesquisados na árvore. Essa faixa de valores pode ou não estarem contidos na árvore.

S A Í D A:

A primeira linha da saída contém a altura máxima da ABB a partir do seu nó raiz seguida da altura da sub-árvore da esquerda e da direita do nó raiz. Esses valores devem ser calculados considerando apenas a árvore construída com os números da primeira linha da entrada.

A segunda linha demonstra o mesmo princípio da primeira linha, porém, se trata da avaliação após processada a segunda linha de entrada. Caso após o processamento da segunda linha a ABB esteja vazia deve-se exibir: "ARVORE VAZIA".

A terceira linha contém uma lista de valores encontrados dentro da faixa pesquisada, referente a terceira linha de entrada. Caso não existam valores a exibir deve-se mostrar apenas o texto: "NADA A EXIBIR"(em caracteres maiúsculos).

Caso haja valores exibidos na terceira linha de saída, as linhas seguintes deverão mostrar a altura de cada nó encontrado na ABB, seguida da altura da sub-árvore da esquerda e da direita do nó raiz.

Exemplos de entrada e saída:

- *input01:*

Entrada	Saída
6 4 3 2 1 -1	2, 2, 1
4 5 -1	2, 1, 2
2 6	2, 3, 5, 6
	2, 1, 2
	0, 0, 0
	1, 1, 1
	0, 0, 0

Tabela 1: Exemplos de entrada e saída 01.

A Figura 1 exibe a árvore AVL referente a entrada de dados constante da linha um do primeiro exemplo, onde a altura a partir do nó raiz é $h = 2$, com altura da esquerda $he = 2$ e da direita $hd = 1$.

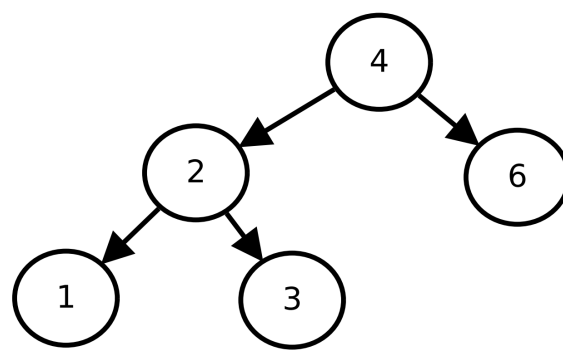


Figura 1: ABB do tipo AVL depois da inserção de todos elementos de entrada.

Na Figura 2, tem-se a árvore binária do tipo AVL, onde o nó de valor 4 (segunda linha de entrada) deverá ser removido e substituído pelo seu sucessor, o nó de valor 6.

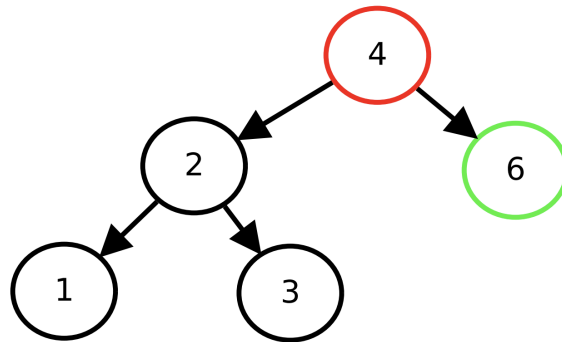


Figura 2: ABB do tipo AVL com destaques para o nó 4 (borda vermelha) a ser removido e para o nó 6 (borda verde) que é o seu do nó 4.

Na Figura 3, tem-se uma árvore desbalanceada após a última remoção (nó quatro), com destaque para o nó de valor 6 ocupando a raiz da árvore.

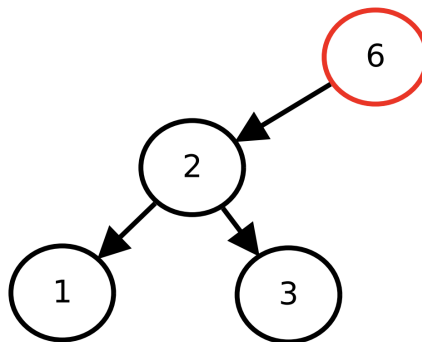


Figura 3: ABB desbalanceada após a remoção do 4.

Na Figura 4, tem-se a árvore já balanceada após rotação a direita.

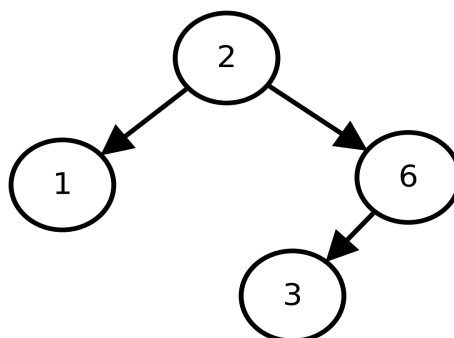


Figura 4: ABB após rotação a direita aplicada na raiz.

Na Figura 5, tem-se a árvore desbalanceada após a inserção do nó 5 (segunda linha de entrada).

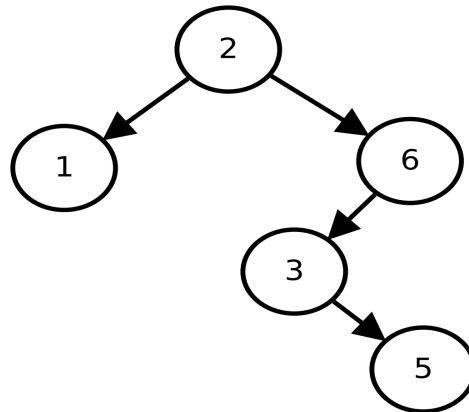


Figura 5: ABB desbalanceada após a inserção do valor 5.

Na Figura 6, tem-se a árvore final balanceada após a última inserção (valor 5, da segunda linha de entrada).

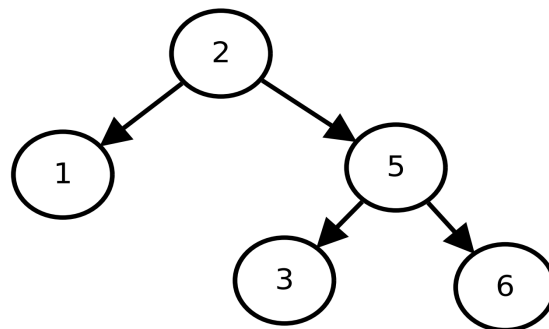


Figura 6: ABB após o balanceamento.

Ao pesquisar pelo nó de valor 6, verifica-se que sua altura é $h = 0$, $he = 0$ e $hd = 0$, pois se trata de um nó folha, portanto sem filhos.

- *input02*

No exemplo 2, a ABB do tipo AVL, formada pelos dados de entrada da Tabela 2, é similar ao *input01*, conforme se observa nas Figuras 6 e 7. A diferença se dá apenas nos valores da faixa de pesquisa.

Entrada	Saída
6 4 3 2 1 -1	2, 2, 1
4 5 -1	2, 1, 2
1 2	1, 2
	0, 0, 0
	2, 1, 2

Tabela 2: Exemplos de entrada e saída 02.

Entrada	Saída
16 14 20 12 11 19 18 15 17 13 -1	3, 3, 3
14 19 15 20 -1	2, 2, 2
18 18	18
	1, 1, 0

Tabela 3: Exemplos de entrada e saída 03

- *input03*:

A Figura 7 exibe a árvore AVL referente aos dados *input 03*, dispostos na Tabela 3. A altura da árvore é $h = 3$, com à esquerda $he = 3$ e à direita $hd = 3$.

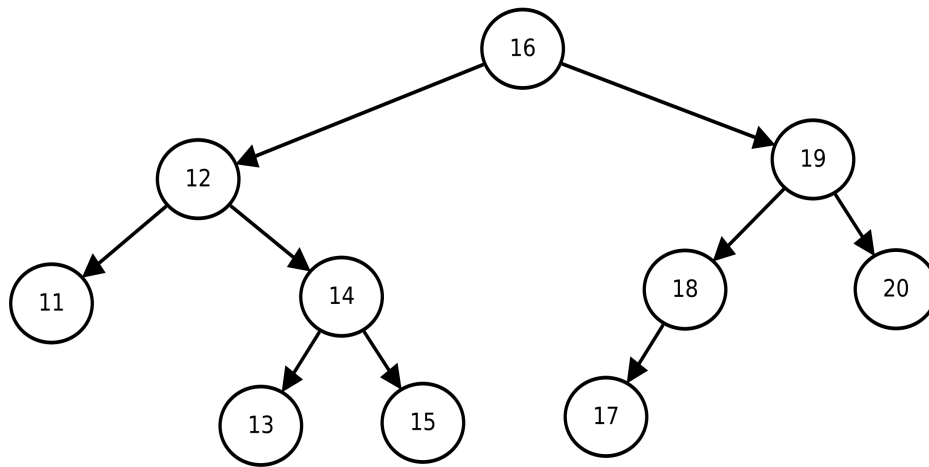


Figura 7: AVL após a inserção dos elementos constantes da primeira linha dos dados de entrada do *input3*.

A Figura 8 exibe a ABB do tipo AVL após as inserções e remoções solicitadas na segunda linha do *input 03*.

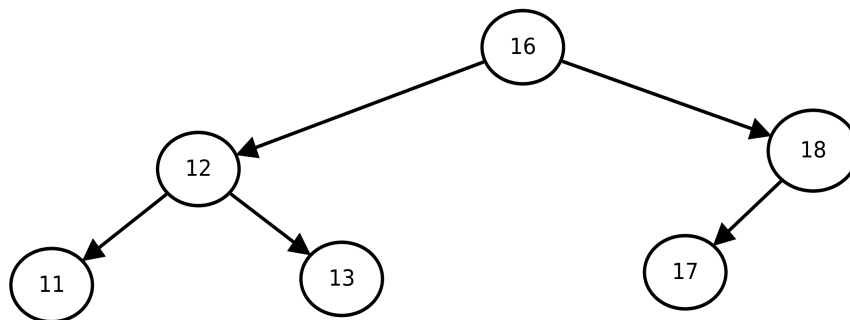


Figura 8: Árvore AVL depois da inserção e remoção dos elementos do *input03*.

A Figura 9 exibe a ABB do tipo AVL com destaque para o nó de valor 18, que será pesquisado, conforme informado na linha 03 do *input 03*. A altura desse nó é $h = 1$, com à esquerda $he = 1$ e à direita $hd = 0$.

- *input04*:

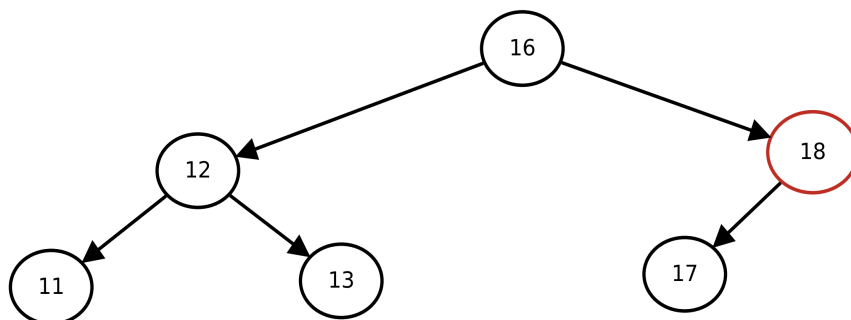


Figura 9: ABB com destaque para a busca do elemento 18.

Entrada	Saída
16 14 20 12 11 19 18 15 17 13 -1	3, 3, 3
14 19 15 20 -1	2, 2, 2
90 100	NADA A EXIBIR

Tabela 4: Exemplos de entrada e saída 04