

# AED2 2022 (1s) - EXERCÍCIO 10 - ÁRVORE RUBRO NEGRA - REMOÇÃO

---

## 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 <http://judge.unifesp.br/aed2S01A2022/>;

## Descrição:

Implemente as operações básicas de um algoritmo de pesquisa em Árvores Vermelho e Preta (AVP). Seu programa **deve** conter os seguintes procedimentos: (1) inicialização, (2) pesquisa, (3) inserção, (4) cálculo da altura de um nó e (5) cálculo da altura-rubro. 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 AVP é a mesma já utilizada nas atividades anterior, sobre ABB e AVL. Isto é, a 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 de  $x$  até um nó folha;
- A **altura-rubro (ou vermelha)** de um nó em uma AVP é a distância entre o nó  $x$  e o seu descendente mais afastado **contando apenas os nós de cor vermelha**, incluindo o próprio nó  $x$ , **se este for da cor vermelha**.

Considere as seguintes condições:

1. A complexidade de cada procedimento implementado deve ser 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 pesquisada 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 contém um único valor inteiro positivo a ser apenas **pesquisado** na árvore. Esse número pode ou não estar na árvore. Caso o valor não seja encontrado, ele **não** deve ser inserido.

## SAÍDA:

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.

Na segunda linha, deve ser impresso o valor da altura para todos os nós pesquisados e encontrados, ou seja, nós que já foram inseridos na árvore, seguido pela altura da esquerda e direita. Cada resultado deverá ser exibido em uma nova linha.

Na última linha de saída deve ser impresso o valor da **altura vermelha** do nó pesquisado (linha 03 dos dados de entrada). Caso esse valor pesquisado não seja encontrado, deve ser exibido "Valor nao encontrado".

### Exemplos de entrada e saída:

- *input01:*

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

Tabela 1: Exemplos de entrada e saída 01

A Figura 1 exibe a árvore Vermelha e Preta referente a entrada de dados constante da linha 01 do primeiro exemplo, onde a altura a partir do nó raiz é  $h = 2$ , com altura da esquerda  $he = 2$  e da direita  $hd = 1$ .

Na Figura 2 tem-se a árvore binária do tipo Vermelha e Preta, onde o nó de valor 2, em destaque com borda verde, deverá retornar a altura, a altura à esquerda e à direita,  $h = 1$ ,  $he = 1$  e  $hd = 1$  e depois deverá ser excluído.

Na Figura 3 tem-se a árvore binária do tipo Vermelha e Preta, onde o nó de valor 2 deverá ser excluído, para isso é necessário a busca do seu sucessor, que no caso

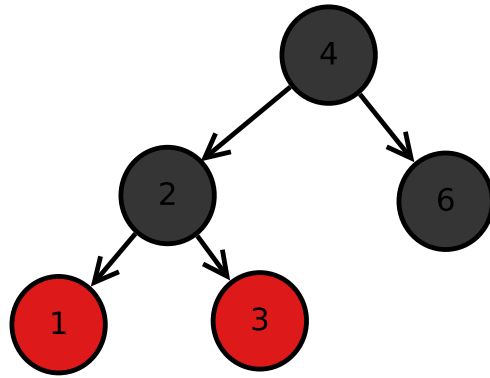


Figura 1: ABB do tipo vermelha e preta depois da inserção de todos elementos de entrada

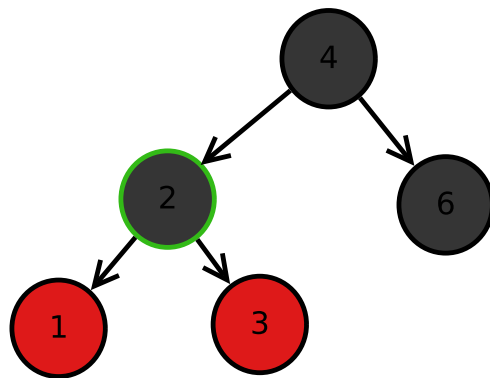


Figura 2: ABB do tipo Vermelha e preta com destaques para o nó 2, com borda verde, a ser pesquisado

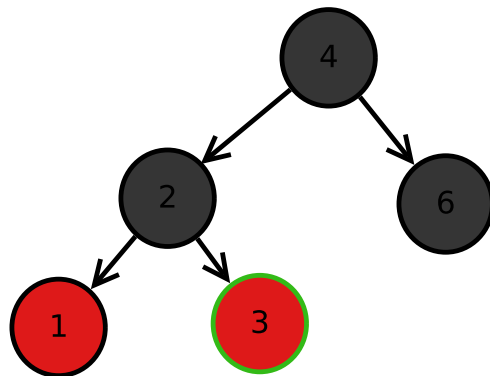


Figura 3: Para a remoção do nó 2 tem-se como substituto o nó sucessor, o nó 3

é o valor em destaque com borda verde, nó 3.

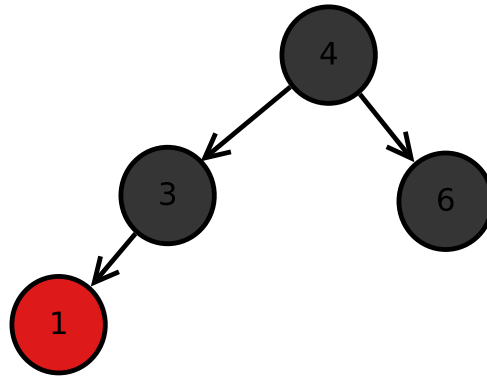


Figura 4: ABB do tipo Vermelha e preta após a exclusão do nó 2

Na Figura 4 tem-se a árvore após a remoção do valor 2.

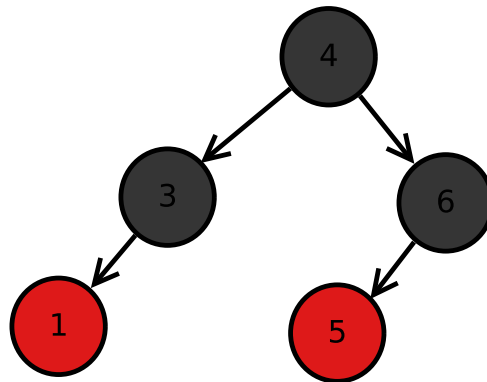


Figura 5: ABB do tipo Vermelha e preta final

Na Figura 5 tem-se a árvore final. Nesta etapa será feita uma busca pelo valor 6 (definido na terceira linha dos dados de entrada), o qual, conforme se pode verificar na mesma Figura 5, tem altura rubro 1.

- *input02*

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

Tabela 2: Exemplos de entrada e saída 02

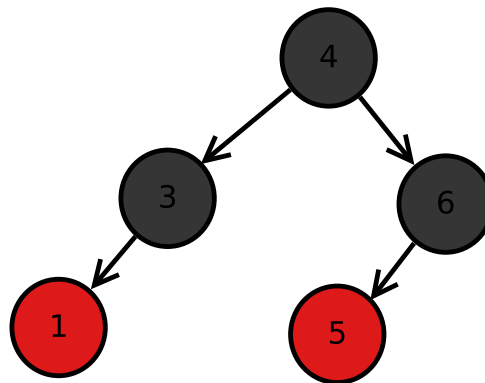


Figura 6: ABB do tipo vermelha e preta final

A Figura 6 exibe a árvore Vermelha e Preta referente a entrada de dados constante da linha 01, igual ao *input01*, porém agora a busca é para o valor 4 que também retornará 1 de altura rubro.

- *input03*

Entrada	Saída
6 4 3 2 1 5 7 8 -1	3, 2, 3
6 4 9 11 10 8 4 -1	2, 1, 2
9	2, 2, 2
	2, 1, 2
	2

Tabela 3: Exemplos de entrada e saída 03

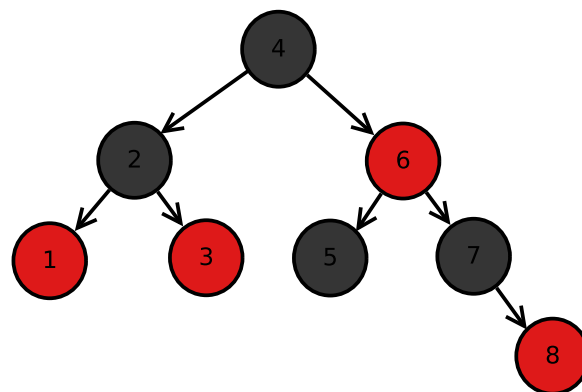


Figura 7: Árvore inicial do input3, após a inserção dos valores da primeira linha

A Figura 7 exibe a árvore Vermelha e Preta referente a entrada de dados constante da linha 01 do primeiro exemplo, onde a altura a partir do nó raiz é  $h = 3$ , com altura da esquerda  $he = 2$  e da direita  $hd = 3$ .

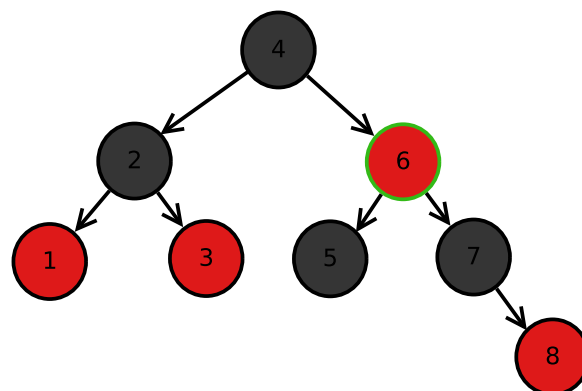


Figura 8: AVP com destaque para elemento de valor 6 buscado na árvore.

A Figura 8 destaca o valor 6, nó com borda verde. Como o valor 6 já está inserido, deve-se exibir a altura do nó 6, ou seja,  $h = 2$ , com altura da esquerda  $he = 1$  e da direita  $hd = 2$  e depois deverá ser excluído.

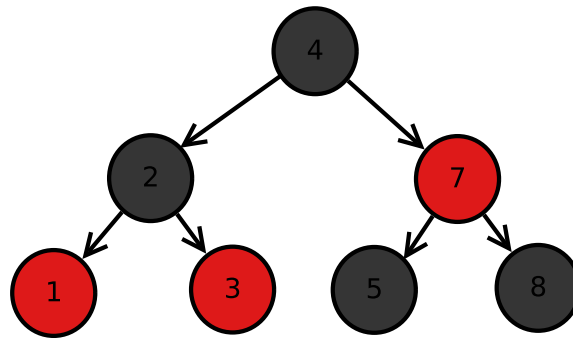


Figura 9: AVP após a remoção do valor 6

A Figura 9 refere-se a árvore AVP após a exclusão do nó de valor 6.

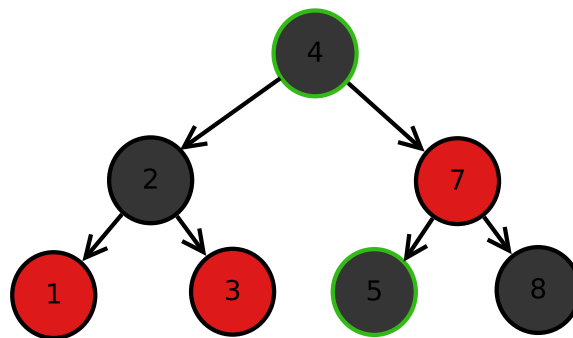


Figura 10: AVP destaca o valor 4 a ser buscado e removido e 5 como o seu sucessor 4

A Figura 10 retrata a busca pelo próximo valor da segunda linha de entrada, referente ao valor 4. Uma vez que este valor também já encontra-se na árvore, deve-se exibir os dados referente a altura do citado nó 4, ou seja,  $h = 2$ ,  $he = 2$  e  $hd = 2$ . Depois o nó deverá ser excluído e substituído pelo seu sucessor, no caso o nó de valor 5, também em destaque na mesma figura.

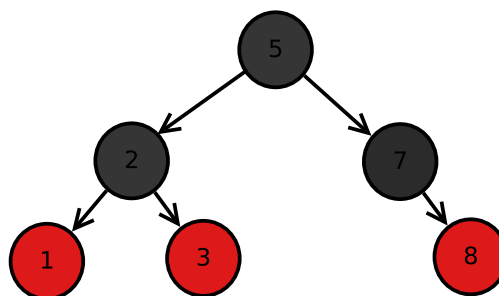


Figura 11: AVP após a remoção do nó 4.

A Figura 11 representa a árvore após a remoção do nó de valor 4.

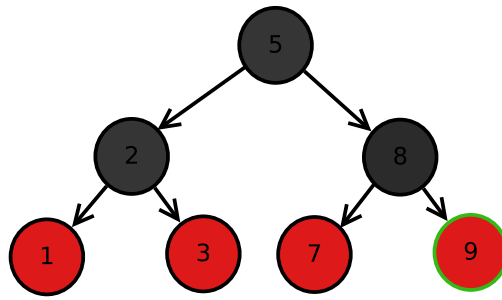


Figura 12: Inserção do valor 9 na AVP.

A Figura 12 destaca o valor 9 inserido na árvore, onde foi necessário transformações para mantê-la balanceada.

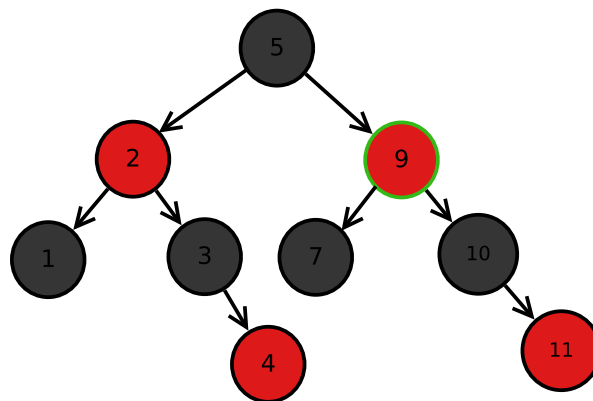


Figura 13: Árvore vermelho e preta final.

A Figura 13 retrata como a AVP ficou após a inserção ou remoção de todos os valores constantes da segunda linha dos dados de entrada. Na próxima etapa será feita uma busca pelo valor 9 (definido na terceira linha dos dados de entrada), o qual, conforme se pode verificar na mesma Figura 13, tem altura rubro 2.



- *input04*

Entrada	Saída
6 4 3 2 1 5 7 8 -1	3, 2, 3
6 4 9 11 10 8 4 -1	2, 1, 2
20	2, 2, 2
	2, 1, 2
	Valor nao encontrado

Tabela 4: Exemplos de entrada e saída 03

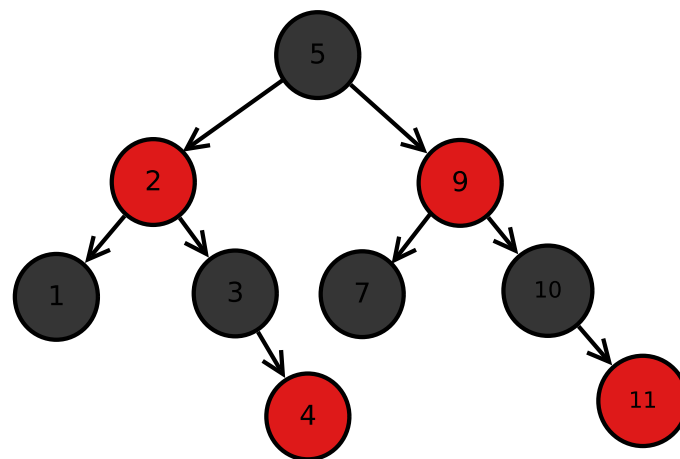


Figura 14: AVP após a inserção e/ou remoção dos elementos constantes da segunda linha dos dados de entrada do *input4*.

A Figura 14 representa a árvore final após a leitura das linhas 01 e 02 dos dados de entrada (igual ao *input03*). Neste caso de teste a busca é para o valor 20. Como este valor não existe na AVP, deve-se retornar a mensagem: "*Valor nao encontrado*".