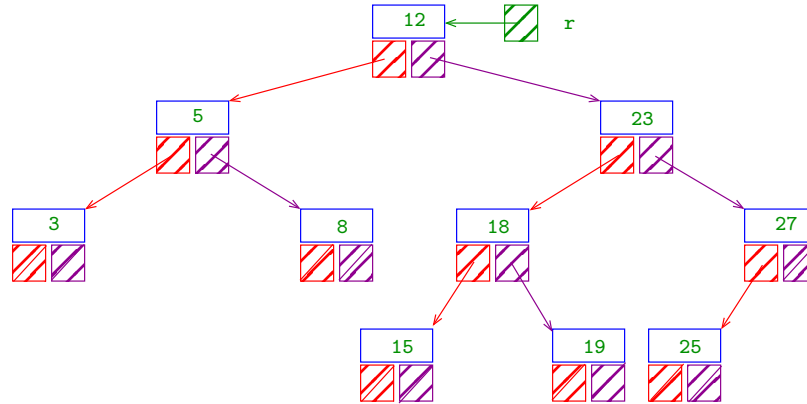


Nome: \_\_\_\_\_

**Questão.** Considere a **árvore binária de busca** ilustrada a seguir.



Liste os conteúdos dos nós em **in-ordem**: Resposta: 3 5 8 12 15 18 19 23 25 27

Qual é a profundidade do nó **r**? Resposta: zero

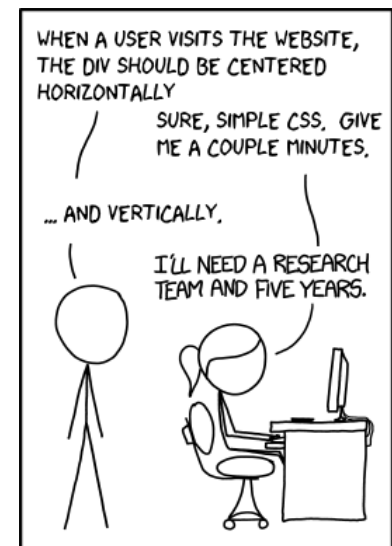
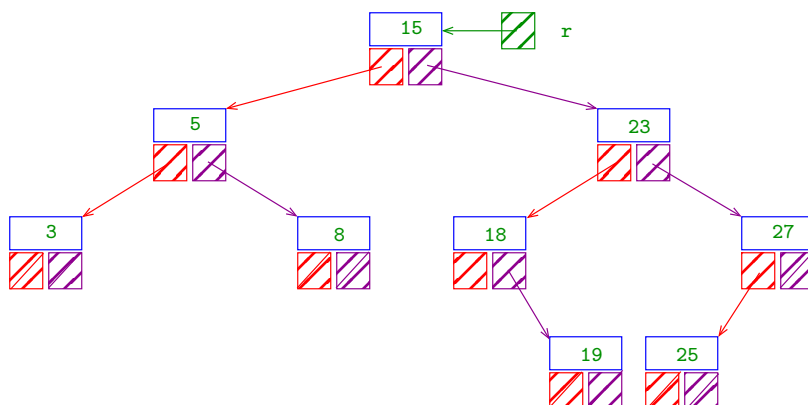
Qual é a **profundidade média** dos nós dessa árvore?

Resposta:  $(0 + 1 + 1 + 2 + 2 + 2 + 2 + 3 + 3 + 3)/10 = 19/10 = 1.9$

Qual é o **número médio de comparações** em uma busca bem-sucedida, supondo que cada chave tem a mesma probabilidade de ser buscada?

Resposta:  $1 + 1.9 = 2.9$

Desenhe a árvore resultante depois da operação **delete(12)**.



IN CS, IT CAN BE HARD TO EXPLAIN THE DIFFERENCE BETWEEN THE EASY

**Observação.** Qual é o número de chaves examinadas/comparadas em uma busca com sucesso (= busca em que a chave está na árvore)?

O **comprimento interno** (= *internal path length*) de uma BT é a soma das profundidades dos seus nós, ou seja, a soma dos comprimentos de todos os caminhos que levam da raiz até um nó. O comprimento interno da BST mostrada anteriormente é 19.

O número de comparações para encontrarmos uma dada chave é 1 mais a profundidade do nó que contém

a chave. Se somarmos todas as profundidades dos nós de uma árvore obtemos o comprimento interno  $C$  da árvore. O número médio de comparações para uma busca com sucesso nessa árvore é portanto  $1 + C/n$ . No caso da árvore acima esse valor é 2,9.

Uma **BST aleatória** é uma BST que se obtém inserindo  $n$  chaves distintas em ordem aleatória numa árvore inicialmente vazia.

Buscas com sucesso numa BST aleatória com  $n$  chaves requerem cerca de  $2 \lg n$  comparações na média.