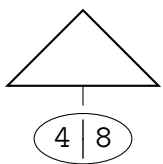


Remoção na árvore 2-3

A remoção na árvore 2-3 é um pouco mais complicada.

E nós vamos apenas examinar como ela é feita — (*a sua implementação vai ficar como exercício*)

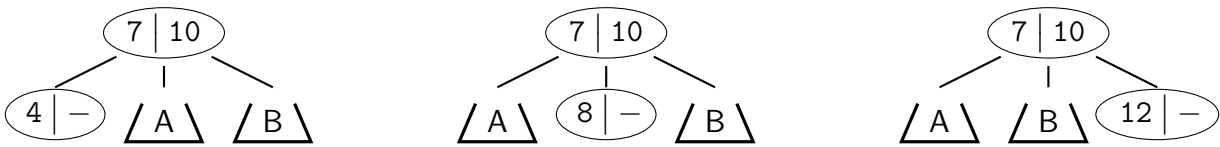
A remoção de um elemento em uma folha do tipo 3 é bem fácil



- a remoção do 8 apenas marca a folha como tipo 2
- e a remoção do 4 passa o 8 para o lugar dele

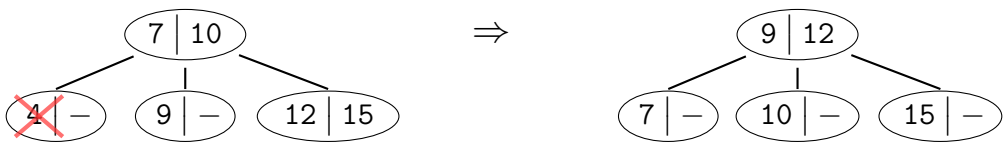
Agora imagine que nós vamos fazer a remoção em uma folha do tipo 2, cujo pai é do tipo 3.

Abaixo nós temos os diversos casos



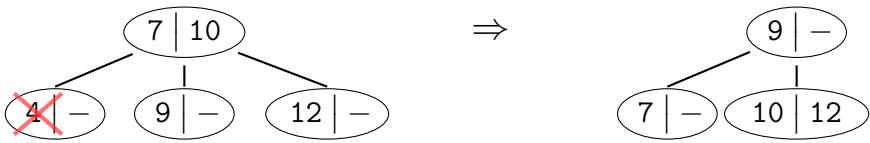
Em todos esses caso, se A ou B é do tipo 3, então a coisa pode ser resolvida trocando elementos de lugar.

Por exemplo,



E no caso em que A e B são do tipo 2, a coisa é resolvida transformando o pai em um elemento 2.

Por exemplo,



Nas duas situações, a operação não se propaga para cima na árvore.

E a remoção é feita em tempo  $O(1)$ .

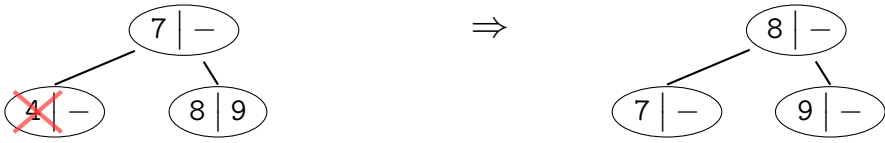
Agora imagine que nós fazemos a remoção em uma folha do tipo 2, cujo pai tem o tipo 2.

Abaixo nós temos os casos

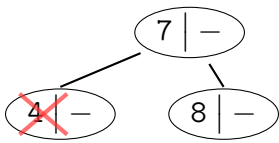


Nos dois casos, se A é do tipo 2, então a coisa pode ser resolvido trocando os elementos de lugar.

Por exemplo,

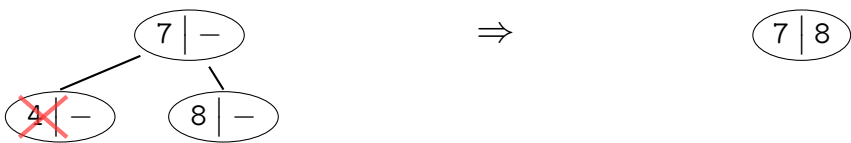


Mas, esse tipo de coisa não pode ser feito quando A também tem o tipo 2



Porque agora, só vão restar dois elementos nessa subárvore.

E a solução é fazer uma desexplosão — (*uma implosão?*)



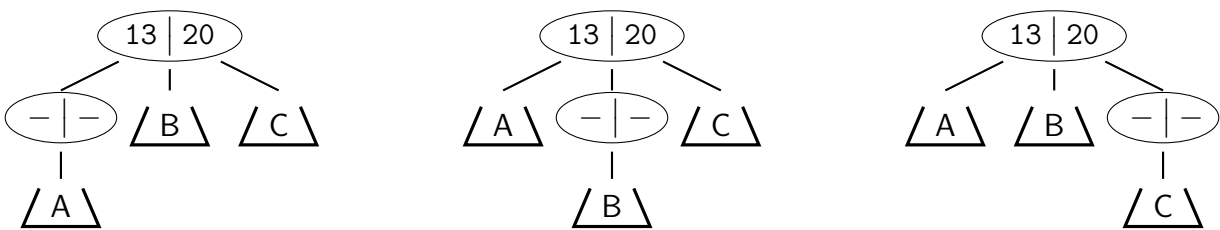
Só que, o elemento que é criado dessa maneira não pode ficar no nível do pai.

Porque senão o nível das folhas ia ficar incompleto.

Essa é a situação em que a remoção é propagada para cima.

Então agora, a gente considera o caso em que um elemento no interior da árvore fica vazio — (*como resultado de uma remoção que aconteceu mais embaixo*)

Abaixo nós temos os diversos casos, onde o pai do elemento tem tipo 3



E aqui a coisa se repete.

Porque a situação pode ser resolvida trocando elementos de lugar.

Ou então fazendo a desexplosão do pai.