

O algoritmo de remoção

Agora que nós já temos a função que remove (e retorna) o menor elemento, nós podemos fazer o caso geral da remoção da seguinte maneira

1. primeiro nós fazemos a descida na árvore para localizar o elemento **k**
2. se esse elemento possui algum filho Nulo, então a remoção é fácil
3. senão, nós removemos o menor elemento da subárvore direita, e o colocamos no lugar de **k**
4. depois, basta fazer a subida pelo caminho realizando os ajustes necessários

A coisa fica assim

```
Algoritmo  Remoção-AVL (k,p)
{
    Se ( p = Nulo )  Retorna (Nulo)

    // 1a ETAPA: descida na árvore

    q <-- p
    Enquanto ( q != Nulo )
    {
        Se ( k = q->val )      break
        Senão Se ( k < q->val )  q <-- q->esq
        Senão                  q <-- q->dir
    }
    Se ( q = Nulo )  Retorna ( p )

    // 2a ETAPA: casos fáceis

    Se ( q->esq e q->dir = Nulo )
    {
        Se ( q->pai = Nulo )      Retorna (Nulo)
        Senão
            Se ( q = q->pai->esq )  q->pai->esq <-- Nulo
            Senão                  q->pai->dir <-- Nulo
            x <-- q->pai
    }
    Senão Se ( q->esq = Nulo )
    {
        Se ( q->pai = Nulo )  {  q->dir->pai <-- Nulo
                             Retorna (q->dir)
                             }
        Senão Se ( q = q->pai->esq )
        {
            q->pai->esq <-- q->dir
            q->dir->pai <-- q->pai
        }
        Senão
        {
            q->pai->dir <-- q->dir
            q->dir->pai <-- q->pai
        }
        x <-- q->pai
    }
    Senão Se ( q->dir = Nulo )
    {
        ( ... análogo ao caso anterior ... )
    }

    // 3a ETAPA: caso geral

    Senão
    {
        q->dir->pai <-- Nulo      // desconectando a subárvore direita

        (m,qd) <-- Remove-menor-AVL ( q->dir )

        q->dir <-- qd             // reconectando a subárvore direita
        Se ( qd != Nulo )
            qd->pai <-- q

        q->val <-- m->val         // substituindo k pelo menor

        x <-- q
    }

    // 4a ETAPA: subida pelo caminho e ajustes

    ( ... semelhante à 2a etapa da inserção ... )
}
```