

Remover Árvore B

```
remover(arvore, chave)
| <pg,i> := buscaB(arvore, chave)
| se pg == nulo então
| | retorne falso
| se não então
| | pg2 := pg
| | se pg2.folha == verdadeiro então
| | | enquanto pg2[i+1] != nulo
| | | | pg2[i] := pg2[i+1]
| | | | i++
| | | pg2.qtd--
| | | pg2[pg2.qtd-1] := nulo
| se não então
| | pg2:=pg2.ptr[i]
| | enquanto pg2.ptr[pg2.qtd] != nulo faça
| | | pg2:=pg2.ptr[pg2.qtd]
| | | pg[i] := pg2[pg2.qtd-1]
| | | pg2[pg2.qtd-1] := nulo
| | | pg2.qtd--
| se pg2.pai == nulo
| | retorne verdadeiro
| se pg2.irmaEsq != nulo e pg2.qtd + pg2.irmaEsq.qtd < 2d então
| | concatena(pg2, pg2.irmaEsq)
| se não se pg2.irmaEsq != nulo e pg2.qtd + pg2.irmaEsq.qtd >= 2d então
| | redistribui(pg2, pg2.irmaEsq)
| se não se pg2.irmaDir != nulo e pg2.qtd + pg2.irmaDir.qtd < 2d então
| | concatena(pg2, pg2.irmaDir)
| se não se pg2.irmaDir != nulo e pg2.qtd + pg2.irmaDir.qtd >= 2d então
| | redistribui(pg2, pg2.irmaDir)
```

```
concatena(pg,irma)
| se pg.qtd<d e pg.pai != nulo então
| | se pg[pg.qtd-1] > irma[0] então
| | | irma<=>pg
| | | pg[pg.qtd] := pg.chavePai
| | | pg.qtd++
| | para i de 0 até irma.qtd-1
| | | pg[pg.qtd] := irma[i]
| | | pg.prt[pg.qtd] := irma.ptr[i]
| | | pg.qtd++
| | pg.prt[pg.qtd]:=irma.ptr[irma.qtd]
| | irma := nulo
| | para i indicePai até pg.pai.qtd-1
| | | pg.pai[i] := pg.pai[i+1]
| | | pg.pai.ptr[i] := pg.pai.ptr[i+1]
| | pg.pai.prt[pai.qtd]:= nulo
| | pg.pai[pg.pai.qtd-1] := nulo
```

```

| | pg.pai.qtd--
| | pg := pg.pai
| | se pg.irmaEsq != nulo e pg.qtd + pg2.irmaEsq.qtd < 2d então
| | | concatena(pg, pg.irmaEsq)
| | se não se pg.irmaEsq != nulo e pg.qtd + pg.irmaEsq.qtd >= 2d então
| | | redistribui(pg, pg.irmaEsq)
| | se não se pg.irmaDir != nulo e pg.qtd + pg.irmaDir.qtd < 2d então
| | | concatena(pg, pg.irmaDir)
| | se não se pg.irmaDir != nulo e pg.qtd + pg.irmaDir.qtd >= 2d então
| | | redistribui(pg, pg.irmaDir)

```

```

redistribui(pg, irma)
| se pg.qtd<d então
| | se pg[pg.qtd-1] < irma[0] então
| | | pagAux[4d]:=pg[]+pg.chavePai+irma[] //recebe nessa ordem
| | | pagAux.qtd := pg.qtd+irma.qtd+1
| | | pg.chavePai := pagAux[(pagAux.qtd)/2]
| | | | para i de 1 até (pagAux.qtd-1)/2
| | | | | pg.ptr[i-1] := pagAux.ptr[i-1]
| | | | | pg[i-1] := pagAux[i-1]
| | | | | irma[i-1] := pagAux[pagAux.qtd+i]
| | | | | irma.ptr[i-1] := pagAux.ptr[pagAux.qtd+i]
| | se não
| | | pagAux[4d]:=irma[]+pg.chavePai+pg[] // recebe nessa ordem
| | | pagAux.qtd := pg.qtd+irma.qtd+1
| | | pg.cahvePai := pagAux[(pagAux.qtd-1)/2]
| | | | para i de 1 até (pagAux.qtd-1)/2
| | | | | irma.ptr[i-1] := pagAux.ptr[i-1]
| | | | | irma[i-1] := pagAux[i-1]
| | | | | pg[i-1] := pagAux[pagAux.qtd+i]
| | | | | pg.ptr[i-1] :=pagAux.ptr[pagAux.qtd+i]
| | pg.qtd:=pagAux.qtd/2
| | irma.qtd:=pagAux.qtd/2

```