

「アルゴリズムとデータ構造」資料 10

AVL 木

奈良女子大学理学部情報科学科 鴨浩靖

2011 年 12 月 12 日 初版

2013 年 1 月 7 日 第二版

2013 年 1 月 29 日 第二版改訂版

2020 年 12 月 21 日 第二版三訂版

単純な順序つき二分木の弱点と対策

弱点 左右のどちらかに偏ると、探索・挿入・削除ともに遅くなる。

対策 挿入・削除の際に木を再構成して、極端に偏らないようにするとよい。

この対策を加えた順序つき二分木を総称して平衡木と呼ぶ。再構成のタイミングと方法にはいろいろあるので、平衡木もいろいろある。

- ▶ AVL 木
- ▶ 2-3 木
- ▶ 赤黒木 (2-3-4 木)

...

AVL 木

どの節についても、

$$-1 \leq (\text{左部分木の高さ}) - (\text{右部分木の高さ}) \leq 1$$

が成り立っている順序付き二分木。

実装の際は、各々の節に他のラベルに加えて

$$(\text{左部分木の高さ}) - (\text{右部分木の高さ})$$

の値 (-1, 0, 1 のいずれか) もラベル付けしておく と便利。

AVL 木の探索

単に二分木として探索すればよい。

時間計算量

最悪 $O(\log n)$

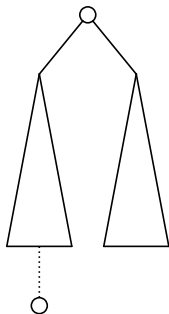
平均 $O(\log n)$

常に、ほぼバランスした状態を保つから。

AVL 木への挿入

ある節の左部分木に挿入が行われて、高さが増えたとき、もとの木の (左部分木の長さ) - (右部分木の長さ) の値に応じて場合分け。

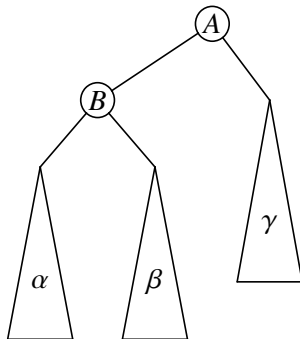
- ▶ (左部分木の長さ) - (右部分木の長さ) = -1 のとき
木の再構成の必要なし。
- ▶ (左部分木の長さ) - (右部分木の長さ) = 0 のとき
この節については、とりあえず、木の再構成の必要はない。根でなければ、親の節について調整が必要。
- ▶ (左部分木の長さ) - (右部分木の長さ) = 1 のとき
再構成が必要。



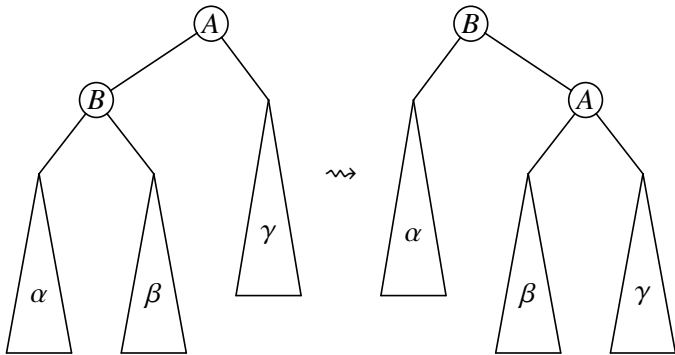
AVL 木への挿入 (つづき)

ある節の左部分木に挿入が行われて、高さが増え、もとの木で
(左部分木の高さ) - (右部分木の高さ) = 1 が成り立っていたとき、
部分木 α が伸びたか部分木 β が伸びたかで、さらに場合分け。

- ▶ 部分木 α が伸びた場合、
一重回転を行う。
- ▶ 部分木 β が伸びた場合、
二重回転を行う。



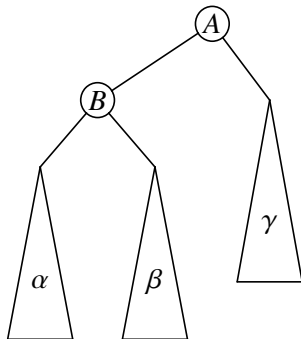
一重回轉



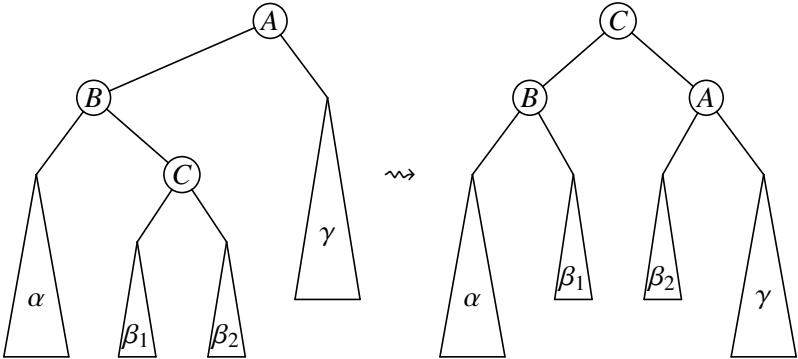
AVL 木への挿入 (つづき) [再掲]

ある節の左部分木に挿入が行われて、高さが増え、もとの木で
(左部分木の高さ) - (右部分木の高さ) = 1 が成り立っていたとき、
部分木 α が伸びたか部分木 β が伸びたかで、さらに場合分け。

- ▶ 部分木 α が伸びた場合、
一重回転を行う。
- ▶ 部分木 β が伸びた場合、
二重回転を行う。



二重回転



計算量

AVL 木への挿入の時間計算量

最悪 $O(\log n)$

平均 $O(\log n)$

平均・最悪とも、二分木としての挿入に $O(\log n)$ の時間がかかり、再構成に $O(\log n)$ の時間がかかるから。

AVL 木からの削除

ある節の右部分木から削除が行われて、高さが減ったとき、もとの木の (左部分木の長さ) - (右部分木の長さ) の値に応じて場合分け。

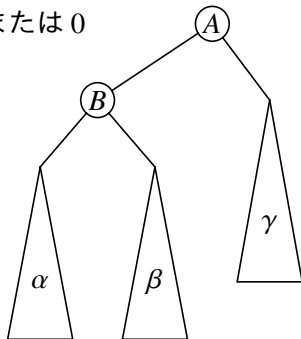
- ▶ (左部分木の長さ) - (右部分木の長さ) = -1 のとき
木の再構成の必要なし。
- ▶ (左部分木の長さ) - (右部分木の長さ) = 0 のとき
この節については、とりあえず、木の再構成の必要はない。根でなければ、親の節について調整が必要。
- ▶ (左部分木の長さ) - (右部分木の長さ) = 1 のとき
再構成が必要。



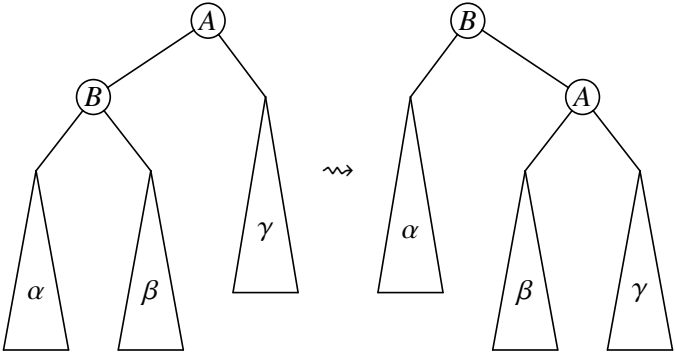
AVL 木からの削除 (つづき)

ある節の右部分木から削除が行われて、高さが減り、もとの木で (左部分木の長さ) - (右部分木の長さ) = 1 が成り立っていたとき、(部分木 α の高さ) - (部分木 β の高さ) の値で、さらに場合分け。

- ▶ $(\alpha \text{ の高さ}) - (\beta \text{ の高さ}) = 1$ または 0 の場合、一重回転を行う。
- ▶ $(\alpha \text{ の高さ}) - (\beta \text{ の高さ}) = -1$ の場合、二重回転を行う。



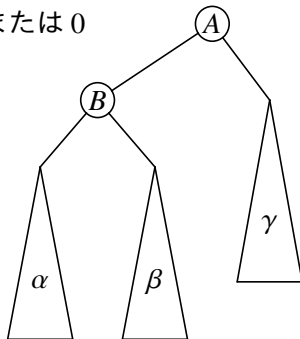
一重回轉 [再掲]



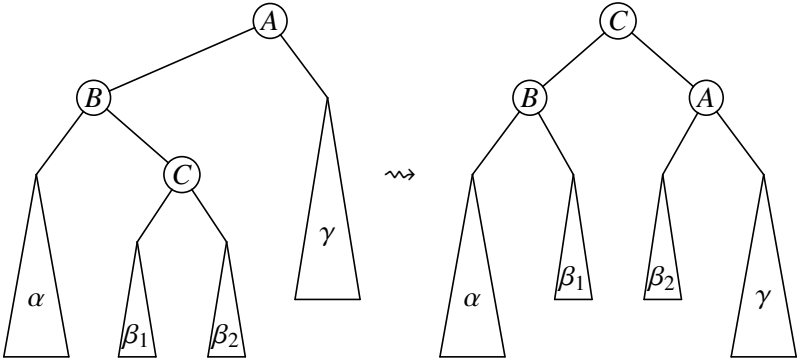
AVL 木からの削除 (つづき) [再掲]

ある節の右部分木から削除が行われて、高さが減り、もとの木で (左部分木の長さ) - (右部分木の長さ) = 1 が成り立っていたとき、(部分木 α の高さ) - (部分木 β の高さ) の値で、さらに場合分け。

- ▶ $(\alpha \text{ の高さ}) - (\beta \text{ の高さ}) = 1$ または 0 の場合、一重回転を行う。
- ▶ $(\alpha \text{ の高さ}) - (\beta \text{ の高さ}) = -1$ の場合、二重回転を行う。



二重回転 [再掲]



計算量

AVL 木からの削除の時間計算量

最悪 $O(\log n)$

平均 $O(\log n)$

平均・最悪とも、二分木としての削除に $O(\log n)$ の時間がかかり、再構成に $O(\log n)$ の時間がかかるから。

まとめ

- ▶ 平衡木
 - ▶ AVL 木