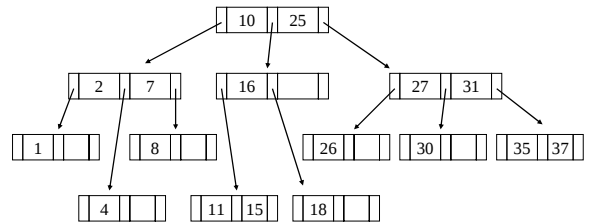


索引データ構造

- データ数: n
- 索引: 値順に要素を並べた配列
 - データの検索: $O(\log n)$ 二分探索の利用
 - データの追加: $O(n)$ 挿入位置以降のデータをずらす
 - データの削除: $O(n)$ 挿入位置以降のデータをずらす
- B木: Balanced Tree
 - データの検索: $O(\log n)$
 - データの追加: $O(\log n)$
 - データの削除: $O(\log n)$

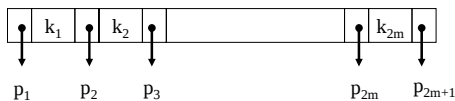
1

2次のB木の例



2

m次のB木の節点



k_i : データ領域

p_j : 他の節点を指すポインタ

3

B木の条件

- 平衡条件: 根節点からすべての葉節点までの距離が等しい
- 根以外の節点について
 - データ数 n は $m \leq n \leq 2m$
 - ポインタ数は $n+1$
- 根節点について
 - データ数 n' は $1 \leq n' \leq 2m$
 - ポインタ数は $n'+1$
- ポインタ p_j が指す部分木の持つデータ値 k_l は以下を満足
 - $j = 1$ のとき: $k < k_1$
 - $j = 2, \dots, n$ のとき: $k_{j-1} < k < k_j$
 - $j = n+1$ のとき: $k_n < k$

4

演習問題

- N 個のデータを保持している m 次のB木の高さ h は $O(\log_m N)$ であることを示せ。
 - 参考: $f(N)$ が $O(\log N)$ である 全ての $N > N_0$ について、 $f(N) < c \log N$ となるような定数 c, N_0 が存在する。
 - ヒント: 高さが最大になるのは、各節点のポインタ数が最小になる場合

5

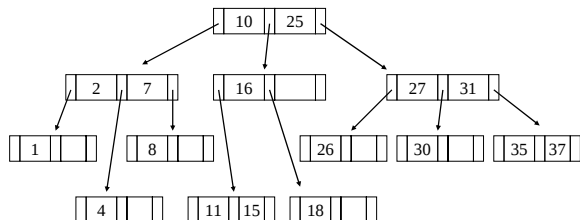
データ検索アルゴリズム

目的: B木内で与えられた値 x を持つ節点を求める。

- 根節点内に x が存在すれば、根節点を返して終了する。(節点内の検索は二分探索を利用)
- 値 x が存在しなければ、 $k_{j-1} < x < k_j$ を満たす j を二分探索で検索する。
- ポインタ p_j が指す部分木に対して、データ検索アルゴリズムを再帰的に適用する。
- 葉節点において x が存在しなければ、検索失敗として終了する。

6

B木におけるデータ検索



7

データ挿入アルゴリズム

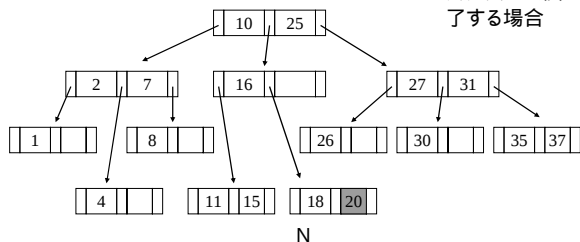
目的: B木上に値xを挿入する。

1. B木内で値xを含むべき節点Nを検索する。(データ検索アルゴリズムを利用)
2. Nが $2m-1$ 個以下の値しか含まないならば、xをNに挿入して終了する。
3. Nが $2m$ 個の値を含むならば、Nに含まれる値にxを加えた $2m+1$ 個の値を、m個の値を含む2つの節点 N_1 、 N_2 と1つの値yに分割する。ただし、 N_1 の値はすべてyより小さく、 N_2 の値はすべてyより大きくなければならない。
4. yをNの親節点N'に挿入する。このとき、データ挿入アルゴリズムを再帰的に適用する。(N'の分割が再帰的に行われることがある)

8

20の挿入

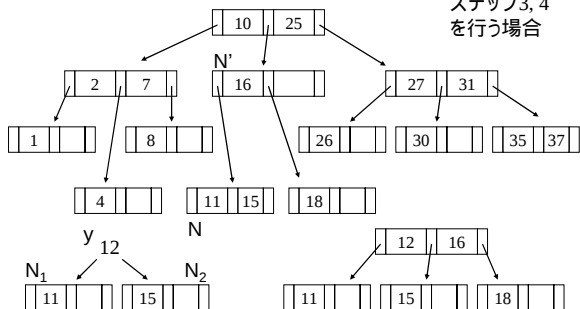
ステップ2で終了する場合



9

12の挿入

ステップ3, 4を行う場合



10

データ削除アルゴリズム

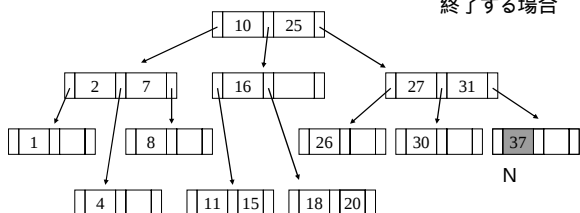
目的: B木上から値xを削除する。

1. B木内で値xを含む節点Nを検索する。(データ検索アルゴリズムを利用)
2. Nが $m+1$ 個以上の値を含むならば、xをNから削除して終了する。
3. Nがm個の値しか含まないならば、Nからxを削除する。
4. Nのすぐ上の兄節点をN'とする。(兄節点が存在しない場合には弟節点をN'とする)
5. Nの親節点の中で、NとN'を分けている値をnとする。
6. N'が $m+1$ 個以上の値を含む場合は以下の処理を行う。
 - 6.1 N'の最小値(N'が弟節点の場合には最大値)n'をN'から削除する。
 - 6.2 値nを節点Nに挿入後、値n'をnと置き換える。
7. N'(兄弟とも)がm個以下の値しか含まない場合は以下の処理を行う。
 - 7.1 値nを親節点から削除して、節点Nに挿入する。(削除アルゴリズムの再帰的実行)
 - 7.2 NとN'を合併する。

11

35の削除

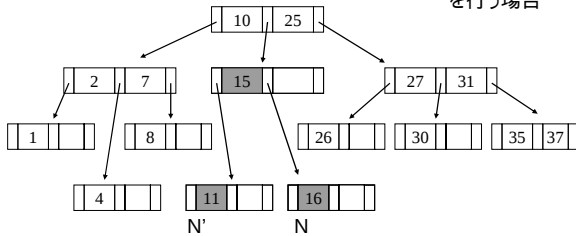
ステップ2で終了する場合



12

18の削除

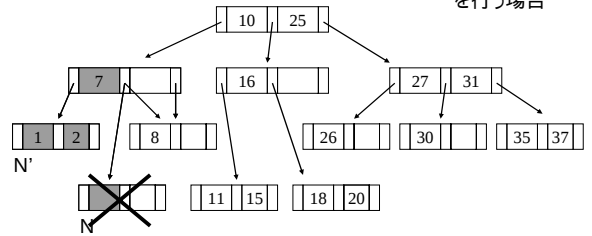
ステップ6
を行う場合



13

4の削除

ステップ7
を行う場合



14

演習問題

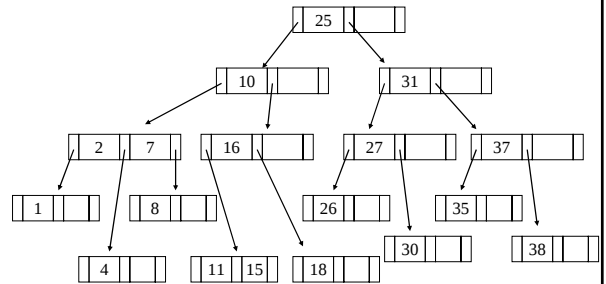
スライド(2ページ)で示したB木に対して以下の操作を行う場合、アルゴリズムの各ステップの動作を説明せよ。

1. 38を挿入する。
2. 18を削除し、その直後に11を削除する。

参考:最終的なB木の状態を16、17ページにそれぞれ示す。

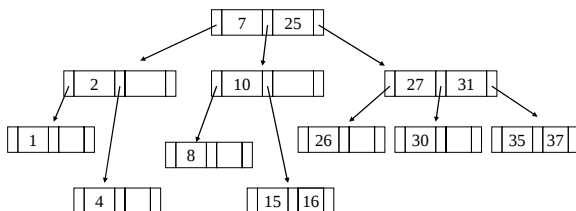
15

38を挿入した後のB木



16

18と11を削除した後のB木



17