アルゴリズム及び演習 第 5 回演習解答

小野 孝男[*]

2007 年 5 月 21 日

1. 最初の 4 個のデータを `insert` すると図 1(a)〜(d) のように完全 2 分木は変化する. 最終的には図 2 のような完全 2 分木が得られる.
2. n 個の頂点からなる完全 2 分木の高さを h とおく. このとき, 深さが 0 から $h-1$ までは全て頂点で埋まっており, 深さ h のみ頂点が欠けているところがあることになる. 一般に深さが k

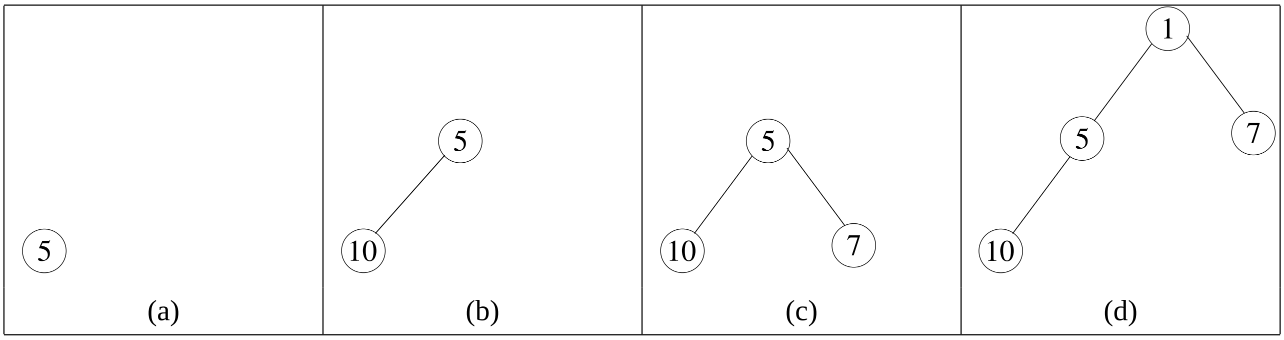

図 1　最初の 4 個のデータを順にヒープに `insert`

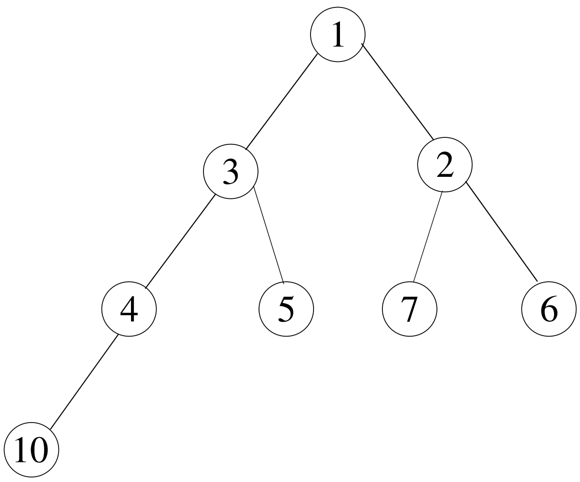

図 2　全データをヒープに `insert` した結果

[*] ono@is.nagoya-u.ac.jp

であるような頂点は高々 2^k 個であるから, 高さ h の完全 2 分木の頂点数 n は深さ h の頂点を m 個とすると

$$n = \sum_{k=0}^{h-1} 2^k + m = 2^h + m - 1$$

で表される．ここで $1 \le m \le 2^h$ より $2^h \le n \le 2^{h+1} - 1$ である．各辺の対数をとって $h \le \log n < h + 1$. 書き換えると $\log n - 1 < h \le \log n$ だから $h = \lfloor \log n \rfloor$.

3. n 頂点の完全 2 分木において葉の数を n_L, 葉でない頂点の数を n_I とすると $n = n_L + n_I$ かつ $n_I \le n_L \le n_I + 1$ が成り立つ (これは, 辺の本数を 2 通りで数える). つまり, $n_I = \lfloor n/2 \rfloor$ である. このことから, 一般に「高さ k 以上の頂点数は高々 $n/2^k$」ということがわかる (k に関する帰納法で証明可能). つまり, 木 B の高さを H, 高さが h であるような頂点の数を $N(h)$ とおくと

$$\sum_{h=k}^{H} N(h) \le \frac{n}{2^k}$$

である. 従って (図 3 からわかるように)

$$\begin{aligned}
\sum_{v \in V} h(v) &= \sum_{h=0}^{H} h \cdot N(h) = \sum_{h=1}^{H} \sum_{k=h}^{H} N(h) \\
&\le \sum_{h=1}^{H} \frac{n}{2^h} \le \sum_{h=1}^{\infty} \frac{n}{2^h} \\
&= n.
\end{aligned}$$

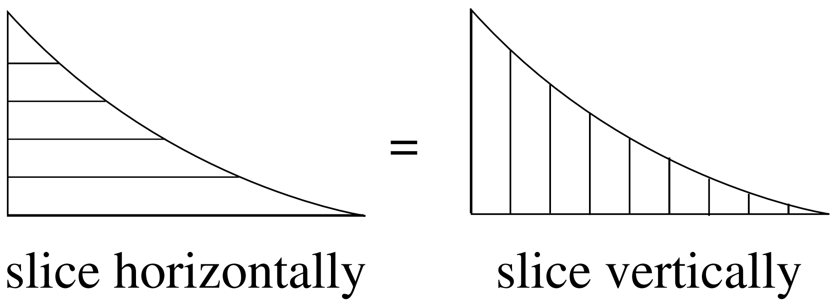

図 3　2 通りの積算

4. 非常に大雑把に:

```
while 入力にデータが残っている do
次のデータを読み込む
if 数値 then スタックにプッシュ
if 演算子 then スタックから 2 個のデータをポップし, 計算したあと得られた結果をプッシュ
done
```