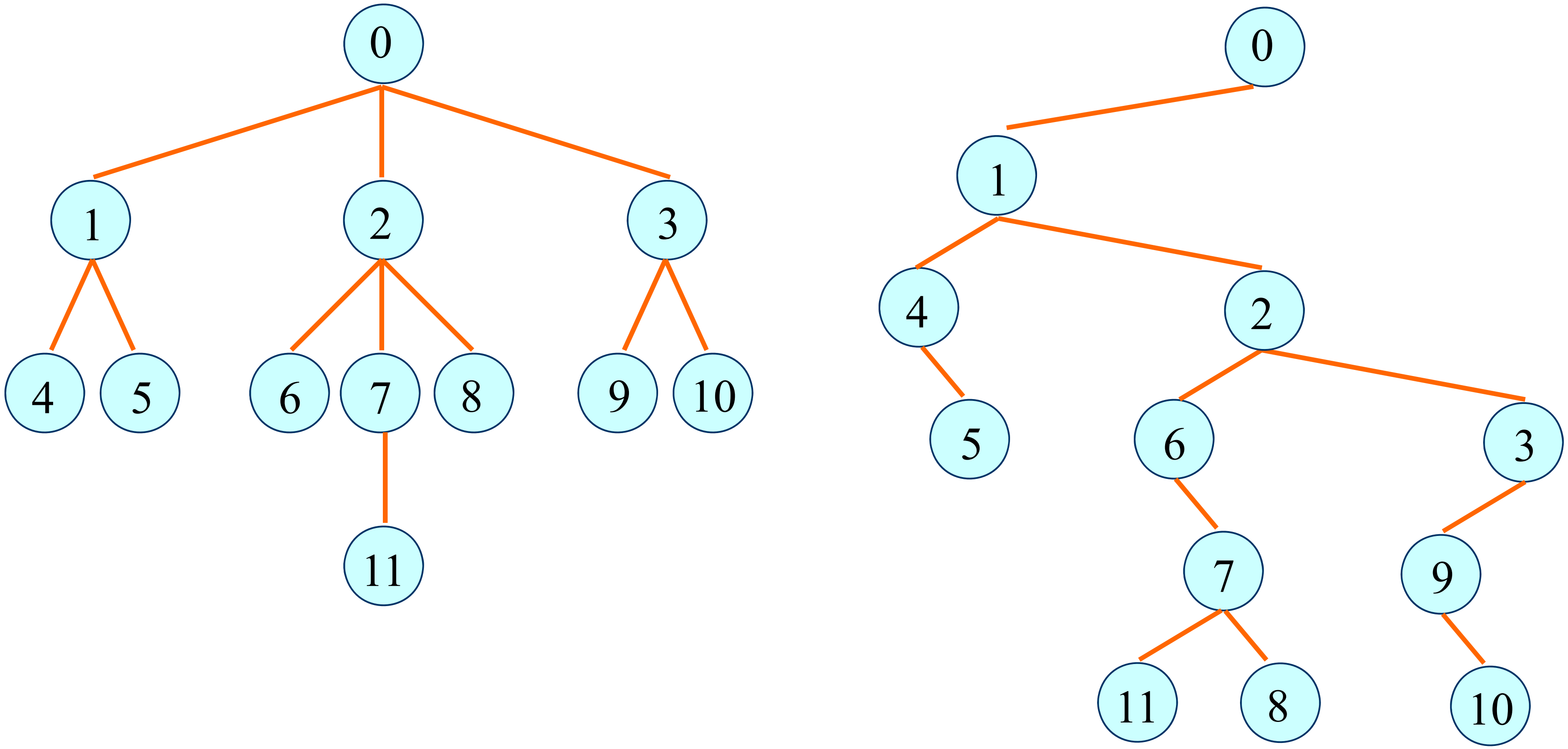
任意の木は，２分木で表現可能：

１段深い部分木は，左（右）部分木に結合．

同じ深さの節点は，右（左）部分木に結合，

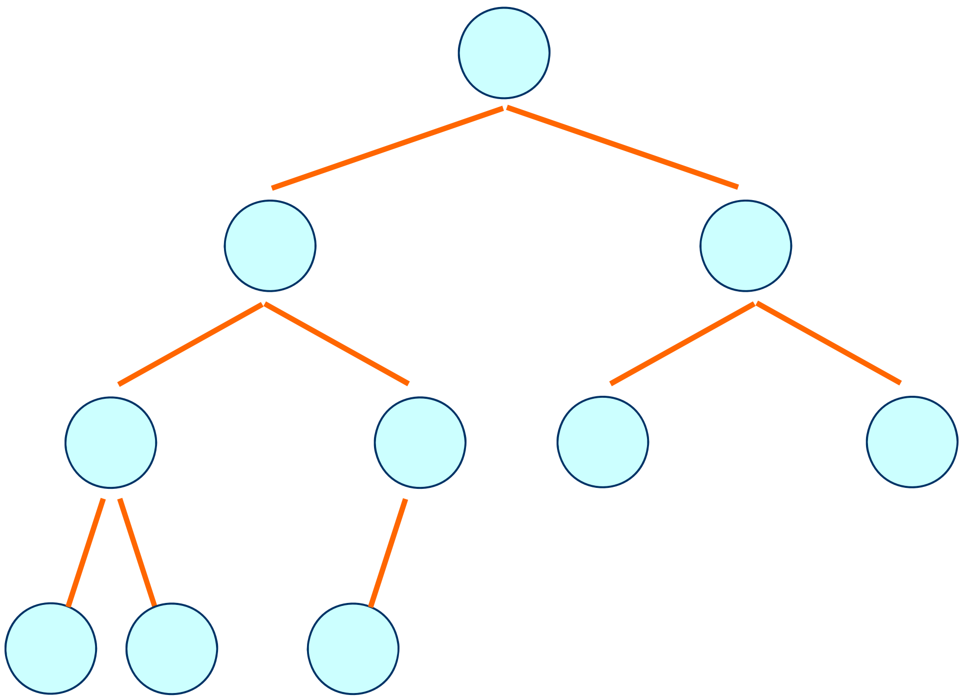


完全２分木：

深さ d の２分木において，

深さ j （0 ≦ j ＜ d）には 2^j 個の節点．

深さ d の葉は，左端に配置．



完全２分木の表現

節点の区別：

１番から ｎ 番の，ｎ 個の節点が存在する．
根から葉へ，左から右へ，番号をつける．
ｉ番目の節点を，番号 ｉ で区別し，その要素を
node[i] と表す．

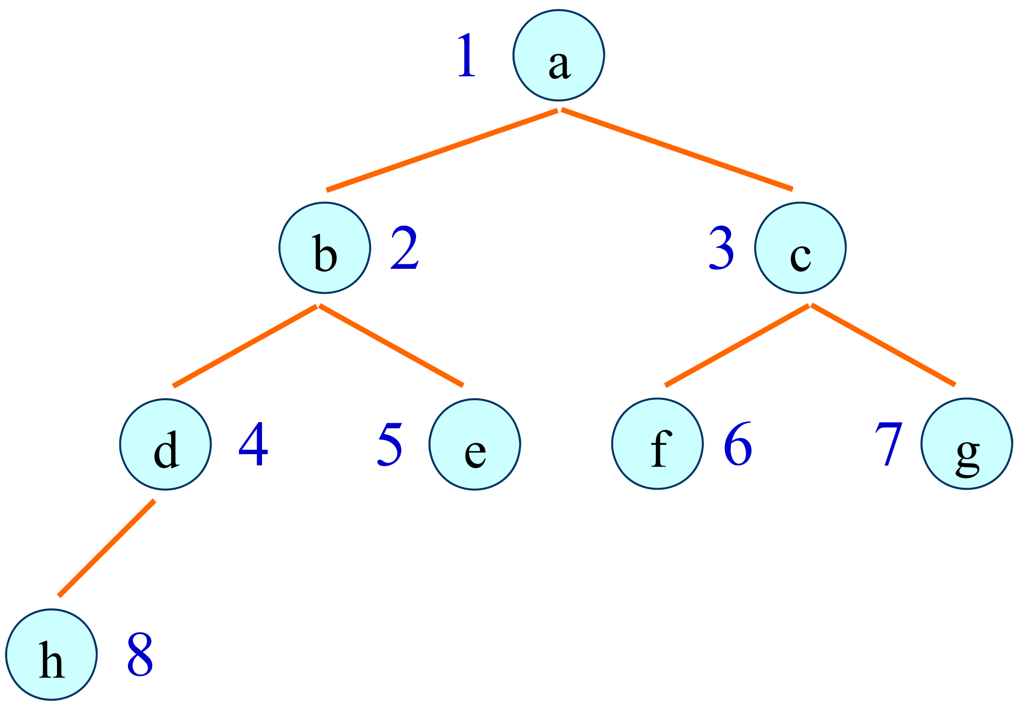
根を 1 で表現．
節点 ｉ に対して，２つの子は，

2*i，2*i + 1

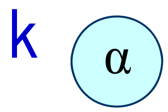
で表現．
親と，その２つの子を，番号で特定できる．

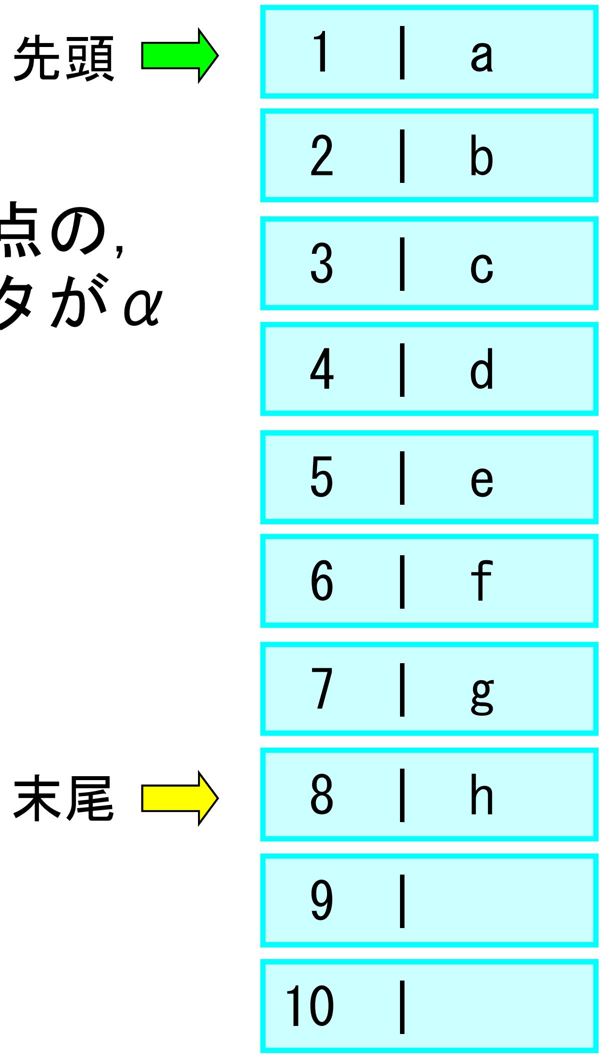
完全２分木：

- １ から n の節点.
- 最深部の葉が左に寄せられている.
- k ＝ n/2 （切捨て）に対し，
  葉は，k ＋ １ から n.



k 番目の節点の，
要素のデータがα





木における半順序関係：

前提：　各節点が，データとして要素を持つ.
　　　　要素は，比較対象としての値を持つ.

半順序関係：子が持つ要素の値は，親が持つ
　　　　　　要素の値以上となっている.

親の値 ≦ 子の値

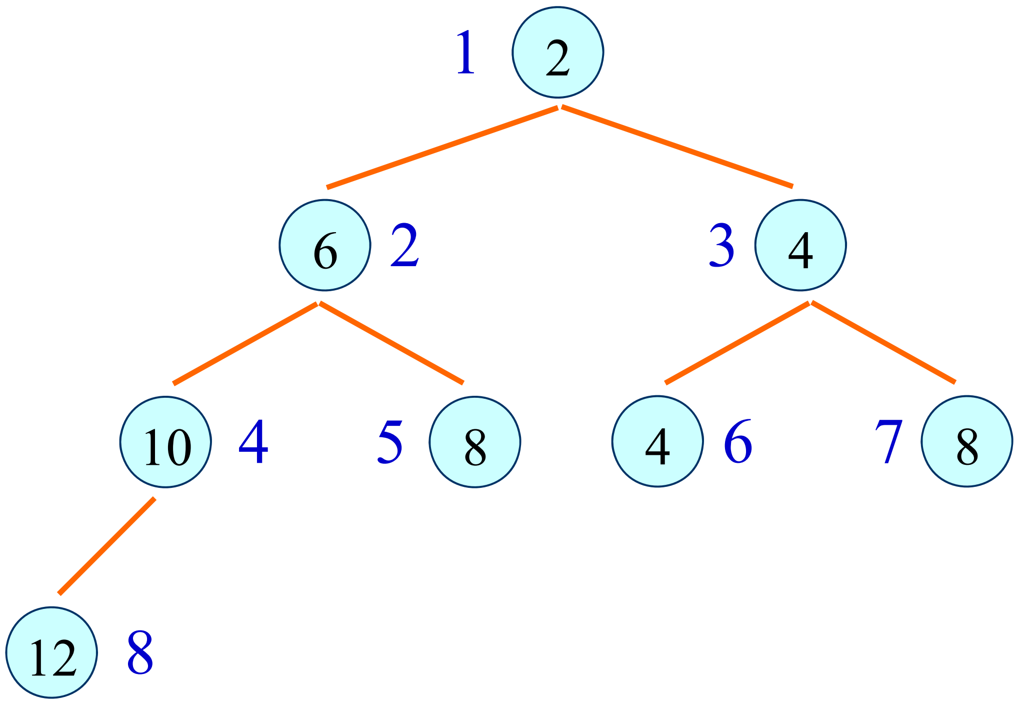
値：　数値，アルファベット，あいうえお，・・・

比較方法：　大小の順，頻度順，画数順，・・・

半順序，完全，２分木：ヒープ（２分ヒープ）

k 番目の節点の，
要素の値がα

k α



ヒープ

先頭	1	2
	2	6
	3	4
	4	１０
	5	8
	6	4
	7	8
末尾	8	１２
	9	
	10	

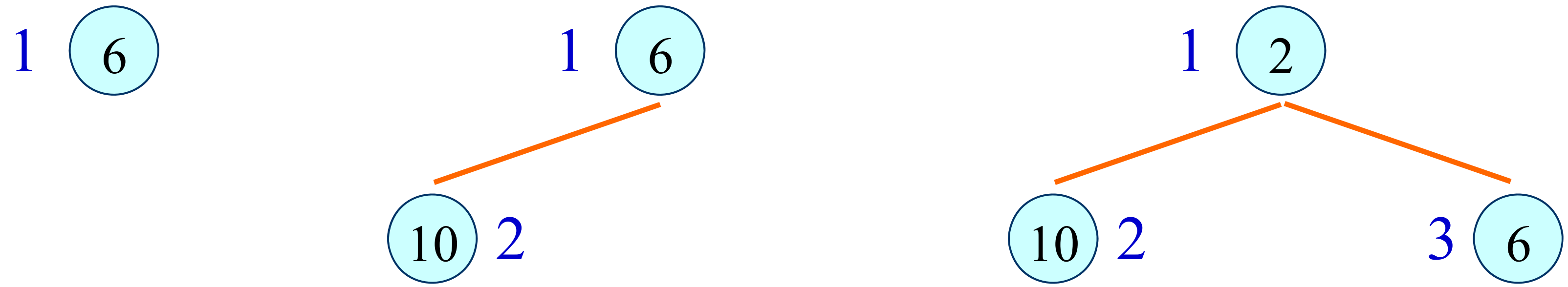
前提：　節点数 n に対しヒープ node の領域は，十分に多く，ヒープ先頭は node[1]．初期では，節点数 n はゼロであり，ヒープ末尾は node[0] の位置にある．

半順序，完全，２分木への要素ｐの追加

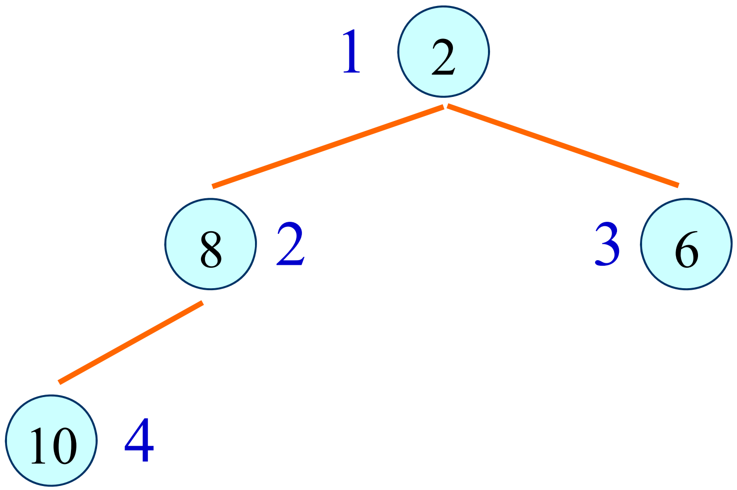
(1) ヒープの末尾の直後に，追加要素ｐを入れる．
　　ヒープの末尾を追加要素ｐの位置に移動する．
(2) 節点数 n を１増やす．
(3) 繰り返す（より上に対し局所的な半順序の保証）．
　　(3-1) 追加要素ｐが根にあれば，(3) の終り．
　　(3-2) 追加要素ｐと親節点要素で値を比較し，
　　　　親が大きいならば，
　　　　　追加要素ｐと親節点要素を入れ替える．
　　　　さもなければ，(3) の終り．

ヒープへの要素の追加：　木の成長過程

6を追加　　10を追加　　2を追加



8を追加



木は，完全２分木の性質を
満足しつつ成長する.