●性質 3.4

完全2分木の高さは，その完全2分木の節点の数を n とすると，$\log_2(n+1) = O(\log n)$ である．

これらの木に関する性質をまとめると，“完全2分木の高さは，節点の数や葉の数に対して対数関数となっており，完全2分木の節点の数や葉の数は，高さの指数関数となっている”ということである．これはさまざまなアルゴリズムの計算量を考えていくうえで重要な性質なので，おぼえておくとよい．

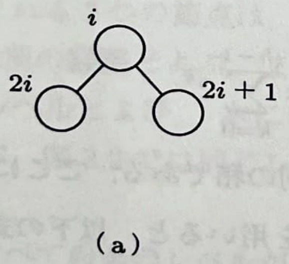
木の実現　木をアルゴリズム中で用いるための実現方法はさまざまなものがあり，それらの多くをここで説明するのは難しい．そこで，ここでは，配列を用いて完全2分木を表現する方法のみを説明する．これは，後述のヒープとよばれるデータ構造で木を使用する場合に用いられる方法である．

まず，木の節点を区別するために，各節点に番号を付ける．すべての節点に異なる番号を付けるために，この番号付けは以下のような方針で行う．

① 木の根を表す節点の番号を1とする．

② 番号 i をもつ節点の2つの子について，左側の子の番号を $2i$，右側の子の番号を $2i+1$ とする(図 3.4 (a))．

この方針により根から葉に向かって各レベルごとに番号付けを行うと，図 3.4 (b) のように各節点に対して固有の連続した番号を付けることができる．



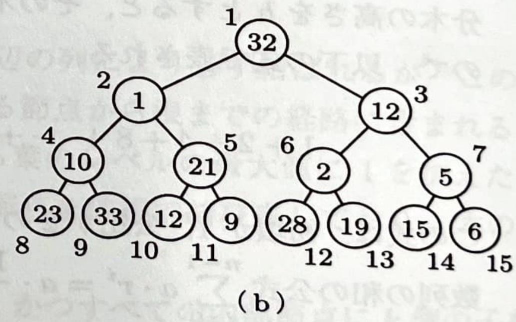


図 3.4　番号付けされた完全2分木とその番号付けの方針

つぎに，木を表す配列 `T` を準備する．配列 `T` のサイズは，木に含まれる節点の数が n である場合は，少なくとも $n+1$ でなければならない．この配列を用いて木を実現するには，番号が i の節点のデータを `T[i]` に格納するとよい．図 3.5 に図 3.4 (b)の木のデータを格納する配列を示す．なお，`T[0]` は使用しないので図から省略している．

	[1]	[2]	[3]	[4]	[5]	[6]	[7]	[8]	[9]	[10]	[11]	[12]	[13]	[14]	[15]
T	32	1	12	10	21	2	5	23	33	12	9	28	19	15	6

図 3.5　完全2分木を表す配列

この配列を用いると，番号が i の節点のデータは `T[i]` に格納されており，番号 i の節点の2つの子のデータは，それぞれ `T[2i]` と `T[2i+1]` に格納されている．また，番