

```

1  /**
2   * Construct the binary heap given an array of items.
3   */
4  public BinaryHeap( AnyType [ ] items )
5  {
6      currentSize = items.length;
7      array = (AnyType[]) new Comparable[ ( currentSize + 2 ) * 11 / 10 ];
8
9      int i = 1;
10     for( AnyType item : items )
11         array[ i++ ] = item;
12     buildHeap( );
13 }
14
15 /**
16  * Establish heap order property from an arbitrary
17  * arrangement of items. Runs in linear time.
18  */
19 private void buildHeap( )
20 {
21     for( int i = currentSize / 2; i > 0; i-- )
22         percolateDown( i );
23 }

```

图 6-14 buildHeap 的架构

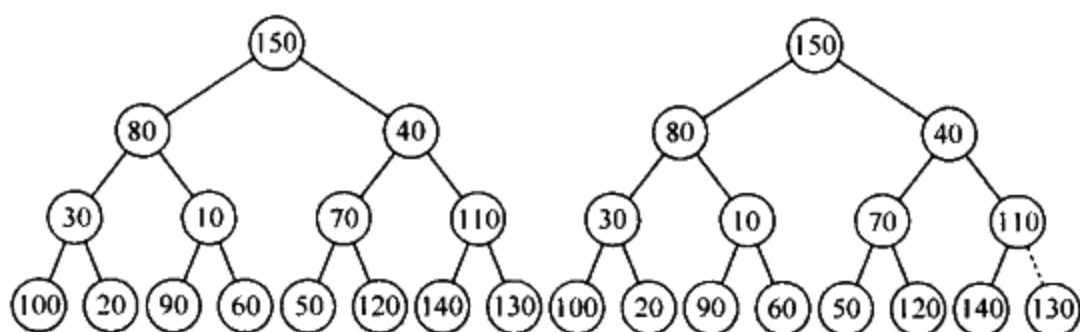


图 6-15 左: 初始堆; 右: 在 percolateDown(7)后

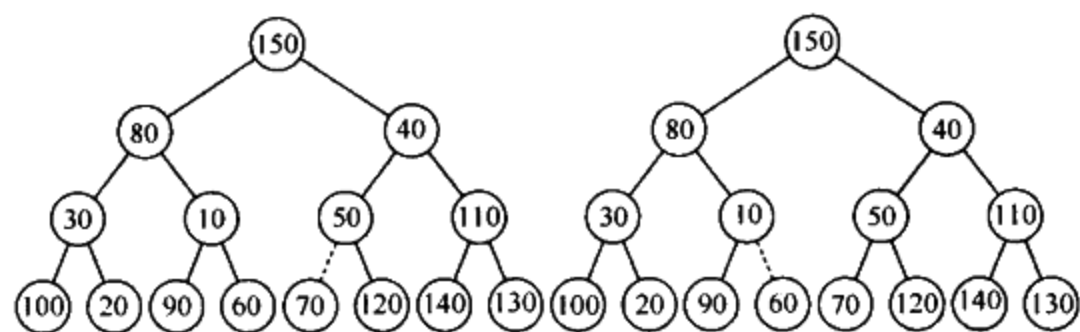


图 6-16 左: 在 percolateDown(6)后; 右: 在 percolateDown(5)后

定理 6.1 包含 $2^{h+1}-1$ 个节点、高为 h 的理想二叉树(perfect binary tree)的节点的高度的和为 $2^{h+1}-1-(h+1)$ 。

证明:

容易看出, 该树由高度 h 上的 1 个节点、高度 $h-1$ 上的 2 个节点、高度 $h-2$ 上的 2^2 个节点以及一般地在高度 $h-i$ 上的 2^i 个节点等组成。则所有节点的高度的和为