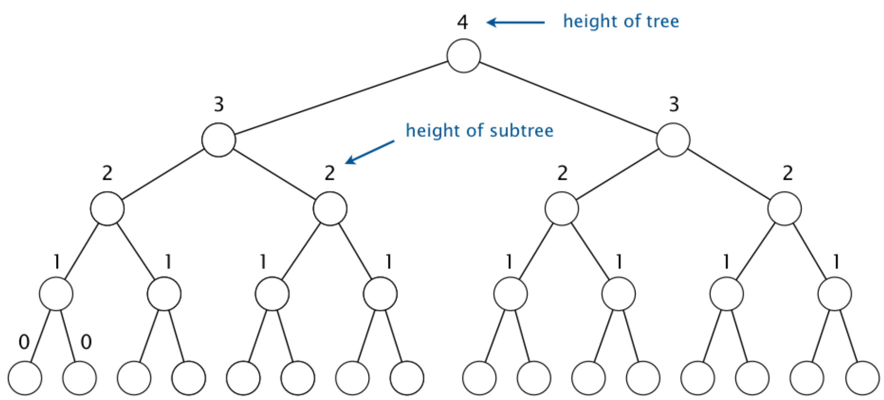
Prove that sink-based heap construction uses at most 2*n* compares and at most *n* exchanges.

(from http://algs4.cs.princeton.edu/24pq/ on 20160927)

Solution. It suffices to prove that sink-based heap construction uses fewer than *n* exchanges because the number of compares is at most twice the number of exchanges. For simplicity, assume that the binary heap is *perfect* (i.e., a binary tree in which every level is completed filled) and has height *h*.



We define the *height* of a node in a tree to be the height of the subtree rooted at that node. A key at height *k* can be exchanged with at most *k* keys beneath it when it is sunk down. Since there are 2^{h-k} nodes at height *k*, the total number of exchanges is at most:

$$\begin{aligned} h + 2(h-1) + 4(h-2) + 8(h-3) + \ldots + 2^h(0) &= 2^{h+1} - h - 2 \\ &= N - (h-1) \\ &\leq N \end{aligned}$$

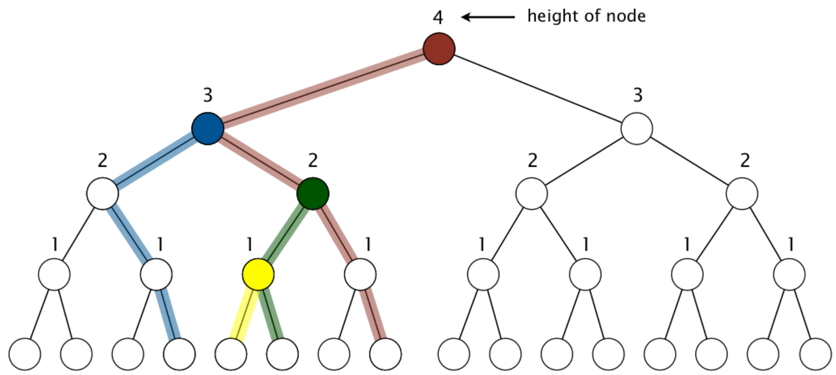
The first equality is for a nonstandard sum, but it is straightforward to verify that the formula holds via mathematical induction. The second equality holds because a perfect binary tree of height *h* has $2^{h+1} - 1$ nodes.

Proving that the result holds when the binary tree is not perfect requires a bit more care. You can do so using the fact that the number of nodes at height *k* in a binary heap on *n* nodes is at most ceil(n / 2^{k+1}).

Alternate solution. Again, for simplicity, assume that the binary heap is *perfect* (i.e., a binary tree in which every level is completed filled). We define the *height* of a node in a tree to be the height of the subtree rooted at that node.

- First, observe that a binary heap on *n* nodes has $n - 1$ links (because each link is the parent of one node and every node has a parent link except the root).
- Sinking a node of height *k* requires at most *k* exchanges.
- We will charge *k* links to each node at height *k*, but not necessarily the links on the path taken when sinking the node. Instead, we charge the node the *k* links along the path from

the node that goes left-right-right-right-.... For example, in the diagram below, the root node is charged the 4 red links; the blue node is charged the 3 blue links; and so forth.



- Note that no link is charged to more than one node. (In fact, there are two links not charged to any node: the right link from the root and the parent link from the bottom rightmost node.)
- Thus, the total number of exchanges is at most *n*. Since there are at most 2 compares per exchange, the number of compares is at most 2*n*.