p.181 5.16

Merge the two leftist heaps in Figure 5.58.

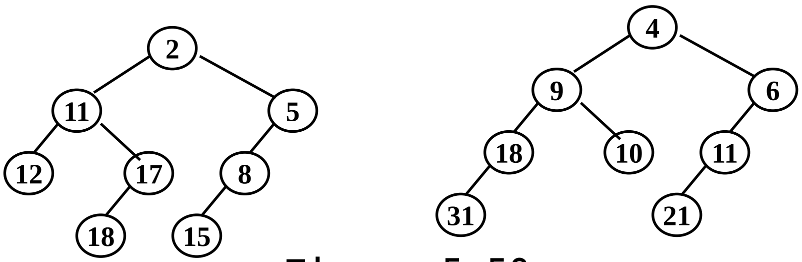
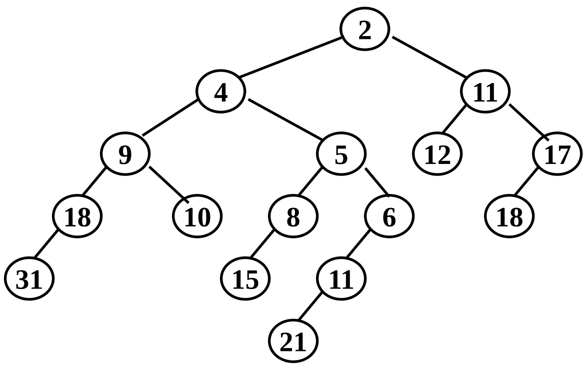


Figure 5.58

Solution:



p.183 5.22

We can perform BuildHeap in linear time for leftist heaps by considering each element as a one-node leftist heap, placing all these heaps on a queue, and performing the following step: Until only one heap is on the queue, dequeue two heaps, merge them, and enqueue the result.

a. Prove that this algorithm is O(N) in the worst case.

b. Why might this algorithm be preferable to the algorithm described in the text?

a. Proof:

In the 1st run, $\lceil N/2 \rceil$ leftist heaps are formed, each contains at most 2 nodes;
In the 2nd run, $\lceil N/2^2 \rceil$ leftist heaps are formed, each contains at most 2^2 nodes;
Following this pattern, it is clear that in the *k*-th run, $\lceil N/2^k \rceil$ leftist heaps are formed, each

contains 2^k nodes.

The worst case is when N = 2^K for some integer K so that equal-sized heaps are merged in each run. Therefore we have:

$$T(N) = O\left(\frac{N}{2}\log 2^0 + \frac{N}{2^2}\log 2^1 + \frac{N}{2^3}\log 2^2 + \ldots + \frac{N}{2^K}\log 2^{K-1}\right)$$

Let us denote the sum by

$$S = 2^{K-1} + 2^{K-2}\times 2 + 2^{K-3}\times 3 + \ldots + 2^{K-(K-1)}\times(K-1) \qquad (1)$$

Then: $$2S = 2^K + 2^{K-1}\times 2 + 2^{K-2}\times 3 + \ldots + 2^{K-(K-2)}\times(K-1) \qquad (2)$$

Subtract equation (1) from (2) we obtain:

$$\begin{aligned} S &= 2^K + 2^{K-1} + 2^{K-2} + \ldots + 2^2 - 2\times(K-1) \\ &= 2\times 2^K - 2K - 2 \\ &= O(N) \end{aligned}$$

Thus $T(N) = O\left(\frac{N}{2^{K+1}}\times S\right) = O(N)$.

b. Answer:

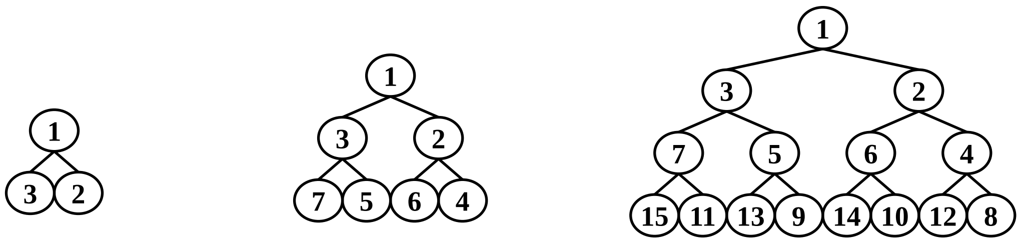
First, it is simpler because it avoids the bottom-up pointer manipulations.

Secondly, since each merge tends to produce unbalanced heap, the resulting heap is more likely to be unbalanced to the left, while the binary heap is always perfectly balanced (which is bad to a leftist heap).

p.183 5.24

```
Show the result of inserting keys 1 to 15 in order into a skew
heap.
```

Solution:



What will happen with 2^k – 1 for any k > 4?