Heaps as collections

Goldsmiths Computing

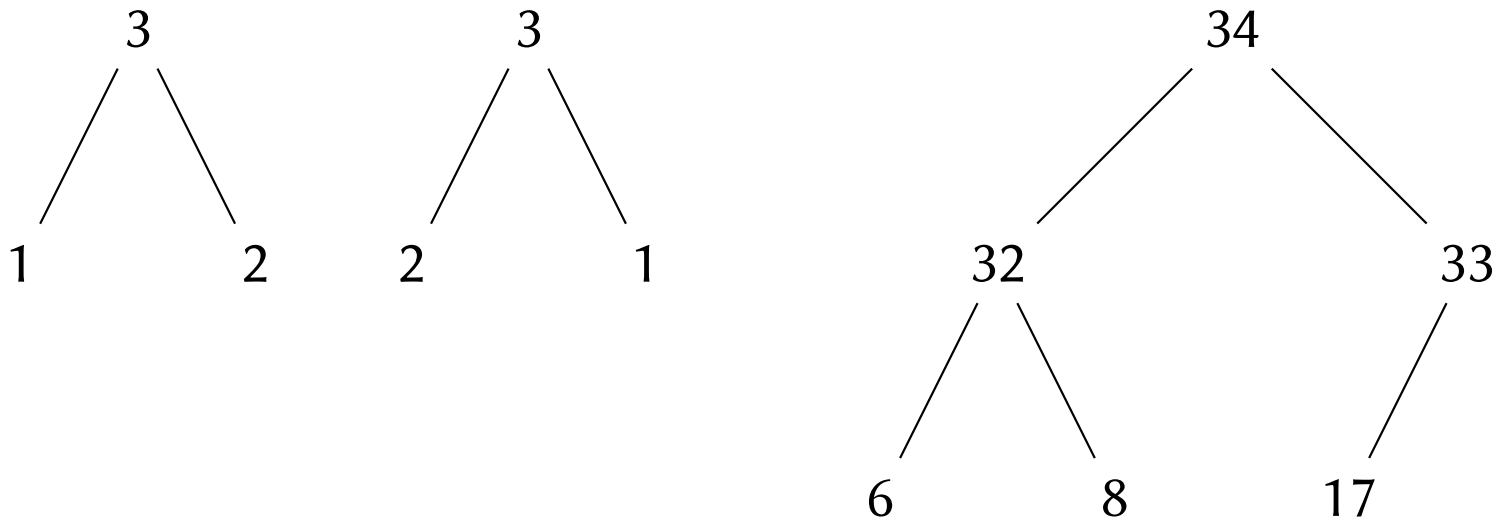
Motivation

An unordered collection for ordered keys which supports efficient construction and efficient extraction of the maximum key.

Definition

A heap is a tree data structure which both satisfies the heap contents property, and also satisfies the (nearly-)complete shape property.

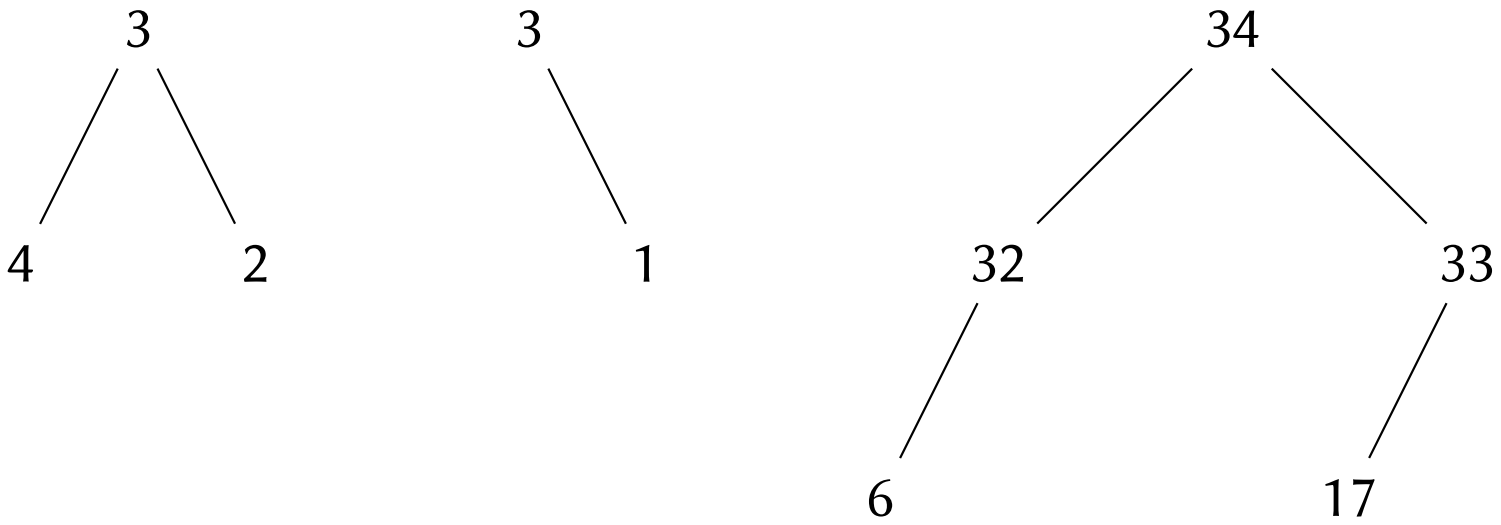
Example heaps



Definition

A heap is a tree data structure which both satisfies the heap contents property, and also satisfies the (nearly-)complete shape property.

Example non-heaps



Collection operations

find

Require: heap :: max Heap

```
function FIND(heap,object)
    if NULL?(heap) then
        return false
    end if
    if heap.key = object then
        return true
    else if heap.key < object then
        return false
    else
        return FIND(heap.left,object) ∨ FIND(heap.right,object)
    end if
end function
```

Collection operations

max

Require: heap :: non-empty max Heap

```
function MAX(heap)
    return heap.key
end function
```

Complexity analysis

find

must in principle go down both branches (*e.g.* to find object smaller than minimum element)

$$\Longrightarrow \Theta(N)$$

max

read key of root node

$$\Longrightarrow \Theta(1)$$

Work

1. Reading
   - CLRS, section 6.1
2. Questions from CLRS:

   Exercises 6.1-1, 6.1-2, 6.1-3, 6.1-4