Tampereen yliopisto
Tampere University

Keko
(heap)

COMP.CS.300 Tietorakenteet ja algoritmit 1

Matti Rintala (matti.rintala@tuni.fi)

Tampereen yliopisto
Tampere University

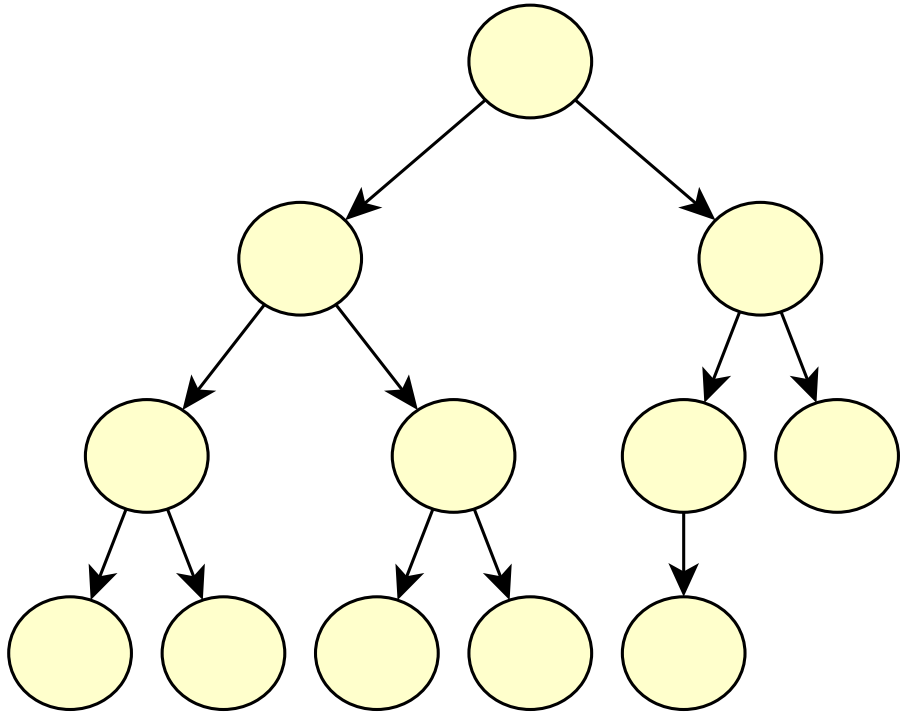
Keon operaatiot

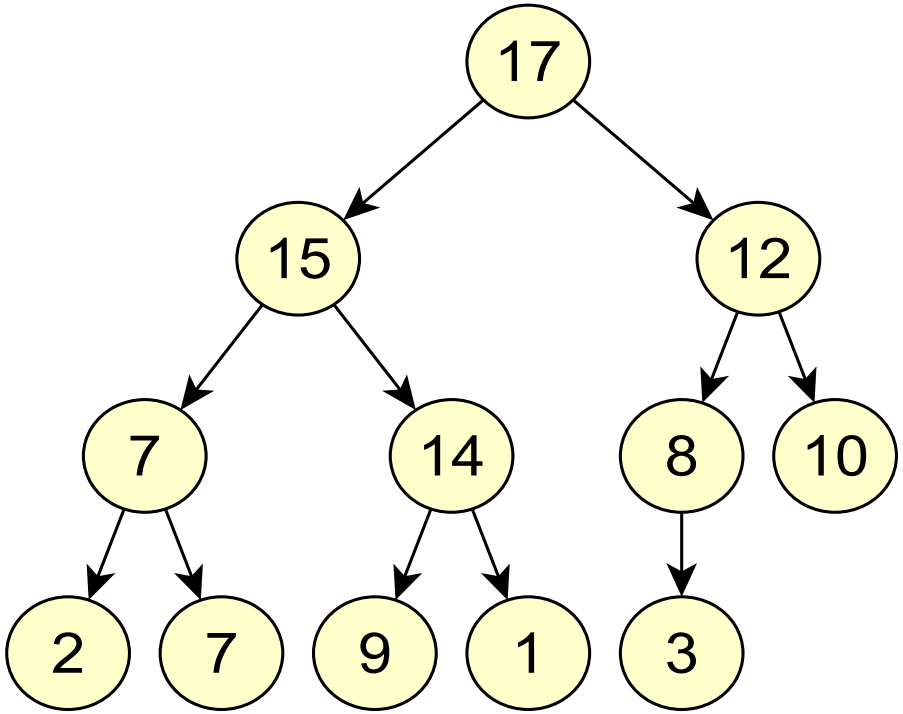
COMP.CS.300 Tietorakenteet ja algoritmit 1

Matti Rintala (matti.rintala@tuni.fi)

Keko

- Binääripuu
- Solmun arvo ≥ lapsien arvot (max-keko)
- tai solmun arvo ≤ lapsien arvot (min-keko)
- Täydellisesti tasapainotettu
  - Haarojen pituusero korkeintaan 1
  - (Pisimmät haarat vasemmalla)




17
15
12
7
14
8
10
2
7
9
1
3

Kekoon lisääminen (Heap Insert)

Heap-Insert(*root*, *node*):

1 ▷ Add *node* as the lowest-level rightmost leaf.
2 while *node* ≠ *root* **and**
node.value > parent*(node).value*
3 *node.value* ⇄ parent*(node).value*
4 *node* := *node.parent*

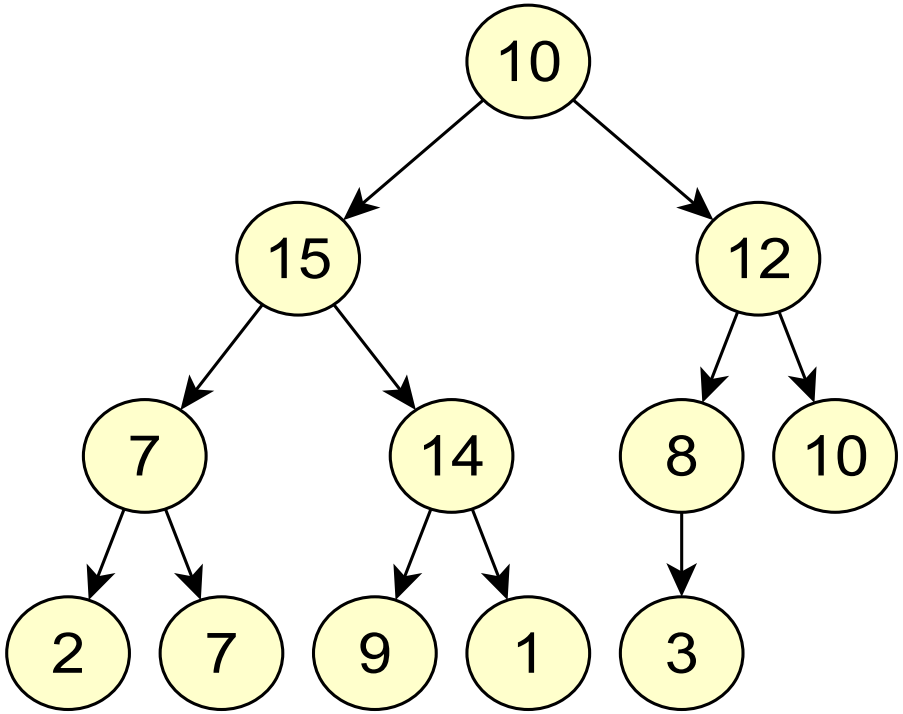
Idea: Laitetaan uusi alkio keon “pohjalle” ja liu’utetaan sitä ylöspäin, kunnes se on arvonsa kannalta mahdollisessa paikassa.

Keon korjaaminen (Heapify)

- Oletus: keko ok, paitsi ehkä *node*

Heapify(*root*, *node*):

```
 1 repeat
 2       orig_node := node
 3       l := Left-Child(node)
 4       r := Right-Child(node)
 5       if l exists and l.value > node.value then
 6           node := l
 7       if r exists and r.value > node.value then
 8           node := r
 9       if node ≠ orig_node then
10           node.value ⇄ orig_node.value
11 until node = orig_node
```


10
15
12
7
14
8
10
2
7
9
1
3

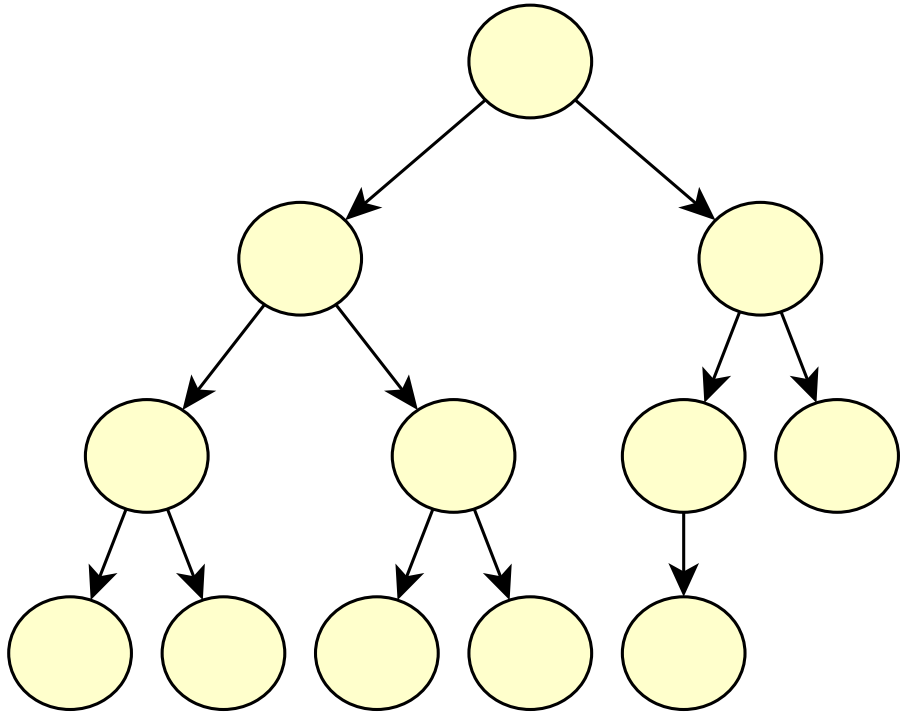
Keon luominen (Build Heap)

- Oletus: Puun alkiot missä tahansa järjestyksessä

Build-Heap(*root*):

1 for *node* **in** ▷ *heap* ***non-leaf*** *nodes in bottom-to-up, right-to-left order*

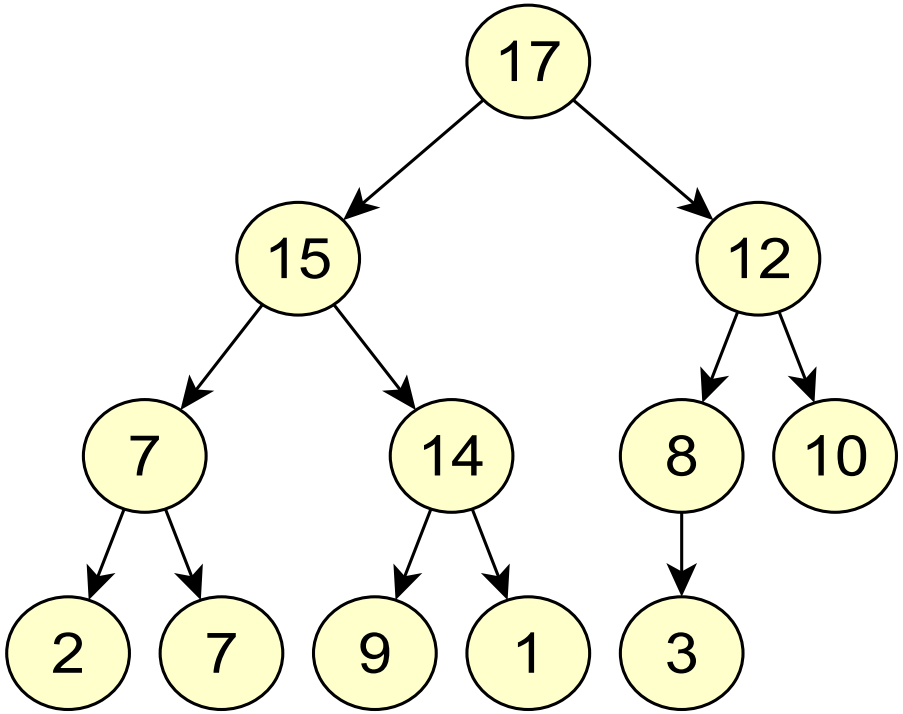
2 Heapify(*root*, *node)*



Suurimman poistaminen (Heap Extract Max)

Heap-Extract-Max(*root*):

1 max := *root.value*
2 node := ▷ lowest-level rightmost leaf
3 root.value := *node.value*
4 ▷ remove *node*
5 Heapify(*root, root)* (Heapify from root down)
6 return *max*


17
15
12
7
14
8
10
2
7
9
1
3

Tampereen yliopisto
Tampere University

Keon operaatioiden tehokkuus

COMP.CS.300 Tietorakenteet ja algoritmit 1

Matti Rintala (matti.rintala@tuni.fi)