



INF2220: algorithms and data structures

Series 4

Topic Priority Queues and Heaps

Issued: 14. 09. 2017

Classroom

Exercise 1 Draw the structure of a binary heap

1. after these priorities have been inserted:
19, 34, 23, 16, 54, 89, 24, 29, 15, 61, 27.
2. after two `deleteMin` operations.

Exercise 2 (Binary heap) Draw the structure of a binary heap

1. after these priorities have been inserted:
10, 12, 1, 14, 6, 5, 8, 15, 3, 9, 7, 4, 11, 13, 2.
2. after three `deleteMin` operations.

Exercise 3 (Leftist Heap I) Is the tree a *leftist heap*? Explain your answer.

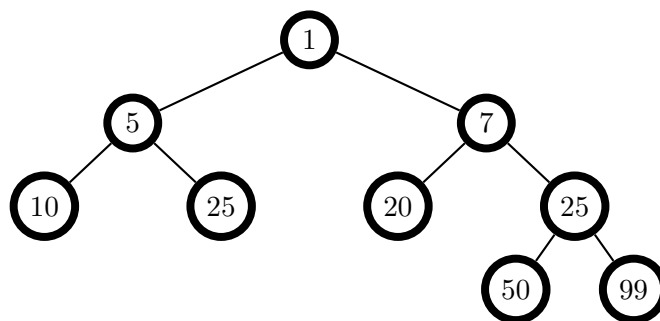


Figure 1: Exercise 3

Exercise 4 (Leftist Heap II) 1. Merge the two leftist heaps. Remember to maintain the structure of a leftist heap.

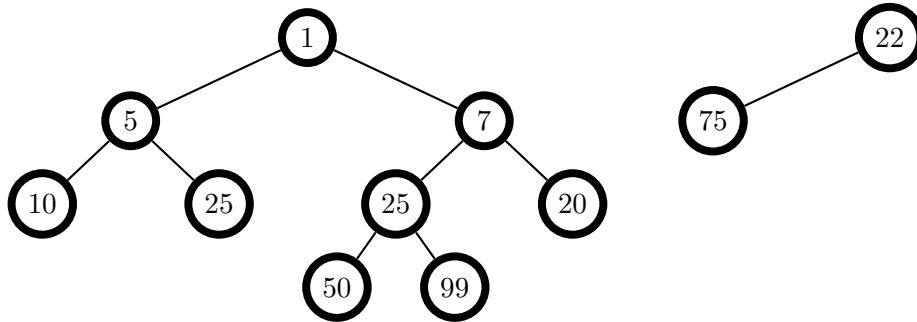


Figure 2: Exercise 4.1

2. Perform a `deleteMin` operation on the resulting leftist heap from Exercise 4.1

Exercise 5 Show that the element with the lowest priority (highest number)

1. can be found in one of the leaf-nodes
2. that it can be found in an arbitrary leaf-node (it's sufficient to show that the requirements of a binary heap holds with different arrangements)
3. show that a binary heap with an even number of elements (N) will have exactly $N/2$ leaf-nodes.

This assignment is perhaps a bit difficult, try to think about the data-structure we used to represent a binary heap

Exercise 6 A file contains only spaces and digits in the following frequency: space (9), a (5), b (1), d (3), e (7), f (3), h (1), i (1), k (1), n (4), o (1), r (5), s (1), t (2), u (1), v (1). Construct the Huffman code.

Exercise 7 (Huffman) A file contains only colons, spaces, newlines, commas, and digits in the following frequency: colon (100), space (605), newline (100), comma (705), 0 (431), 1 (242), 2 (176), 3 (59), 4 (185), 5 (250), 6 (174), 7 (199), 8 (205), 9 (217). Construct the Huffman code.

Lab

Exercise 8 Implement the interface below with a `BinaryHeap`.

```

interface HeapInterface<T>{

    public void insert(int pri, T o);
    public T deleteMin();
}
  
```

Exercise 9 Make a program which sorts a sequence of numbers using a heap.

Exercise 10 A binary heap is a special case of a `DHeap`, where the heap is allowed to have D children. i.e., a binary heap is a `DHeap` with $D == 2$. Implement a

TernaryHeap (DHeap with $D == 3$)

HINT:

now we get three children to examine to find out whether or not they have a higher priority than ourselves, functions to calculate where the elements can be found in the array is given below.

```
private int dad(int pos){
    return (int) Math.ceil( ((double) pos - 1.0) / 3.0);
}

private int first(int pos){ return (3 * (pos - 1)) + 2; }
private int second(int pos){ return (3 * (pos - 1)) + 3; }
private int third(int pos){ return (3 * (pos - 1)) + 4; }
```

Exercise 11 What is the maximum height of a TernaryHeap with N elements?