DNHI Homework 7
Trees

Problem 1

A priority queue containing characters is implemented using max-heap and stored as an array. The capacity of the array used is 10 elements and the first 7 locations are occupied (indexes 0 through 6). Show the array after each of the following operations is performed (each operation should be modifying the array resulting from the previous step).

index	0	1	2	3	4	5	6	7	8	9
value	'z'	'f'	'j'	'e'	'b'	'g'	'h'			

Assume that the name of the priority queue is `pq`.

1. `pq.enqueue('w');`
2. `pq.dequeue();`
3. `pq.enqueue('c');`
4. `pq.enqueue('k');`
5. `pq.dequeue();`
6. `pq.dequeue();`

Problem 2

Given the array representing the max-heap in problem 1, show the tree representation of the heap. Show the tree representation of the final heap in problem 1 (after all of the operations are performed).

Problem 3

Specify the steps needed to sort a list of last names stored in a text file. You should use heap-sort. For each step, specify the complexity (using order notation) of the operations.

Problem 4

Given the following min-heap, show the final state of the heap after executing the following operations: `enqueue(1)`, `enqueue(4)`, `dequeue()`.
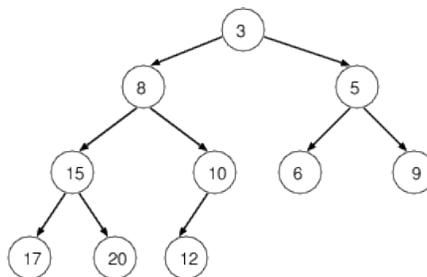
Problem 5

Specify the characteristic properties and requirements of the following trees:

- tree
- binary tree

- binary search tree

- AVL tree

- heap

Problem 6

Given the following node definition for a general tree

```
Node
    int data
    Node firstChild
    Node sibling
```

write a method that given a reference to the node prints to the screen the values stored in all of this node's children.

Problem 7

Enter the following nodes into an AVL tree: 45, 12, 3, 98, 20, 30, 75, 55, 100, 0, 60, 85. For each node that triggers a rotation, state what node it is and show the tree after the necessary rotation.

Problem 8

Given the AVL tree below show what the tree looks like after each of the following nodes is removed: 75, 100, 98.
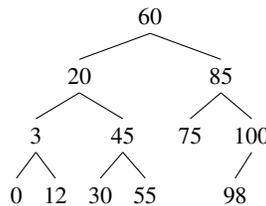
Problem 9

For each node in the tree in problem 8, state what the balance factor of that node is.

Problem 10

Given the following key values: 439, 340 129, 342, 278, 947, 371 and a hash table stored in an array of size 10, show where in the array each of the keys is going to be stored if the hash function is computed as `h(x)= x % 10` and the following collision resolution method is used:

- separate chaining

- linear probing

Problem 11

Explain what load factor has to do with a performance of a hash table.

Problem 12

Given a heap with 3 levels (level 0, level 1 and level 2) what is the largest number of nodes that the heap may contain? what is the smallest number of nodes that the heap may contain?

Problem 13

Consider the implementation of a tree where each node can have arbitrary many children. In the data structure used for storing the tree the children of a given node are organized in a singly lined list using the **sibling** field and the parent has **firstChild** field. Nodes are storing integer values. The declaration of a **Node** class is as follows:

```
class Node {
    int data;
    Node firstChild;
    Node sibling;
}
```

Draw the tree that is represented by the following nodes.