Assignment 2

Due date: 09:00am, Wednesday, the 17th of August 2016.
This assignment is worth of 5% of the total course mark.

General Instructions

You may do this assignment as a **team of two students** if you both are undergraduate students. Individual work of an undergraduate student is also allowed and will be treated in the same way as a team work. Team members may be from different tutorial groups. You have to do it **individually** if you are a postgraduate student.

Submissions have to include coversheet including names, student ids, and your tutorial groups such that submissions can get marked.

Hand in your solutions to the box "ADSA" on level 4, Ingkarni Wardli (close to reception) **by the deadline**. **No late submissions will be accepted.**

Exercise 1 *Induction Proof and Recursion (20 points for 1.1 + 15 points for 1.2)*

1. Let $N = \{0, 1, 2, \ldots\}$ be the set of natural numbers. Access to data structures is often governed by the recurrence

$$T(n) = \begin{cases} a, & \text{if } n = 1 \\ c + T(n/2), & \text{if } n > 1 \end{cases}$$

   Prove by induction that $T(n) \in \mathcal{O}(\log n)$. Do not attempt to use the Master Theorem for this proof.

2. Remind how merge-sort works. Show that recursive merge-sort is $\Theta(n \log n)$ by the master theorem.

Exercise 2 *Algorithm design (15 points)*

Describe a $\Theta(n \log n)$-time algorithm that, given a set S of n integers and another integer x, determines whether or not there exist two elements in S whose sum is exactly x.

Exercise 3 *Binary heaps (20 points for 3.1 + 15 points for 3.2 + 15 points for 3.3)*

1. Assume you have an array-based binary heap a with the contents:

   $$1, 4, 7, 8, 9, 10, 14, 12, 15, 13, 17, 12$$

   Show the contents of a after each of the following two operations. Show your working for each operation including the content of the list at its intermediate stages. You can assume a is large enough to contain all the values inserted.

   - Insert 5 into a.
   - Delete the least element from a.

2. Is it right or wrong that in a heap of depth d, there must be at least 2^d elements. (Assume the depth of the first element (or root) is zero). If it is right, provide your prove. If it is wrong, explain why you think so.

3. Prove that the binary tree represented by the binary heap of n elements has height $k = \lfloor \log n \rfloor$.

End of Questions