1) a) Describe the design paradigm *Dynamical Programming*. Which problem does it address? In which situations
can it be used? (7 Points)
b) Design an algorithm for the *knapsack*-problem: Given an integer $K$ and $n$ integers $1 \leq s_i \leq K$, find a subset of
$S = \{s_i\}$ such that the sum over these $s_i$ is exactly $K$ (or determine that no such set exists). (13 Points)

2) Consider the following problem: You are given a sequence of $n$ integers that contains $\log n$ different integers.
a) Design an algorithm to sort this sequence using $O(n \log \log n)$ element comparisons. (15 Points)
b) Why does this runtime not violate the lower bound of $\Omega(n \log n)$ comparisons? (5 Points)

3) a) A divide-and-conquer algorithm for multiplying two $n \times n$ matrices reduces the calculation to 7 products of
$n^2 \times n^2$ matrices and 18 matrix additions of $n \times n$ matrices. (This addition is given by the rule
$(A + B)_{i,j} = A_{i,j} + B_{i,j}$)
Give a recurrence solution for the runtime $T(n)$ required to multiply two $n \times n$ matrices and give a $O$-estimate for
$T(n)$. (12 Points)
b) Solve the following recurrence relation with full history:
$$T(n) = n + \sum_{i=1}^{n-1} T(i), \quad T(1) = 1.$$ (8 Points)

4) a) Give the definitions of a *bipartite graph*, a *matching* and state the *alternating path theorem*. (7 Points)
Let $G = (V, E)$ be a tree. A *vertex cover* for $G$ is a subset of vertices $U \subset V$, such that every edge is adjacent to at
least one vertex in $U$. In general there are several possible vertex covers, we are interested in those for which the
size of $U$ is minimal, we call such a cover a *minimal vertex cover*.

For example:

b) Show that there is a minimal vertex cover that does not contain any leaves of the tree. *(Hint: The edge to a leaf
can always be covered by the vertex which is parent to the leaf)* (2 Points)
c) Design an efficient algorithm to find a minimal vertex cover. Use the observation from b) to reduce the problem
to a smaller graph. The algorithm should run in time $O(|V|)$ (show this!). (11 Points)