

Foundation of Computer Science — FM2

Assignment 1

1. Which of the following choices is the tightest upper bound for the functions

(a) $f(n) = \frac{1}{2}n(n + 1)$,

(b) $f(n) = \frac{1}{2^n}$,

(c) $f(n) = \frac{n^2}{1+n}$,

$O(n)$, $O(n^3)$, $O(1)$ or $O(n^2)$?

2. Is $f(n) = n \log n$ of order $O(n^2)$? Is $f(n)$ also $\Omega(n^2)$?

3. Illustrate the Mergesort algorithm by sorting the list

(3, 9, 6, 10, 4, 1, 2, 8).

Why does Mergesort follow the divide-and-conquer paradigm?

4. Consider the following problem **Sum of Subset (SOS)**:

Given: non-negative integers $m, a_1, a_2, \dots, a_m, b$

Question: Is there is set $J \subseteq \{1, 2, \dots, m\}$ such that $\sum_{i \in J} a_i = b$?

(a) Solve the SOS problem with dynamic programming.

Hint: Use a table $\text{SUM}(i, j)$ storing the maximal values that can be obtained as a sum of numbers from a_1, a_2, \dots, a_i such that this sum does not exceed the number j .

(b) Find out what the *knapsack problem* is.

How can you modify your algorithm solving SOS in order to solve the knapsack problem?