Algorithms. Assignment 2

In the following exercises, let us treat additions of arbitrary real numbers (not only of digits!) as elementary operations.

Problem 3

Assume that n houses $H_1, \ldots, H_n$ are located (in this order) along a straight road. For all $i = 1, \ldots, n-1$ we know the distance x_i between H_i and H_{i+1}. We want to compute the distances y_{ij} between H_i and H_j, for all pairs (i, j).

(a) A naive algorithm closely follows the problem specification, in that it takes every pair (i, j) and computes y_{ij} from scratch, by summing up the distances in between.
Show that this algorithm runs in $O(n^3)$ time, and no faster. (That is, explain also why the algorithm actually needs cubic time, and that it is not just your analysis which is too generous.)

(b) Give a more clever algorithm that needs only $O(n^2)$ time. Do not forget to explain the claimed time bound.

(c) Can the time bound $O(n^2)$ be further improved? Why, or why not?

Problem 4

A warehouse is divided into n rooms of sizes $s_1 \geq \ldots \geq s_n$. Here, the sizes are already sorted in descending order. We would like to rent storage space of size exactly s. But only complete rooms can be rented, and none of the given sizes equals s. The next option is to rent two rooms of total size s, that is, find two indices i and j with $s_i + s_j = s$, or figure out that no 2-room solution exists. We can trivially solve this problem in $O(n^2)$ time, by generating all pairwise sums and comparing each one to s.

(a) Give a more clever algorithm that needs only $O(n)$ time.

(b) How can you be sure that your algorithm proposed in (a) cannot miss an existing solution by mistake?