BIL-533

HW 3

***2.1-2.***

Rewrite the INSERTION-SORT procedure to sort into non-increasing instead of non-decreasing

order.

***2.1-4***

Consider the problem of adding two *n*-bit binary integers, stored in two *n*-element arrays *A* and *B*. The sum of the two integers should be stored in binary form in an -element array *C*. State the problem formally and write pseudocode for adding the two integers

***2.2-2***

Consider sorting *n* numbers stored in array *A* by first finding the smallest element of *A* and exchanging it with the element in . Then find the second smallest element of *A*, and exchange it with . Continue in this manner for the first elements of *A*. Write pseudocode for this algorithm, which is known as ***selection sort***. What loop invariant does this algorithm maintain? Why does it need to run for only the first elements, rather than for all *n* elements? Give the best-case and worst-case running times of selection sort in Θ -notation. (Since we have considered the selection algorithm’s pseudocode, you are supposed to calculate the best-case and worst-case running times of selection sort in Θ -notation)