**CECS 302 Homework Assignment 4 (Due 4/24/23)**

**1) (30 pts)** This problem deals with **sorting algorithms**.

a. Explain the difference between the terms **Maintained Sorting** and **Sorting On Demand**. Provide at least one example of an application where each one is preferable.

b. Explain the principal differences between the **Insertion Sort** and **Heap Sort** algorithms. Be certain you explain i. any requirements the algorithms have to be used, ii. a general idea of how the algorithms operate, and iii. which algorithm is the most efficient in the **worst-case scenario**. Your response should be 1-2 paragraphs in length.

**2) (30 pts)** This problem deals with **sets** and **set implementations.**

a. Explain the principal differences between the **Unsorted List ADT** and the **Set ADT**.

b. Provide an example when using the bit-vector implementation of a set can be advantageous, and another example when it is a poor choice.

**3) (40 pts)** This problem deals with **hash tables** and **hashing functions.** Your responses to each sub-part should be 2-5 sentences in length.

a. Explain the principal efficiency advantage of hash tables in comparison to other data structures we have observed thus far.

b. Explain what issues can hamper this performance advantage, and what steps are taken to mitigate the slow-down.

c. Suppose employees at a company are assigned an 8-digit employee ID according to the following criteria:  
i. The first four digits correspond to a department code (ex: the accounting department may have code 0051, the IT department code 3505, etc.)  
ii. The second four digits correspond the order in which the employee was hired (ex: the second employee would have the last four digits “0002,” the one-hundredth “0100,” etc.)  
Now suppose you are given the task to store employee information according to employee ID in a small hash table and are given the following options for the hashing function:   
i. Compute the hash index using the **first** four digits of the employee ID, mod 53.   
ii. Compute the hash index using the **last** four digits of the employee ID, mod 53.  
Assuming the company has 100+ employees and 5 departments, explain which approach is better for this dataset, and why. Should your selection change if option ii only used the last *two* digits of the ID, mod 53. Explain why or why not.

**Responses to all problems should be in .doc(x) or .pdf format. Please submit your assignment to Blackboard as a single file (not zipped, in this case) with the filename “LN\_FN\_4” where LN is your last name and FN is your first name.**