Today: HW04 part 2 due. Sorting; §§ 8.{1–4}.
Next class: Term project proposals (PDF, online submission). Sorting; §§ 8.{5–8}, 21.{5,6},
Reminders: HW02 extra credit options. Term projects. Midterm exam 2 next week.

1. Write your group identifier (e.g., C3) and its members' names Underline your name.

2. Trace the action of *bubble sort* on the following array. (a) Depict the state of the array after each iteration of the outer loop. (b) Also depict the state of the array after each iteration of the inner loop for the fourth iteration of the outer loop.

   ```
   59 84 36 43 30 50 13 25 56 21
   ```

   ```
   59 36 43 30 50 13 25 56 21 84          30 36 43 13 25 50 21 56 59 84
   36 43 30 50 13 25 56 21 59 84          30 36 43 13 25 50 21 56 59 84
   36 30 43 13 25 50 21 56 59 84          30 36 13 43 25 50 21 56 59 84
   30 36 13 25 43 21 50 56 59 84          30 36 13 25 43 50 21 56 59 84
   30 13 25 36 21 43 50 56 59 84          30 36 13 25 43 50 21 56 59 84
   13 25 30 21 36 43 50 56 59 84          30 36 13 25 43 21 50 56 59 84
   13 25 21 30 36 43 50 56 59 84
   13 21 25 30 36 43 50 56 59 84
   13 21 25 30 36 43 50 56 59 84
   13 21 25 30 36 43 50 56 59 84
   ```

3. Repeat Question 2 for *selection sort.*

1

4. Repeat Question 2 for *insertion sort*.

    59 84 36 43 30 50 13 25 56 21

5. A *subsequence* of sequence S is any sequence that can be obtained from S by deleting zero or more of its elements. For example, $(1, 4, 9, 2)$ is a subsequence of $S_1 = (3, 1, 4, 5, 9, 2)$, but $(1, 9, 4)$ is not. A subsequence S' of S is called a *k-subsequence* if each pair of adjacent elements in S' has $k-1$ intermediate elements in S. For example, $(1, 5, 2)$ is a 2-subsequence of S_1, and $(3, 5)$ is a 3-subsequence of S_1, but $(1, 5, 9)$ is not a k-subsequence of S_1 for any value of k (although it is a subsequence of S_1). A k-subsequence with n elements is called *maximal* if there is no k-subsequence with $n + 1$ elements.

    (a) List all *maximal 5-subsequences* and *maximal 3-subsequences* of the sequence of Question 2.

    (b) Trace the action of *shell sort with increments 1, 3, 7* on the array of Question 2. Use selection sort to sort by each increment. Depict the state of the array after each sorting by each increment is completed.

    (c) How many maximal g-sequences does a sequence of n items have (as a function of g and ni?

    59 84 36 43 30 50 13 25 56 21

6. Count the exact number of *comparisons* and *swaps* made by each application of sorting in the above questions.