1. Suppose we are comparing implementations of insertion sort and merge sort on the same machine. For inputs of size $n$, insertion sort runs in $8n^2$ steps, while merge sort runs in $64n \log n$ steps. For which values of $n$ does insertion sort beat merge sort?

2. For each of the following running time functions $f(n)$, compute how long a problem of size $n = 1000$ will take to run if $f(n)$ is measured in microseconds. Convert the time for each function to the largest unit in \{seconds, minutes, hours, days, months, years, centuries\} that gives a value of at least 1.

   $\lg n \quad \sqrt{n} \quad n \quad n \log n \quad n^2 \quad n^3 \quad n^4 \quad 2^n \quad n!$

3. Consider the searching problem:

   Input: a sequence $A = \langle a_1, a_2, \ldots, a_n \rangle$ and a value $v$
   Output: an index $i$ such that $v = A[i]$ or NIL is $v$ is not found

   The linear search algorithms scans through the sequence one element at a time until an element $A[i] = v$ is found, at which point it stops and returns $i$.

   (a) Write pseudocode for a linear search.
   (b) Using a loop invariant, prove the algorithm is correct. Make sure your loop invariant satisfies the 3 necessary properties.
   (c) What is the best case, worst case, and average case running time for linear search, using $\Theta$ notation? For the average case, assume $v$ is equally likely to match any element in the array. Justify your answers.

4. Consider the bubble sort algorithm:

   \[
   \text{BubbleSort}(A) \\
   \text{for } i \leftarrow 1 \text{ to } n \\
   \quad \text{do for } j \leftarrow n \text{ downto } i + 1 \\
   \quad \quad \text{do if } A[j] < A[j - 1] \\
   \quad \quad \quad \text{then exchange } (A[j], A[j - 1])
   \]

   (a) State precisely a loop invariant for the inner for loop and prove that the invariant holds. Follow the structure used in the book and in class.
   (b) Using the termination condition of the above loop invariant, state a loop invariant for the outer for loop. Use this loop invariant to prove that the algorithm is correct.
   (c) What is the worst case running time for bubble sort, in $\Theta$ notation? Justify your answer.

5. Prove by induction that the $i^{th}$ Fibonacci number satisfies the equality $F_i = (\phi^i - \hat{\phi}^i)/\sqrt{5}$, where $\phi = (1 + \sqrt{5})/2$ is the golden ratio and $\hat{\phi} = (1 - \sqrt{5})/2$ is its conjugate.

6. Explain why the following statement is meaningless: “The running time of algorithm $A$ is at least $O(n^2)$.”

7. Prove Theorem 3.1 on page 46.

8. Prove the following using the definitions of $O$, $\Omega$, $\Theta$, $\omega$ and $\theta$, and the theorems in your book.
   (a) $2^{n+1} = O(2^n)$
   (b) $n \log n = o(n^2)$
   (c) $\ln n = \Theta(\log_2 n)$
   (d) $n^\epsilon = \Omega(\log n)$ for any $\epsilon > 0$
   (e) $n! = \omega(2^n)$
   (f) $\log(n!) = \Theta(n \log n)$