CSE 373 Midterm Review List
OPEN BOOK, OPEN NOTES

1. Complexity

   - Be able to perform an analysis of a given algorithm to determine the "number of statements executed" T(n) by the algorithm for some given number of inputs n. Be able to convert this result to Big-O notation. Be able to analyze either iterative or recursive procedures. (You will not need to formally solve recurrence relations.)
   - Be able to compare the time complexities of various algorithms using Big-O notation.
   - Be able to explain and illustrate the difference between order of complexity and constant of complexity.

2. Lists, Stacks, and Queues

   - Be familiar with the basic operations for lists, stacks, and queues; be able to use them as needed.
   - Be able to compare the algorithms for these operations with respect to sequential and linked implementations. Comparisons can be about what they do, the time complexity, and the required space.
   - Be able to write recursive or nonrecursive procedures dealing with linear structures.

3. Trees

   - Be familiar with the abstract operations for binary search trees. Be able to use them as needed or to show what they do to a given tree.
   - Be able to write recursive or iterative functions that operate on general trees, plain binary trees, or binary search trees.
   - Be able to compute balance factors for the nodes of binary search trees.
   - Be able to show how the Insert operation works on an AVL tree, including the rebalancing operations given in the text.
   - Be able to show how splaying is done in a SPLAY tree.
   - Be able to show how Insert and Find work for a B+ tree.
   - Be able to explain the time complexity of any of the above algorithms.

4. Hashing

   - Be able to answer questions about hashing concepts.
   - Be able to show how hashing with a given hash functions works for separate chaining or for open addressing with linear or quadratic probing or with double hashing.
   - Be able to answer complexity questions about hashing.