Test Format
- Multiple choice
- Short Answer
- Traces
- Programming

Know the following

- Algorithm analysis
- copy constructor
- dangling reference
- Big-oh
- .h, .cpp
- null pointer
- class
- Sequential search
- dereferencing operator
- private
- binary search
- base case
- public
- selection sort
- general case
- member data
- insertion sort
- recursion
- method
- pointer
- tail recursion
- default constructor
- dynamic array
- activation record
- parameterized constructor
- memory leak
- run-time stack

- Be able to declare a class with all 3 types of constructors, define some of its methods, declare an instance of a class, and call a method of a class.

- Be able to show the trace of sequential and binary search
- Be able to show the trace of the selection and insertion sorts
- Be able to compute the number of comparisons made by the search and sort algorithms
- Be able to discuss the best, average and worst case complexity of the search and sort algorithms and know the criteria for choosing one algorithm over another
- Be able to write the two search and two sort algorithms described in class or recognize an error in one of the algorithms given

- Know the 4/5 steps to write a recursive function (that is, determine input space, what the result should be, any necessary formulas, base case(s) and the general case(s).)
- Know what 3 conditions mark a successful use of recursion (that is, what you need to do to verify a recursive function is correct, that is check working base case(s), check general case(s) approach base case(s), and assuming recursive call(s) work, show general(s) work.)
- Be able to trace a recursive function
- Be able to write a recursive function
  - Some example: Fibonacci, factorial, getSum, recursiveSequentialSearch, recursiveBinarySearch

Study
- Chapters 7, 8, 9, 10, 14 and the Checkpoints and Review Questions at the end of sections and chapters, especially
  - Pg. 498: 4, 7, 8
  - Pg. 642: 2, 3, 6, 7
  - Pg. 946: 1, 2, 4, 5, 6, 8
- Lecture slides on Searching, Sorting, Recursion
- Homeworks, Quizes, and program assignments