COMI 2510 Advanced Programming and Design
Lesson 8: Sorting, Searching, and Algorithm Analysis
Lab

1. Complete this on your own. Review the first video for a brief tutorial on using VisuAlgo before you begin.

Visit the following website:
https://visualgo.net/en

Step through the following algorithms until you can describe how they work and trace algorithm on your own.

- Bubble Sort
- QuickSort
- Selection Sort
- Insertion Sort

2. Efficient computation of Fibonacci numbers and factorials.

a) In the last lab, you derecursed a recursive factorial function. Return to that code and modify it so it outputs the elapsed time in both seconds and milliseconds for both algorithms. Create an application that calls both methods and generates output showing the time required to compute the factorials of 8, 9, 10, 11, and 12.

b) Modify the application in Code Listing 16-7 (FibNumbers.java) by replacing the recursive method for computing Fibonacci terms with an iterative version and try to make it as efficient as possible. Modify the application so it outputs the elapsed time in both seconds and milliseconds. Finally, run the application and generate output showing the time required to compute the terms of the sequence in positions 45, 46, 47, 48, 49, and 50.

3. Search comparison

Complete the following table calculating the average and maximum number of comparisons the sequential search will perform, and the maximum number of comparisons the binary search will perform.

	Array Size				
	50 elements	500 elements	10^4 elements	10^5 elements	10^6 elements
Sequential search (average comparisons)					
Sequential search (maximum comparisons)					
Binary search (maximum comparisons)					

In an average case involving an array of n elements, how many arrays elements does a sequential search algorithm have to access to locate a specific value?