

UOP ECET 370 Week 5 Lab 5 Search Algorithms and Techniques

Check this A+ tutorial guideline at

<http://www.assignmentcloud.com/ecet-370/ecet-370-week-5-lab-5-search-algorithms-and-techniques>

For more classes visit

<http://www.assignmentcloud.com>

General Instructions

Exercise 1 uses the programs in DocSharing labeled "Search algorithms."

Exercise 1: Review of the Lecture Content Create a project using the ArrayList class and the Main class provided in DocSharing. The ArrayList class contains implementations of the first three search methods explained in this week's lecture: sequential, sorted, and binary search. The Main class uses these three methods. These programs test the code discussed in the lecture. Compile the project, run it, and review the code that is given carefully.

Exercise 2: Search Algorithms and Techniques Expand the project developed in the previous exercise to perform the following experiment: time the three search methods several times each and record the results. Compare the running times of the three search methods (sequential search, sorted search, and binary search) which are obtained during the experiment. What conclusions can be drawn?

Exercise 3: Searching Applications Select one of the following two problems to solve:

Problem 1: Design and implement an algorithm that determines whether or not a given array of elements, list1, is completely contained within another given array of elements, list2. Consider two different scenarios: 1) both arrays are sorted; 2) both arrays are unsorted. **Problem 2:** Design an algorithm that when given a collection of integers in an unsorted array, determines the second smallest number (or second minimum). For example, if the array consists of the values 12, 23, 9, 17, 3, the algorithm should report the value 9, since it is the second smallest number in the array. Write a function that receives an array as a parameter and returns the second smallest number. To test your function, write a program that populates an array with random numbers and then call your function. **Exercise 4: Hashing** Suppose that the type of key in a hashing application you are implementing is String (Sections 19.6 and 19.7 in our textbook explain hash functions for strings). Design, implement, and test a hash function that converts a key to a hash value. Assume that the size of the hash table is a prime number.