

IC8
ITEC 2150 Spring 2013
Searches

1. Create a Java program that asks the user if they want to search an unsorted or sorted list for a number of their choice.
2. Generate a list containing 1000 numbers.
3. Create a BinarySearch class. This class should have a method search which will take an Integer ArrayList and the number the user wants to search for. This method will be used to search the sorted list.
 - a. This method should divide the list in half and test if the number is the same as the mid-point. If it is, the list returns.
 - b. If the midpoint of the list is less than the number the user wants to search for, add one to the midpoint, divide the difference between the new low index and the existing high index in half and add to the low index. Repeat the test to see if the midpoint is the number.
 - c. If the midpoint is higher than the number the user wants to search for, subtract one from the midpoint for the new high index and divide the difference of the high index – low index in half and add this to the low index. Run the test again.
 - d. Return the number of comparisons done.
4. Create a LinearSearch class. This class should have a method search which will take an Integer ArrayList and the number the user wants to search for. Before calling this method, the list should be shuffled. Set up a loop to step through the list one by one and check to see if the number matches the number in the list at the index position. If it matches, return the number of times the numbers were compared.
5. If the number is not found for either search, return the negative of the number of searches performed.
6. Display either the number of searches to find the item or the number of searches executed without finding the number.
7. Bonus – resort your list after performing an unsorted list so the sorted search will work correctly.
8. Rubric – note both function and code structure will be graded to reach these points.
 - a. Coding standards 20 points
 - b. Linear Search 30 points
 - c. Binary Search 40 points
 - d. Test program 10 points
 - e. Sort + 20 points. Only available if you get the above code to work.