CS 101 - Algorithms & Programming I

Spring 2022 - HOMEWORK 2

Due: May 13, 2022 23:30

1. Implement a Java class to represent objects of your choice. Your class should be complete and should include private attributes, static attributes, accessor/mutator methods, toString/equals methods. You should also implement a method, lessThan() that takes an object of the same type as a parameter and returns true if this object is less than the parameter, false if not. You will determine the attributes that make an object less than another. Implement any other functionality required for a complete class.

2. In the application class, **implement** a method to sort a list of objects (type defined in step 1). You may use the sorting algorithm of your choice (examples: bubble, selection, insertion) however you should not use any sort methods in the JDK/third party sort methods.

3. In the application class, **implement** a binary search method that searches a list of objects and returns the index of the matching object if it exists in the list, -1 if the object does not exist.

   Binary search algorithm:

```
Procedure binary_search
   A ← sorted array
   n ← size of array
   x ← value to be searched

   Set lowerBound = 1
   Set upperBound = n

   while x not found
      if upperBound < lowerBound
         EXIT: x does not exists.
      set midPoint = lowerBound + ( upperBound - lowerBound ) / 2
      if A[midPoint] < x
         set lowerBound = midPoint + 1

      if A[midPoint] > x
         set upperBound = midPoint - 1

      if A[midPoint] = x
         EXIT: x found at location midPoint
   end while
end procedure
```

4. In your application,

a. Create a list of objects (either ArrayList or array) and sorts the list using your sort method. Include a comment - what is the efficiency of your selected sort algorithm?
b. Search the list of objects and display the matching object, or an appropriate message if the object is not found. Include a comment - what is the efficiency of the binary search algorithm?

Note: your Class and application should be unique and should not be the same/similar to examples from the class/lab/text.