COSC 1437 Programming Lab & Assignment (Fall 2016)

Topics

- **`String`** processing, **`ArrayList`**, text file handling, sequential search, binary search, recursion

Activities

- Write a program which will read a text file into an **`ArrayList`** of **`String`**s. Note that the given data file (i.e., "sortedStrings.txt") contains the words already sorted for your convenience.
- Read a search key word (i.e., string) from the keyboard and use sequential and binary searches to check to see if the string is present as the instance of **`ArraryList`**.
- Refer to "**SearchInt.java**" (given in "SearchString.zip") and the following UML diagram for the details of required program components (including the class name, data fields, and methods) for this assignment.

SearchString
+TO_STOP: String **+NOT_FOUND: int** **+count1: int** **+count2: int** **+count3:int**
+sequentialSearch (array: ArrayList<String>, value: String): int **+binarySearch(array: ArrayList<String>, value: String): int** **+binarySearch(array: ArrayList<String>, low: int, high:int, value: String): int** **+main(args: String[]): void** **+main(): void**

- HINTS & STEPS:

  (H1) Declare the two symbolic constants as public, static, and final. To be more specific, declare them as follows: **public static final String TO_STOP = "-1";** and **public static final int NOT_FOUND = -1;**

  (H2) Declare **count1, count2,** and **count3** as static, each of which keeps track of number of comparison operations in **sequentialSearch()**, iterative **binarySearch()**, and recursive **binarySearch()**, respectively.

  (H3) **binarySearch(array: ArrayList<String>, value: String)** is the header for iterative binary search method and **binarySearch(array: ArrayList<String>, low: int, high: int, value: String)** is the header for recursive binary search method.

  (H4) Every search method should be written a separate method, which **returns the index in the instance of ArrayList if the search key word exists**, or **returns -1(i.e., NOT_FOUND) if the search key word doesn't exist.**

  (H5) First, declare all the first four methods (i.e., sequentialSearch(), the two binarySearch()s, and main()) to be static (i.e., static(=class) methods) and complete the remaining program accordingly. After you confirm that your program works properly as shown in the running example,**declare the first three methods (sequentialSearch() and the two binarySearch()s) to be non-static (i.e., non-static(=instance) methods, commenting out the corresponding method headers. What do you need to change in your static main() if you want to use the non-static(=instance) methods.** Briefly discuss (or explain) this in your program file.

(H6) Declare the fifth method (i.e., main()) as a non-static (i.e., instance) method in "SearchString.java". Notice that this non-static main() does not have parameters and thus this main() is different from the static main(String[] args) in terms of using a different modifier as well as a different signature. By the way, is it fine to declare main() like this? If yes, what are you practicing now with this? Overloading or overriding? Otherwise, explain why it is not valid.

(H7) Implement a driver class (say "TestSearchString.java") and show how to use/call the methods in "SearchString.java" to get the same output shown as running example below. Briefly discuss (or explain) how to call static(=class) methods and non-static(=instance) methods in "SearchString.java" from the driver ("TestSearchString.java").

TestSearchString
+main(args: String[]): void

(H8) What do you need to change if you change the access modifier from public to private after step (H7) for all the constants and variables in the data field? How about if you change all the occurrence of public (i.e, both in data field and each method header) as private? Were you able to run your program either in SearchString.java or in TestSearchString.java?

(H9) What if you don't have the two main()s in SearchString class. What do you need to change in main() in TestSearchString class to get the exactly same result you get at step (H7) with either all public data fields or all private data fields?

Running Example

Type the filename ot read: **sortedStrings.txt**
13040 words are populated in the instance of ArrayList.
Type a word to search (-1 to stop): **computer**
sequentialSearch() : computer is found in [2424] (comparison=2425).
iterative binarySearch(): computer is found in [2424] (comparison=11).
recursive binarySearch(): computer is found in [2424] (comparison=11).
Type a word to search (-1 to stop): **science**
sequentialSearch() : science is found in [10027] (comparison=10028).
iterative binarySearch(): science is found in [10027] (comparison=13).
recursive binarySearch(): science is found in [10027] (comparison=13).
Type a word to search (-1 to stop): **java**
sequentialSearch() : java is not found (comparison=13040).
iterative binarySearch(): java is not found (comparison=13).
recursive binarySearch(): java is not found (comparison=13).
Type a word to search (-1 to stop): **-1**
Press any key to continue …

What to Hand in

- Submit the two java programs (i.e., "SearchString.java" and "TestSearchString.java") via Blackboard.