Asgn 3 Demo Specs (& Related Notes)

WHAT TO DO FOR THE DEMO

1. Manually delete LogSession.txt file
2. Run Setup program
3. Run ShowIndexFileUtility program
4. Run UserApp program
5. Run ShowIndexFileUtility program
6. Print LogSession.txt file (from NotePad or WordPad – see note on right regarding nice formatting – DO NOT EDIT THE FILE ! ! !)
7. Print all your program code files.
8. Circle what I've described below (by hand).

WHAT TO HAND IN **(all in the order specified below)**

1. Cover sheet (signed)
2. Printout of LogSession.txt file
3. The program code: (IN THIS ORDER)
   - Setup program
   - UserApp program
   - RawData class
   - CodeIndex class
   - UI class
   - any other code files/classes used in your program

HOW MUCH COMMENTING IS NEEDED?

- top-comments at the top of each physical file
- visual separators between methods (e.g., a comment-line of *'s)
- any "tricky"/unusual code

NOTE on NICELY FORMATTED OUTPUT

- You MUST USE a FIXED-WIDTH FONT (like Lucida Console or Courier New or…) so everything lines up nicely in the LogSession file's printouts.
- Use a smaller font size and/or landscape orientation to **eliminate wrapped-around or truncated lines**

SOME NOTES:

- ReRead specs for A3
- YES, you have to use OOP and the programs/classes/methods specified
- QC & IN & DC transactions & SetUp MUST use public methods in CodeIndex class: Search, Insert, Delete (or similar names) to do the actual handling
- HashFunction method must be called HashFunction – it must be a SINGLE private method in CodeIndex – it can not just be some "chunk of code" embedded in some other method
- UserApp itself contains the "big controller" (e.g., a switch statement based on tranCode) to decide which CodeIndex service method to call to actually handle the transaction
- RawData contains 2 strings & 2 getters for id and code

CIRCLE BY HAND:

- the method called HashFunction
- Any use of the constant literal 20 (where MAX_N_HOME_LOC should have been used instead)
- Inside Search (and also Delete, if it does its own searching rather than calling Search) the INCREMENT line where you move to the next node in the search path