Assignment 2 -- Part II (User Docs)
March 3, 2019

Overview

As our software gets more complex, user docs become warranted. This is the case with Assignment 2.

So Assignment 2 now consists of two assignments (Part I -- already submitted) and Part II described here. Each is worth 200 points (400 points total).

Documents

The user docs should include at least the following:

1. README-FIRST.TXT -- 2 Points
2. Installation Guide (txt or pdf) -- 13 Points
3. User Guide (txt or pdf) -- 25 Points
4. JAVA files that demonstrate your software -- Nine or more JAVA files -- 80 Points
5. Example Output -- Nine or more TXT files -- 80 Points

Of course the JAVA files that are part of user docs need to work properly as they demonstrate and exercise your software.

A ZIP file "Statistics Software.zip" is attached to this assignment, and is an example set of user docs. Please follow the format of these docs to create your own.

1-3: Guides

See "README-FIRST.TXT" for reference (in ZIP).

Your "Installation Guide" should provide step-by-step instructions to your customer for installation of the software. See "Installation Guide.pdf" (in ZIP).

Your "User Guide" provides step-by-step instructions to help your customer use your software. See "User Guide.pdf" (in ZIP).

4: JAVA Files That Demonstrate the Software

Programmer customers tend to be example-oriented. Thus you want to provide clear and concise examples that show how to use your software.

See "src\examples\StatisticianExamples.java" for reference (in ZIP). Basically this is just a place to demo how to construct and use objects using your software.

Your software customers are interested in your Stack, Queue, Bag, Deque, and IndexableList. Both linked and arrayed. So your examples should demonstrate how to construct and use objects of these types. The key functionality (that you want to demo) of these objects is:

- add (e.g. push, enqueue, etc)
- remove (e.g. pop, dequeue, etc)
- peek (e.g. peek)
- traversal (the most important functionality -- i.e. we want to show the customer how to get an iterator from each structure (using "asIterator") and then how to use the iterator (simply with "hasNext" and "next").

Note that "push" is basic core functionality while "pushAll" is convenience (and similar for the other data structures). We can focus on basic core functionality for our customers.

5: Example Output

See the "Example Output" directory (in ZIP). This may be the most important part of your docs. Showing your customer actual output from your software shows the user how your software should work, and gives them a target to strive for as they get your software up and running.

Hints

- Keep the docs as concise and clear as possible.
- Iterate on the docs. The first pass maybe would demo just object construction. Then iterate and improve the docs a few times.
- Although you do **not** have to provide docs about objects behind the scenes (e.g. linked lists, nodes, etc) it is suggested that you do "mini-docs" for your own use and to benefit your own dev process.

Assignment 2 Part I

- You have already submitted Part I which are the solutions to the Assignment 2 problems
- Sometimes when preparing user docs, you may decide to make tweaks/change your software. You may re-submit your software for Part I, and an average of your first and second submit will be taken for a score
- If re-submitting your software, please use the latest version of the "interfaces". The latest are in github these. The instructions are:

    1. Go to:
        github (https://github.com)
    2. Search for:
        gary-peterson/linear-data-structures
    3. Open Project:
        linear-data-structures
    4. Download:
        latest version of "interfaces.zip"