Lab 3 Feedback Dr. Yoder, Spring 2014, CS2852

You don't need to print this page. But it could be handy for reference.

Overall

-2 – Sections out of order
-2 – Cover missing

UML Diagram

Please see Lab 2 report for feedback.

15 – Report is updated
0 – Report is not updated – does not include the new required methods.

① -1 At least one of the required methods is not static.
② -1 Wrong names used for required methods
③ -1 Wrong type used for required methods (e.g. List<…> instead of Collection<…>)

Discussion

Changes since Lab2

5 - Clear description of "what and why" that addresses all critical changes
4 - Somewhat clear description that may not addresses all critical changes or does not provide clear reasons for the changes
3 - Description does not addresses significant changes or is difficult to understand
0 - Is not present in submission

① To make reading clearer, instead of listing all changes in a single flowing paragraph, break apart changes into separate paragraphs and provide structure to make it clear how the paragraphs relate to each other.
② This circle marks edits made to improve the flow of the text.

Benchmarking Results

5 – Nicely-formatted results with requested results
4 - Minor errors in formatting or data presented
3 - Little to no formatting – or data is missing required elements.
0 - Is not present in submission

① +0 Providing results for different numbers of desired dots beyond the three requested.
② Table formatting can be improved. E.g., put each result in a separate column, avoiding having text and numbers mixed in cells in the body of the table. Line up numbers so they are easier to compare, e.g. right-justify or justify on the decimal point so comparing orders of magnitude is easier.
③ Time formatting can be improved. Try to get few digits before a decimal point, and convert to hours and minutes where possible.
④ Including all results in one table would make it easier to compare the times.
⑤ -1 Table not captioned or not referenced from text. Referring to all tables and figure from text is an important habit to develop for all your college reports. It gives the reports a professional feel and makes reading the reports easier. One short paragraph that succinctly describes key changes and the reason for them is also sufficient. Also acceptable is a long bulleted list, with a bullet for each change.
⑥ -1 It appears that your times are in nanoseconds, even though they are labeled in milliseconds

⑦ Reverse caption and text – text should generally be longer than caption, and ideally tie in with the rest of the text in the report. (e.g. with the analysis of asymptotic running time.

Asymptotic Analysis

5 - Correct analysis presented in an easy to understand manner.
4 - Minor errors and/or unclear or weak explanation
3 - At least one significant error and/or no explanation of answers
0 - Is not present in submission

① Repeating something twice does not make the order $O(n^2)$. This requires nested loops or code that requires a pairing of one value of n with every other – e.g. If you have three colors of shirts, and two color of pants, it only takes $3+2=5$ steps to put them in the dresser, but it takes at least $3*2 = 6$ steps to consider each possible pair. So putting stuff in a dresser is $O(m+n)$, but choosing an outfit can be $O(mn)$.
② Typing $O(n^2)$ can be done by typing O (n <CTRL>-<SHIFT>-<PLUS> 2 <CTRL>-<SHIFT>-<PLUS>) where <A>-<B>-<C> indicates pressing the three keys at the same time.

Source code

getDesiredDots

① -3 Does not throw appropriate exceptions
② -1 Provide informative string to exception

getDesiredDotsItr

③ -2 Should use System.nanoTime(). System.currentTimeMillis() and new Date().getTime() give the time in milliseconds – up to 1,000,000 times less accurate!
④ -2 Doing a forced cast from a Collection to a List is dangerous – the Collection might not be a list. Furthermore, in this lab, the Collection is a reminder that we want to use iterators – not get(…) – to access the elements.
⑤ -2 Using get() in itr increases the Big-O running time by a factor of n.
⑥ -2 Does not compute critical value in the loop.
⑦ +1 Breaks removeLowestItr into sub-methods.
⑧ -2 Too complicated! Uses many very-poorly named variables in a way that would be extremely difficult to debug. To clean this up, see if you can start fresh with a totally different approach. Try to use as few variables as possible, and give them meaningful names.
⑨ -10 method is entirely non-functional