COMP20250 15% Java CA

Last Updated: November 2019

1 Key Submission Details

Deadline: Nov 29th 2019, 5pm (uploaded to Moodle). Code may be submitted up to an hour later.

Late Submissions Standard UCD policy on late submissions applies; see `https://www.ucd.ie/t4cms/latesub_po.pdf`. Your submission is deemed late if one or both deliverables are submitted late.

Plagiarism: The submission must be yours and yours alone. If you are unsure what is or is not plagiarism, the following is a none exhaustive list of example activities you cannot do:

- Copy the completed files of another student and submit them as your own
- Share copies, images or print outs of your code with another student (by e-mail, FB messenger, WhatsApp etc.)
- A group of students working on a single solution and then all submitting the same work or subset of the same work (regardless of whether variable, method, class names or ordering have been changed)
- Students collaborating at too detailed a level. For example, consulting each other after each line / block / segment of code and/or sharing the results.

For more details see:

- `https://www.cs.ucd.ie/sites/default/files/cs-plagiarism-policy_december2017.pdf`, and
- `http://www.ucd.ie/students/guide/academicregs.html`

Any submission suspected of plagiarism will be submitted to the School of Computer Science Plagiarism subcommittee for further investigation.

2 Task

For this assignment, you should answer the sample exam question present on Moodle three times such that you would expect:

1. the first solution receive a B grade or higher,
2. the second solution receive a C grade,
3. the third solution receive a D grade.

All parts of all three solutions **must**

- compile
- be free of arithmetic errors, intentional logical errors, and infinite loops
- be complete, i.e. an answer must be provided to parts a, b, c, and d. You cannot differentiate the quality of the solution based on part e.

To substantiate your claim on the quality of the solutions, prepare and execute ONE set of JUnit tests that evidences the rank order of your solutions. All three solutions should be tested with the same set of unit tests.

You can have three JUnit test .java files, if needed. However, all methods with a `@Test` annotation must be the same.

3 Deliverables

3.1 Report

Prepare a short report (max 5 pages – excluding cover sheet) that discusses the testing strategy of your JUnit tests specifically with respect to identifying better vs. worse solutions. You should discuss how each test provides an insight into the solution quality, and correspondingly why one solution is better than another. The discussion on solution quality **should only** be on the basis of JUnit test results.

3.2 Code

Provide a .java file of your JUnit tests. Each test should be appropriately commented to note:

1. which question part they are evaluating, and
2. what they are seeking to test (this can be a repeated version of detail in the report)

Provide a .java file for each of your solutions. Each solution should be commented to identify which part(s) of the code correspond to the part(s) of question. You should also include a comment to note whether the solution is to be considered the B, C, or D grade solution.

4 Grading Criteria

See next page.

Criteria	A+, A, A-	B+, B, B-	C+, C, C-	D+, D, D-	≤ E+
Testing Strategy (9/15)	A very thorough set of unit tests is implemented. They capture all aspects of all parts of the question. A large number of fringe cases, erroneous input, and valid input scenarios are captured as well as expected outputs and behaviours. In general, the presented unit tests enable a very established basis for comparing solutions.	A thorough set of unit tests is implemented. They capture most aspects of all parts of the question. A reasonable number of fringe cases, erroneous input, and valid input scenarios are captured as well as expected outputs and behaviours. In general, the presented unit tests enable a solid basis for comparing solutions. More tests would have enabled a more informed discussion.	A good attempt at unit testing all parts of the question is present. Some fringe cases, and erroneous input scenarios are captured as well as valid input scenarios expected outputs and behaviours. In general, the presented unit tests enable a reasonable basis for comparing solutions. A more detailed testing strategy would have enabled a deeper, more informed discussion.	Evidence of unit testing for most parts of all questions. Limited testing of fringe cases, erroneous input, and valid input scenarios are captured as well as expected outputs and behaviours. The presented tests were able to provide some detail to compare solutions, but somewhat superficially.	Testing is superficial or (mostly) not present. At least one question part (i.e. a, b, c, or d) has not been tested. One or more files don't compile or cannot be run due to errors or similar.
Discussion (4/15)	A well executed discussion of the testing strategy which accurately explains all relevant choices and/or decisions concerning the design of the JUnit tests to differentiate the quality of each solution.	The discussion accurately explains most key relevant choices and/or decisions concerning the design of the JUnit tests to differentiate the quality of each solution.	The discussion accurately explains a selection of relevant choices and/or decisions concerning the design of the JUnit tests to differentiate the quality of each solution. There are some JUnit test cases, however, that are either not addressed in the discussion or not adequately addressed to understand their inclusion and/or role.	The discussion is accurate, but lacks significant depth and/or detail.	Discussion seems arbitrary, largely inaccurate and/or is not well informed by the performed JUnit tests.
Report (2/15)	No spelling mistakes, all figures readable, page limit adhered to.	Minor typographical errors. All figures well-conceived. Page limit adhered to. No unnatural or informal uses of language, e.g. "I taught"	A small number of typographical errors. All figures well-conceived. Page limited exceeded marginally. No unnatural or informal uses of language, e.g. "I taught"	Report is mostly well-written, but has a few too many typos, spelling mistakes, poor layout choices, sub-optimal figures. No unnatural or informal uses of language, e.g. "I taught"	Report is sub-standard with respect to one or more of the following: Spelling, Grammar, Figures, Language.