

EECS2030 (Section F) Fall 2022

Guide to Written Test 1

WHEN: 16:20 – 16:50, Tuesday, September 27

CHEN-WEI WANG

Last Updated: September 22, 2022

- You **must** take the written test **in-person**: any remote attempt will be marked zero automatically.
- This written test is **strictly** individual: identified collaborations will be reported to Lassonde for a **breach of academic honesty**.
- You are given **30 minutes** to complete the submission. The time limit is **strict**.
- This written test accounts for 6% of your course grade.
- Unlike the labs (and the later programming tests), there will be **no** starter project for you to download and import.
- All questions will be answered on the Section F eClass site.
- You will be **solely** responsible for any **loss of time or marks** due to any of the following failing:
 - You have a working EECS account to login into a WSC lab machine.
 - You have a working PPY account to login into the eClass site (subject to Duo Mobile verification).

1 Rules

- Upon your arrival, please wait **outside** WSC 106/108.
 - The test will take place only in these two rooms.
 - Once the rooms are set up for the test, you will be allowed for entry (\approx 16:15).
- You may **only** bring to your seat:
 - Stationary (e.g., pen, pencil, eraser)
 - Sketch paper (blank on both sides).
You will be asked to return the sketch paper at the end of the test.
 - Water bottle
 - Mobile device (for Duo Mobile verification only)
During the test, always put the device face-down.
- All other personal belongings should be placed in front of the lab room.

- As soon as you are seated, login into a machine (using your EECS account), and then use a web browser (e.g., Firefox) to login into the Section F eClass site (using your PPY account).
 - First complete the quiz on *academic integrity* (≈ 1 minute).
 - The written test will be *opened* for submission at **16:20 EST**.
 - This is a **closed-book** test: use of any internet resources or notes is forbidden.
 - You are **forbidden** to use any programming IDE (e.g., Eclipse) during the test.
 - The written test will be *closed* for submission at **16:50 EST**.
- In principle, there will be **no** questions allowed during the test.
 - TAs will **not** answer questions.
 - If really necessary, the instructor will respond to your question, but you may just be advised to read the question(s) again more carefully.

2 Format

- Most ($\geq 70\%$) of the questions will be multiple-choice questions. For examples:
 - A true or false question
 - A question with a **single** correct answer
 - A question with **multiple** correct answers

e.g., Say you are given 5 answers for the question: 2 of them are correct (and 3 of them are incorrect). Accordingly, for each correct answer you choose will receive a credit of $\frac{100\%}{2} = 50\%$, whereas for each incorrect answer you choose will receive a penalty of $\frac{-100\%}{3} = -33.3\%$.

Say you chose one correct answer and one incorrect answer, then you would receive $50\% + (-33.3\%) = 16.7\%$ of the full marks. Also, the minimum mark you can receive is 0 (e.g., when you chose one correct answer and two incorrect answers).

This mechanism is to ensure that one cannot just receive full marks by simply choosing *all* answers.
- There might be written questions requiring you to, e.g.,:
 - Write a fragment of Java code
 - Explain how a given fragment of Java code works at runtime
 - Explain why a given fragment of Java code works

3 Coverage for the Test

- You need **not** study Lab1 for the test.
- The concepts about Github and terminal commands are **not** covered in the test.
- All topics covered in the review of OOP in Java:
 - TUTORIAL VIDEOS (PART 1 & PART 2) LINK
 - IPAD NOTES [PDF]
 - CLASSES AND OBJECTS (up to and including **Slide 79**) [PDF]
 - Written Notes:
 - * INFERRING CLASSES/METHODS FROM JUNIT TESTS [PDF]
 - * DECLARING AND MANIPULATING REFERENCE-TYPED, MULTI-VALUED ATTRIBUTES [PDF]

4 Study Tips for the Test

- The test is meant to **test your understanding** of the taught concepts (which is different from a programming test in which you are expected to write Java programs with no syntax or type errors).
- Go through the slides and annotated iPad notes to review the concepts and examples. Re-watch parts of the lecture/tutorial videos if necessary.
- Pay special attention to the logic explained on *tracing Java code* (e.g., use of a boolean variable to control the entrance into and exit from a loop, visualizing object creations and method calls).
- Given a piece of Java code, you are expected to judge:
 - If it does **not** compile, then what **syntax errors** or **type errors**?
 - If it compiles:
 - Will an **exception** occur (e.g., `IndexOutOfBoundsException`, `NullPointerException`)?
 - If no exception will occur at runtime, then what **console output** will it produce? Are there any **logical errors** (i.e., the output is not as expected)?

5 Example Test

- An example test will be made available on the Section F eClass site (under the **Written Tests** section) **by the end of Friday, September 23**. You can attempt this test for as many times as you wish.

This example test will be **closed** for submissions shortly before the actual test starts.
- It is important to note that:
 - These practice questions are meant for **familiarizing yourself with the format and workflow of the test** and **covering only some of the topics required by the actual test**: you are expected to study **all** materials as listed in Section 4.
 - The level of difficulty of the actual test may be **higher**.