EECS3342 (Section E) Fall 2024
Guide to Programming Test 2

WHEN: Wednesday, November 6 (during your <u>enrolled</u> lab session)	
WHERE: LAS 1002/1002B	
DURATION: 50 minutes	
LAB01/LAB02	LAB03/LAB04
13:30 PM to 14:20 PM	14:30 PM to 15:20 PM

CHEN-WEI WANG

1 Policies

– This programming test is <u>in-person</u> and **strictly** individual: plagiarism check may be performed and suspicious submissions will be reported to Lassonde for **a breach of academic honesty**.

– This programming test will account for **10%** of your course grade.

– This test is **purely** a programming test, assessing if you can construct contexts and machines in the `Rodin` tool, using **valid** `Rodin` syntax <u>**free of**</u> syntax, type, and logical errors.

– <u>**Structure of the Test**</u>:

   - At the start time on the test day, all LAS machines will be rebooted to the "lab-test mode" (where there is <u>**no**</u> network connection and you are expected to use the <u>**Rodin tool only**</u>).
   - During the test, you will be expected to:
      * Launch the Rodin tool on a designated <u>workspace</u>.
      * Create a new project with context(s) and machine(s) with the designated names.
      * Construct models (i.e., static contexts and dynamic machines) of the given problem, with the <u>**required**</u> constant/variable/event/parameter and axiom/invariant names.
        Caveat. It is absolutely critical for you to use the <u>**exact names**</u> (and <u>**not**</u> to introduce any extra constant/variable/event/parameter and axiom/invariant) as required by the test instructions; otherwise, it makes it difficult for
      * No interactions with the Rodin prover are expected in this test.

   - You are <u>**solely responsible**</u> for:
      * **leaving enough time (≈ 3 minutes) to export the completed Rodin project as an .zip archive file and upload/submit it to WebSubmit**; and
      * **submitting the right files for grading**.

– **<u>Submission for Grading</u>**:

- Like your labs, submission (of an archive `.zip` file) for this programming test must be through the WebSubmit link (which will be provided during the test).

- It is your sole responsibility for making sure that the correct version of the exported archive file is submitted. **After** clicking on the `submit` button on WebSubmit, you should **re-download** the archive file and make sure that it contains the right files to be graded. **<u>No</u>** excuses or submissions will be accepted after your attempt times out.

– **General Grading Criteria**

By importing your submitted archive file to Rodin:

- Part 1 of the test (formal specification of a context) will account for $\approx 60\%$ of the test marks; and the remaining $\approx 40\%$ of the test marks will be related to the manual proofs (via the Rodin prover).

- If the project contains any errors (e.g,. syntax errors, type errors), the TA will attempt to fix the errors for you, if they can be fixed quickly. If succeeded, there will be a *20% penalty* on the resulting mark.

- If the required constant/variable/event/parameter and axiom/invariant names you use are **<u>not</u>** exactly as instructed, the TA will attempt to fix this for you. If succeeded, then there will be a *10% penalty* on the allocated marks for that component.

- Unless otherwise specified in the instructions, it is expected that all associated proof obligations (POs) of your model (which satisfy the constraints as specified in the instructions) get discharged <u>automatically</u>.

2 Format

The format of this programming test will be <u>**identical**</u> to that of your <u>**Lab4**</u>: **1)** given informal problem descriptions, construct a Rodin model such that some associated POs get discharged automatically; and **2)** manually discharge the remaining POs by interacting with the prover.

– You will be given a **starter project** to import into the Rodin tool, and the project would have been set up for the proof tactics so you need **<u>not</u>** worry about setting it up yourself.

– As a reminder of the basic syntax, the following document will be made available to you during the test:

> `https://wiki.event-b.org/images/EventB-Summary.pdf`

You're advised to go over the above document prior to the test so that you can easily find what you need during the test.

– You will not need to do any setup for proof tactics.

– You will be expected to manipulate the Rodin tool (by creating models), see Lab1.

3 Coverage for the Test

– Lab4 (primarily)

– Reviewing the math review lecture, as well as the formal specification in Lab1 – Lab3, can be helpful.

4 Practice Questions

– Some practice questions have been made available on the ***Section E eClass site***. Solutions to the practice questions will be made available by the end of Monday, November 4.

– This practice test will **not** be graded, but you may practice submitting it.

– It is important to note that these questions are meant for familiarizing yourself with the **format** and **workflow** of the test, and they represent **only** as an example: you are expected to study **all** materials as listed in Section 3.

5 Simulating the Programming Test

It is highly recommended that you simulate taking the programming test by following these steps:

Preparation

– Login into a machine under remotelabs (using your EECS account): `https://remotelab.eecs.yorku.ca/`. Choose a machine under the `ea` category.

– Launch the <u>Firefox web browser</u> (under Activities) and login into the Section Z eClass site.

– Download and open the `PracticeTest1.pdf` file from eClass onto the Desktop.

Start the Test

– Start a timer (say for 60 minutes).

– Launch <u>Rodin</u> (under Activities)

– Tackle the test by by constructing the model(s) as instructed.

– **Before you submit, you should make sure that there is no error in any of the files.**

Submission

– **It is a recommended practice that you submit intermediate versions of your Rodin project (e.g., every 15 to 20 minutes).**

– Upload the exported `.zip` archive file of your completed Rodin project to the WebSubmit link for grading:

`https://webapp.eecs.yorku.ca/submit/?acadyear=2024-25&term=F&course=3342&`
`assignment=PT2`