1 Suggested Testing Strategy

In order to test your program, you may need to write a small program or script which YOU MUST NOT SUBMIT. The program at first reads the output file, and de-serialize the HashMap object. After it de-serialized the HashMap object, you need to check the PrivateStates objects which held in the HashMap (Note: you need to import your PrivateState classes). The output is not deterministic, however, some parts of it are deterministic. And, one can think on scenarios in the input file that result in deterministic outputs.
Following the parts of the output that you must check in each test.
Important note: Below are just mandatory checks for general tests. One can write other scenarios by which will require other stronger checks.

- Logging
    1. All the actors appear in the input file are in the HashMap.
    2. All the actions in the input file are in the logger of the respective actor.
- Department Private state:
    1. All the courses open in a department should appear in the courses list.
    2. All the students in a department should appear in the students list.
- Course and Student Private state:
    1. The number of registered student must not exceed the number of spaces in the course.
    2. The list of prequisites as in the json file.
    3. If a student appears in the course's registered students list, then must check that he has the prequisites and the course must appear in his grades sheet along with the grade given in the input file.

Following the strategy to test the extra actions.

- Register With Preferences: must check that the student is registered for one course at most from the list (you may assume he did not try to register to other courses in the preferences list again).
- Unregister: must check that the course does not appear in the student's grades sheet, and the student does not appear in the course's students list.
- Close course: must check that the course does not appear in any student grades sheet, and the course's students list is empty and the number of available spots is -1.

- Administrative Check: In general case, you must check the student gets one of the signatures of the corresponding computer (either Sig Fail or Sig Success). However, one can write an input file where this signature is deterministic (e.g. a test where the signature cannot be Sig Success), therefore, you still must do the check of success or fail.