HW 5: Automated Unit Test Generation

Authors: Samanta Suitsik C24393 samanta.suitsik@ut.ee,
Siim Tänavots C08145 siim.tanavots@ut.ee

Task 1-4 repository: https://github.com/SamantaSuitsik/Automated-Unit-Test-Generation2

Task 1

Equivalence classes:

input 1 = month
input 2 = day
input 3 = year

EC1: month = [1, 12], valid
EC2: day = [1, 31], valid
EC3: year = [1801, 2021], valid
EC4: output = “Invalid Next Year”, invalid
EC5: output = “invalid Input Date”, invalid
EC6: month = [MIN_INTEGER, 0], invalid
EC7: day = [MIN_INTEGER, 0], invalid
EC8: year = [2022, MAX_INTEGER], invalid
EC9: output in range of [“1/2/1801”, “12/31/2021”], valid

Comparison:
EC1, EC2, EC3, EC4, EC9 are covered by manually written test suite MTv1.
EC1, EC2, EC3, EC5, EC6, EC7, EC8, EC9 are covered by automatically generated test suite GTv1.
These test suites both test transition from one month to another for both 30 day and 31 day months. Overlapping EC-s are EC1, EC2, EC3, EC9.
The GTv1 tests invalid inputs like negative month, September with 31 days or just a invalid day like 290. It also just tests simple dates where only the day is changing and nothing else. GTv1 tests transition from February to March in a non leap year and also tests out if February can have 29 days if it’s not a leap year. GTv1’s all 21 tests pass with a line coverage of 88% and branch coverage of 93%.
MTv1 test suite tries to go over the year restriction with the input of 12/31/2021 that results in a failure. It also tests if a leap year’s February can have 29 days with the input of 2/28/2004. This test also fails. For this test suite, 2 tests out of 4 pass with a line coverage of 66% and branch coverage of 55%.

The automatically generated test suite GTv1 is not as effective in defect-detection in comparison to the manually created test suite. MTv1 has only the critical parts tested out (leap year and violating restrictions) contrary to the GTv1 which tested out all kinds of valid as well as invalid inputs, but missed out on niche inputs like the leap year. The code

coverage however is better in GTv1 thanks to the large amount of tests. But it doesn't matter much in case of effectiveness when it does not find any failures.

Task 2.2

GTv12 test suite covers EC1, EC2, EC3, EC4, EC5, EC6, EC7, EC8, EC9, so all the EC-s that GTv1 covers, except GTv12 also covers EC4.
The GTv12 has 23 tests with 100% line coverage and 98% branch coverage. All the tests pass. It does not find the leap year fault, like the MTv1 does.
In this suite, there are more tests to assess the month transitions in comparison to the GTv1. It also contains a test that verifies the correct year change and a test with the input of 12/31/2021 which violates the restriction and results in a “invalid next year” output. This test suite covers critical areas better and does not test invalid inputs so much as the GTv1 does. In conclusion, the GTv12 test suite is better than the GTv1.

Task 3

Execute tests MTv1 and GTv12. What do you observe? Is there anything surprising?

MTv1 testLeapYear ran successfully, unlike before uncommenting the leap year function.

Compare GTv2 to GTv12. What is different regarding test verdicts, number of tests, code coverage, and EC coverage?

Both test suites pass all of their tests and they both cover all EC-s. Test suite GTv2 has 2 more tests and has 6% more line coverage and 3% more branch coverage than test suite GTv12. GTv2 uses negative day and month inputs in some of its tests, meanwhile GTv12 doesn't go lower than zero. GTv2 doesn't have a test with zero as input. GTv2 seems to test leap year dates a bit more than GTv12. Since they’re both generated test suites, I'd say GTv2 is better, since it has higher coverage.

Task 4

First refactoring; execute tests MTv1 and GTv2. What do you observe? Is there anything surprising?

GTv2 has a failed test, because the test expects the run method to return “Invalid Input Date”, but it actually returns a date.

Second refactoring; execute tests MTv1 and GTv2. What do you observe? Is there anything surprising?

The tests of test suites MTv1 and GTv2 all pass, MTv1 has slightly smaller line coverage than before, because there is more code that it doesn’t test.

Task 5

I didn’t have any personal Java repositories to test on that were larger than the NextDate project, I couldn’t build the POS system I developed in the Software Engineering course so I couldn’t use TestSpark on it and I also couldn’t get TestSpark to work on any large public repositories that I found on Github, so I didn’t do this task.