Lab 2 Test Design Techniques (ISP)

Lab Objective:
This lab aims to help students learning the concepts and practicing the skills of test case design using **input space partitioning (ISP)**.

1. SUT Description:
Similar to Lab1, the project to be tested for this lab is Geo, an open source Java framework for geohashing. Please keep developing new test code in your Lab 1 repository which should already been created on GitLab, DO NOT create or fork another new repository in this lab.

2. Test Environment
The test environment will be IntelliJ IDEA. You should import the project which forked and add all the libraries imported by gradle. Also, you need to use JUnit4 as your unit testing framework.

3. Test Suite Generation
In this lab, you are required to design and implement unit tests for the following classes by using the Input Space Partitioning (ISP) technique to increase the statement and branch coverage of Lab1.

- Base32 (in the com.github.davidmoten.geo): Has 5 methods
- Coverage (in the com.github.davidmoten.geo): Has 5 methods
- CoverageLongs (in the com.github.davidmoten.geo): Has 5 methods
- GeoHash (in the com.github.davidmoten.geo): Has 23 methods
- Geomem (In com.github.davidmoten.geo.mem): Has 6 methods
- Info (In com.github.davidmoten.geo.mem): Has 6 methods

Take a minute going through the API specifications of each class described above in Javadoc.

4. Test Plan
To begin with, a plan must be created. You must document the test plan in your lab report. The plan must include a brief description about the requirements of the test (such as the goal of statement or branch coverage), the strategy and activities you plan to perform to meet the requirements, **the approach to achieve the goal (such as using Input Space Partitioning and and All Combination Coverage)**, and **the success criteria** of completing the test (such as satisfying the coverage goal). (one page would be sufficient for the test plan)

Carry out your test plan. To keep your workload manageable, you are required to create test cases for **15** out of 50 methods (38 methods of com.github.davidmoten.geo and 12 methods of com.github.davidmoten.geo.mem) at least. Choose **15** methods and create test cases for them. Some of the chosen methods can be the same as Lab 1. Try to keep each test case in a separate testing method if possible.

Push your code to GitLab after a part of tests is done. By doing so, you can see the incremental change of testing coverage from CI status.
You must push your code to GitLab and ensure that the CI is running successfully for at least three or more times.

In this lab, the **statement coverage** must be achieved at least 70%.

5. Lab Report
The Lab report must include (but not limit to) the following sections:

- Test Plan: briefly describe test requirements, planed test activities, **approach**, and success criteria for the test.

- Test Design : for the method under test (MUT), using input space partitioning (ISP). In addition to test inputs and expected outputs , you have to provide the details of the test case design using the templates. Note that for using ISP, you need to provide the characteristics and corresponding partitions for each MUT. A test case design template for ISP can be downloaded here.

- Test Result: for each method under test, provide the test results (in terms of Pass/Fail or screenshots for each test case associated with the method under test.

- Test Coverage: provide statement code coverage (or branch coverage) for each method under test and the overall statement coverage (or branch coverage) for the test.

- The Coverage Comparison of your Lab1 and Lab2, including statement coverage (or branch coverage) analysis.

- The screenshots of `CI/CD -> Pipelines` and 3 or more of you different CI status.

You can reference the sample of Lab report submission [here](#)

Please convert your Lab report to `.pdf` file, `.docx` file will not be accepted.
The Lab report and ISP test case design document must be put in the corresponding directory in your project
(e.g. `GeoProject -> LabReport -> Lab2`).
Make sure you have pushed your code and Lab report on the [GitLab](#)successfully.

Finally, you must open an new **issue** on [TA's GeoProject](#) with the following information and format to notify TA
that you have done your lab.

Please submit project and the report to gitlab, and email to TA your project link. **TA would check your homework on gitlab.**