

Assignment 1: Unit Testing

Deadline: **March 31, 2022, 23:55**

The goal of the first series of exercises is to introduce the topic testing from the viewpoint of a developer and to gain experience with tools for unit testing.

Assignment: Writing unit tests for the RingBuffer

The class under test is *RingBuffer* (adopted from Introduction to Programming in Java, R. Sedgewick and K. Wayne, Addison-Wesley, 2008). The code is provided in the zip archive *RingBufferExample.zip*. Familiarize yourself with the class *RingBuffer* and write appropriate unit tests.

Requirements and Setup

- Java JDK 11 or higher: <https://openjdk.java.net/>
- Apache Maven: <https://maven.apache.org/>
- JUnit 5: <http://www.junit.org> (Note: Dependencies are handled by Maven.)
- Setup steps:
 1. Download the source code package from moodle. It contains the code for the class *RingBuffer*, a corresponding test class *RingBufferTest*, and a *pom.xml* for building the project and running the tests with Maven.
 2. When you run Maven¹ for the first time, all required plugins and dependencies will be downloaded. Usage: *mvn test* - note that the existing test will fail!

Instructions

- Implement as many test cases as you consider useful but provide at minimum 5 different test cases (test methods in the class *RingBufferTest*), which test different aspects of the class under test.
- Important: Write meaningful test cases that cover different functionalities of the *RingBuffer*. Focus your effort on writing “useful” test cases and “clean” unit test code following best practices². Be prepared to discuss your thoughts and ideas when presenting the results.
- Include test cases for situations where an exception is expected to be thrown. Test that the expected exception is actually thrown³.
- Note: The class under test is an early version sample code from the book mentioned above. If you find a bug, write a failing unit test that shows the identified problem.

Submission & Presentation

Hand in the source code of your tests and any related files you created as zip archive and **upload it to the Moodle course platform before the deadline**.

Work in teams. Only one upload per team is required, i.e., one team member uploads the files to Moodle.

The work has to be **presented in the lecture on Thr, 5. May 2022**. Each team has to be prepared to present their results. The teams that present will be selected at the beginning of the lecture. No slides (ppt etc.) have to be prepared, just bring a laptop with a development environment including all the tools you used and the source code of your solution.

¹ <https://maven.apache.org/guides/getting-started/maven-in-five-minutes.html>

² E.g., see sample chapter from the book *Tahchiev P. et al.: Junit in Action (2nd ed.)*, Manning Pub., 2010

³ <https://www.baeldung.com/junit-assert-exception>