

Task 1: Perform Code Review

Create a new branch based on the "Blackbox" branch (not based off of master) called "Review":

<https://github.com/scottthomas21/ser316-spring2021-B-sthoma58/tree/Review>

On markdown file perform code review logging defects:

<https://github.com/scottthomas21/ser316-spring2021-B-sthoma58/blob/Review/CodeReview.md>

Commit to review branch:

<https://github.com/scottthomas21/ser316-spring2021-B-sthoma58/commit/38f767399c839a4a2cc6e00ae0ac578224fa2dfe>

Task 2: Fixing the defects

Create issues on GitHub for three defects:

- 1) <https://github.com/scottthomas21/ser316-spring2021-B-sthoma58/issues/1>
- 2) <https://github.com/scottthomas21/ser316-spring2021-B-sthoma58/issues/2>
- 3) <https://github.com/scottthomas21/ser316-spring2021-B-sthoma58/issues/4>

Resolve Defects:

- 1) <https://github.com/scottthomas21/ser316-spring2021-B-sthoma58/commit/2184bb34e9214076d7cd8d2648a7c23633944991>
- 2) <https://github.com/scottthomas21/ser316-spring2021-B-sthoma58/commit/8a0d5a947fb283ef3fd1fb204acca242a39a925>
- 3) <https://github.com/scottthomas21/ser316-spring2021-B-sthoma58/commit/73de2d7da8e87822b484ca552f20e13779f51c98>

Task 3: White-box testing (with some black box)

Step 1: Implement calculateDamage(...) in BattleScenario.java

Create a new branch, base this branch off of your "Blackbox" branch and name it "White- box":

<https://github.com/scottthomas21/ser316-spring2021-B-sthoma58/tree/WhiteBox>

Implement calculateDamage(...) in BattleScenario.java:

<https://github.com/scottthomas21/ser316-spring2021-B-sthoma58/commit/3819033d2d34cf790d928f15ecf66be617aaafde>

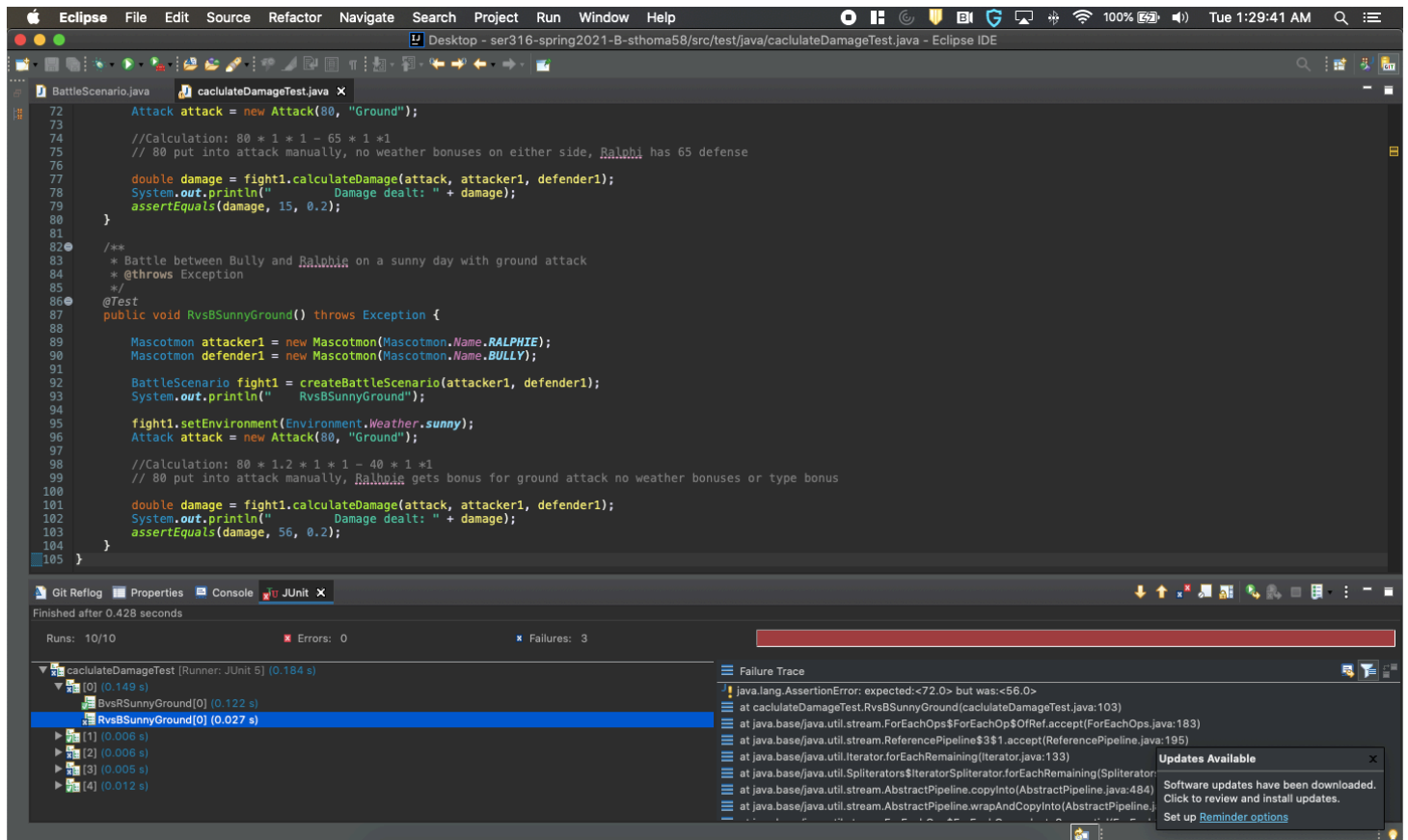
Create a new test class and call it caclulateDamageTest.java:

<https://github.com/scottthomas21/ser316-spring2021-B-sthoma58/commit/0f2414acf97f52022b4c206a1814d14f8a8c9e30>

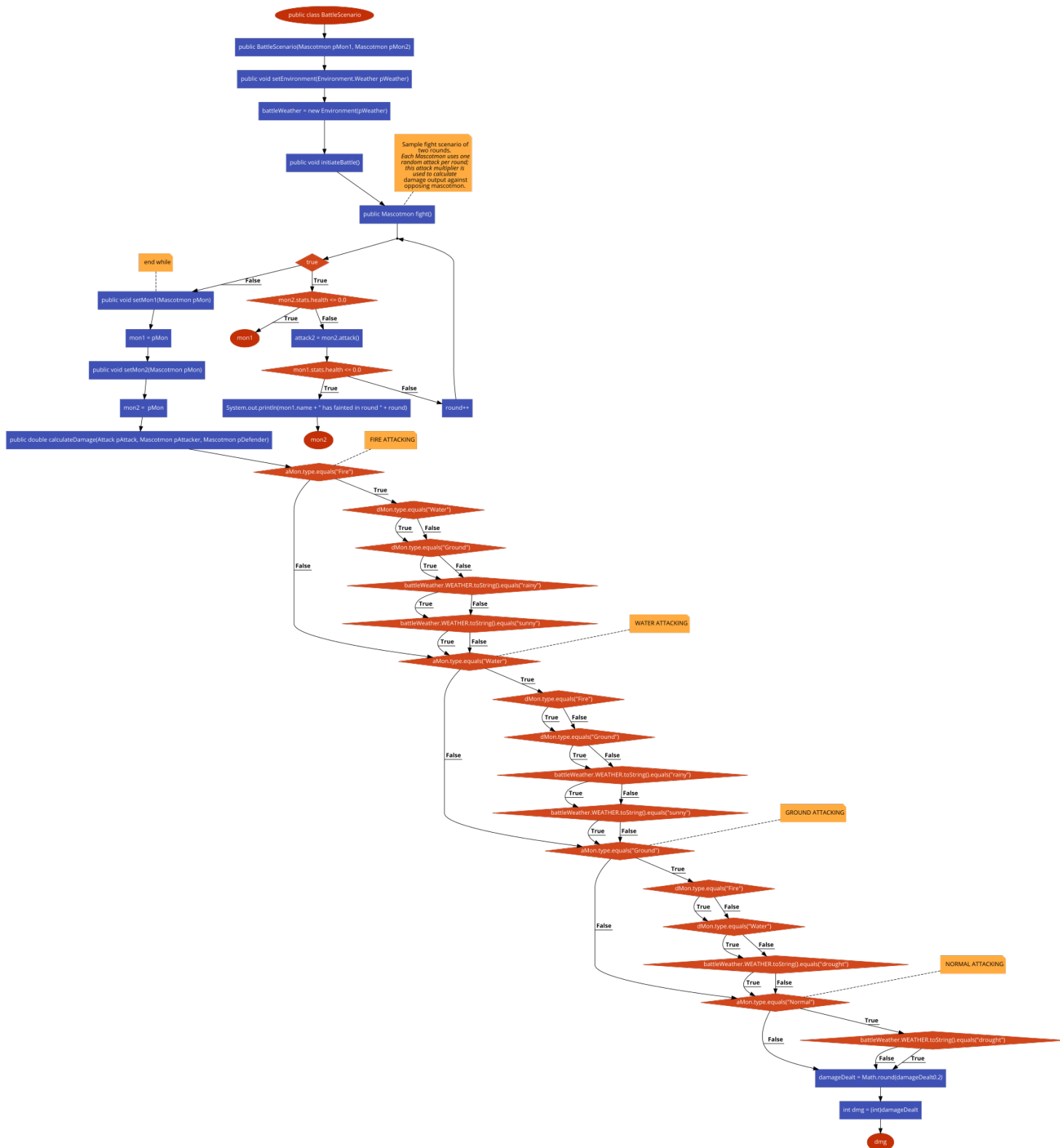
Delete the GivenBlackBox.java test:

<https://github.com/scottthomas21/ser316-spring2021-B-sthoma58/commit/bd0bef3cb5adc76c023e5bd25b6448a35f6d8708#diff-bf623a7d4ecfb23dcf66ef6bcc08476a7aa847c6c3385674bf143c2f53e6b634>

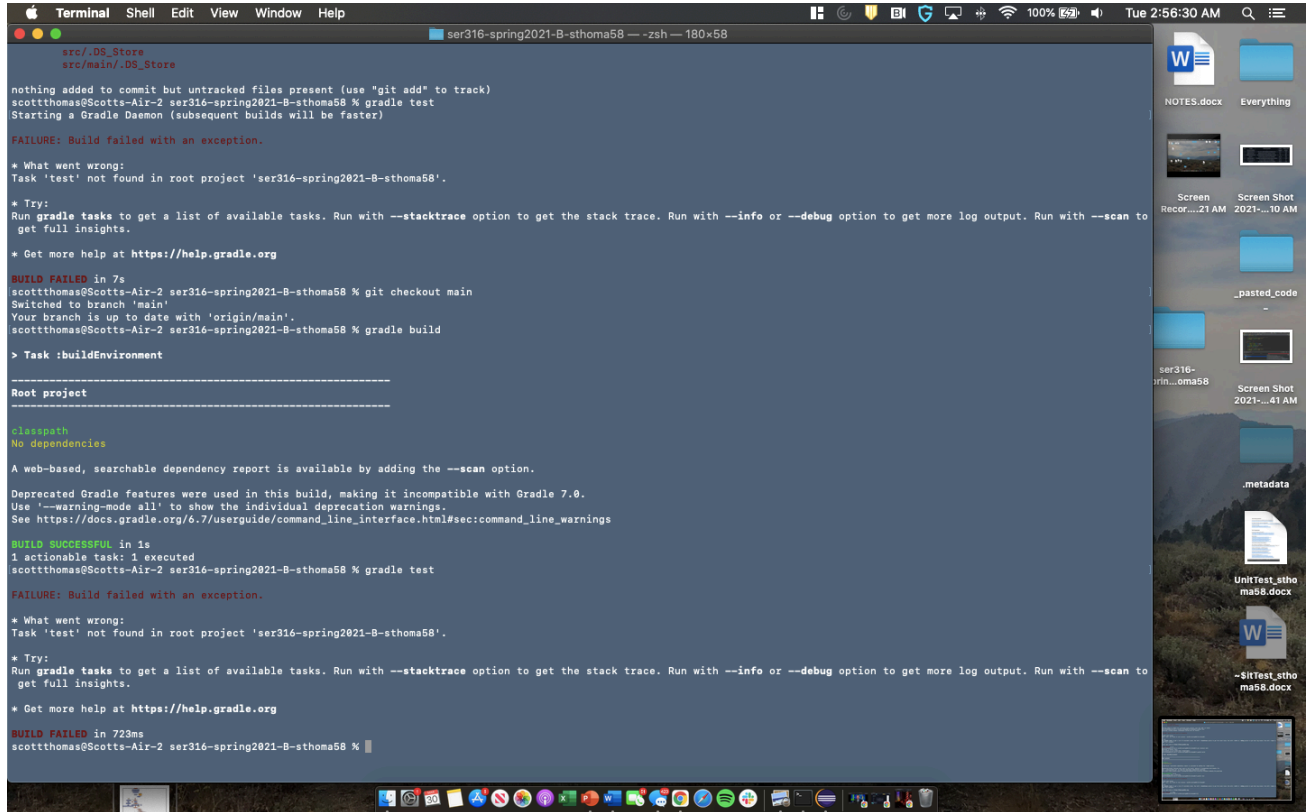
Screenshot of passing tests:



Draw a control flow graph for the fight() method (before any changes are made):



Task 4: Gradle - 5 Points



```
src/.DS_Store
src/main/.DS_Store

nothing added to commit but untracked files present (use "git add" to track)
scottthomas@Scotts-Air-2 ser316-spring2021-B-sthoma58 % gradle test
Starting a Gradle Daemon (subsequent builds will be faster)

FAILURE: Build failed with an exception.

* What went wrong:
Task 'test' not found in root project 'ser316-spring2021-B-sthoma58'.

* Try:
Run gradle tasks to get a list of available tasks. Run with --stacktrace option to get the stack trace. Run with --info or --debug option to get more log output. Run with --scan to get full insights.

* Get more help at https://help.gradle.org

BUILD FAILED in 7s
scottthomas@Scotts-Air-2 ser316-spring2021-B-sthoma58 % git checkout main
Switched to branch 'main'
Your branch is up to date with 'origin/main'.
scottthomas@Scotts-Air-2 ser316-spring2021-B-sthoma58 % gradle build

> Task :buildEnvironment

Root project
-----
classpath
No dependencies

A web-based, searchable dependency report is available by adding the --scan option.

Deprecated Gradle features were used in this build, making it incompatible with Gradle 7.0.
Use --warning-mode all to show the individual deprecation warnings.
See https://docs.gradle.org/6.7/userguide/command_line_interface.html#sec:command_line_warnings

BUILD SUCCESSFUL in 1s
1 actionable task: 1 executed
scottthomas@Scotts-Air-2 ser316-spring2021-B-sthoma58 % gradle test

FAILURE: Build failed with an exception.

* What went wrong:
Task 'test' not found in root project 'ser316-spring2021-B-sthoma58'.

* Try:
Run gradle tasks to get a list of available tasks. Run with --stacktrace option to get the stack trace. Run with --info or --debug option to get more log output. Run with --scan to get full insights.

* Get more help at https://help.gradle.org

BUILD FAILED in 723ms
scottthomas@Scotts-Air-2 ser316-spring2021-B-sthoma58 %
```