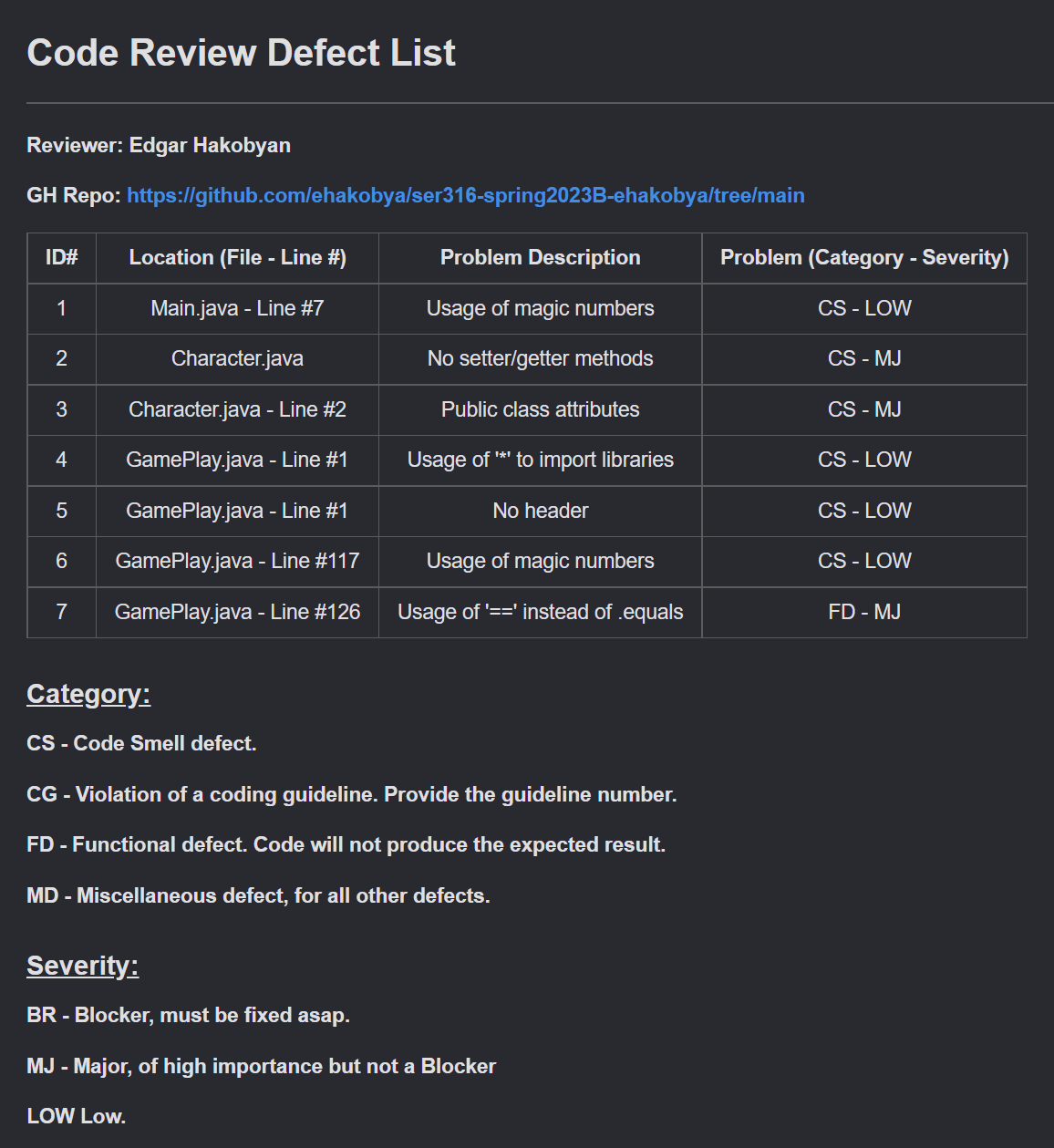
Edgar Hakobyan

SER 316 : Assignment 3

Date: 04/03/2023

**Task 1: Code Review**



**Task 2: Fixing the defects**

GitHub issues

<https://github.com/ehakobya/ser316-spring2023B-ehakobya/issues/6>

<https://github.com/ehakobya/ser316-spring2023B-ehakobya/issues/7>

<https://github.com/ehakobya/ser316-spring2023B-ehakobya/issues/8>

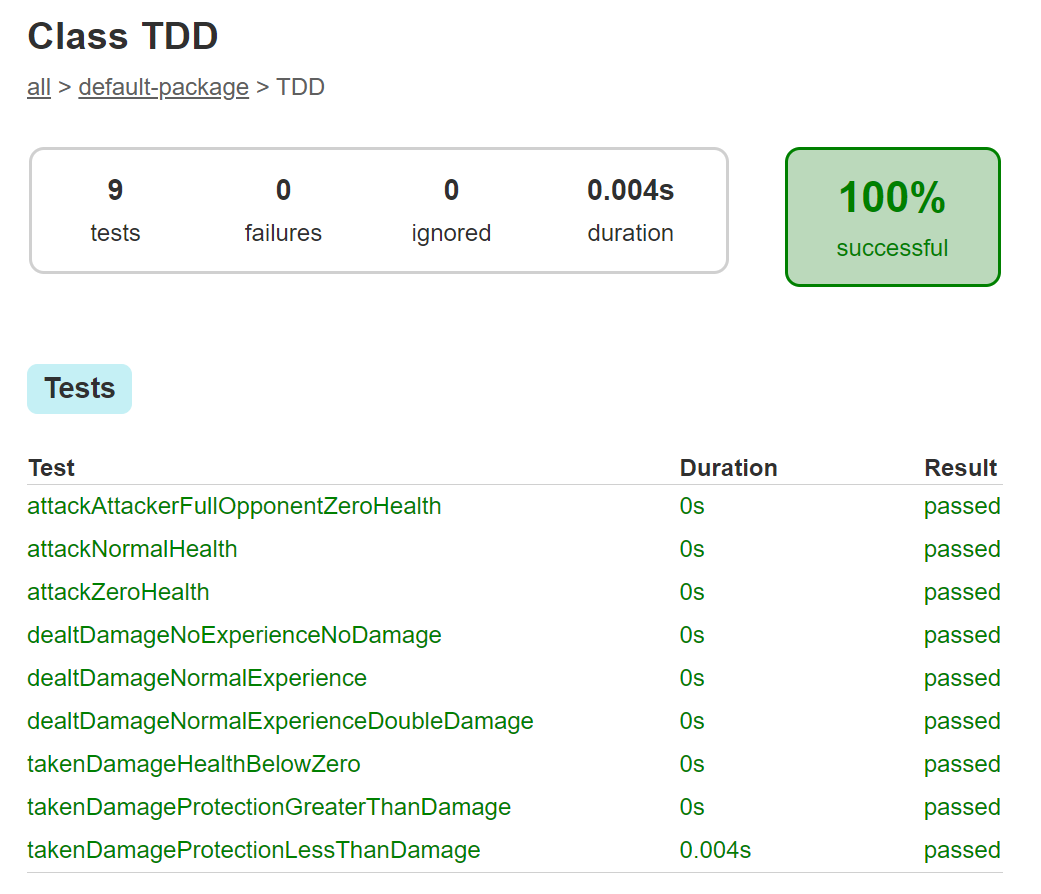
Issue fixes

<https://github.com/ehakobya/ser316-spring2023B-ehakobya/commit/2e41a47dbd363a8e40c0d29eee2233ba2b20b9e9>

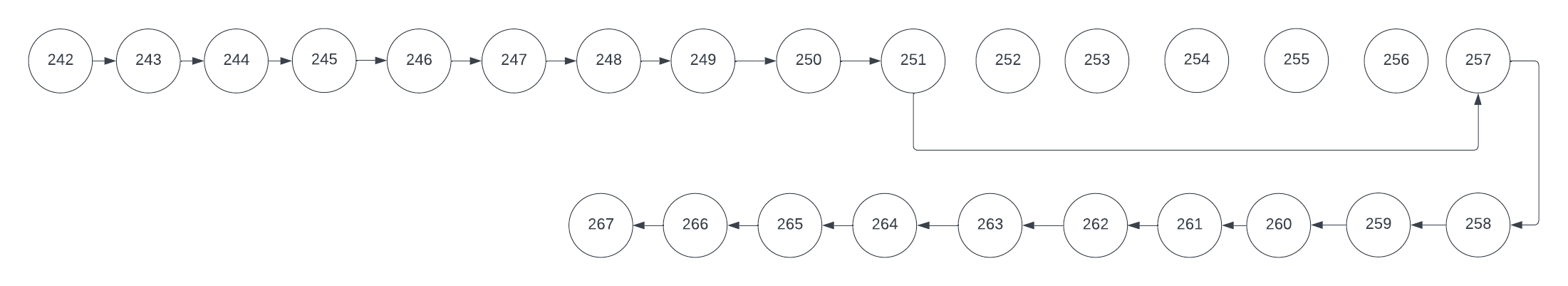
<https://github.com/ehakobya/ser316-spring2023B-ehakobya/commit/811c2b253de9e3b41d3ae4a4e532f1de39806bae>

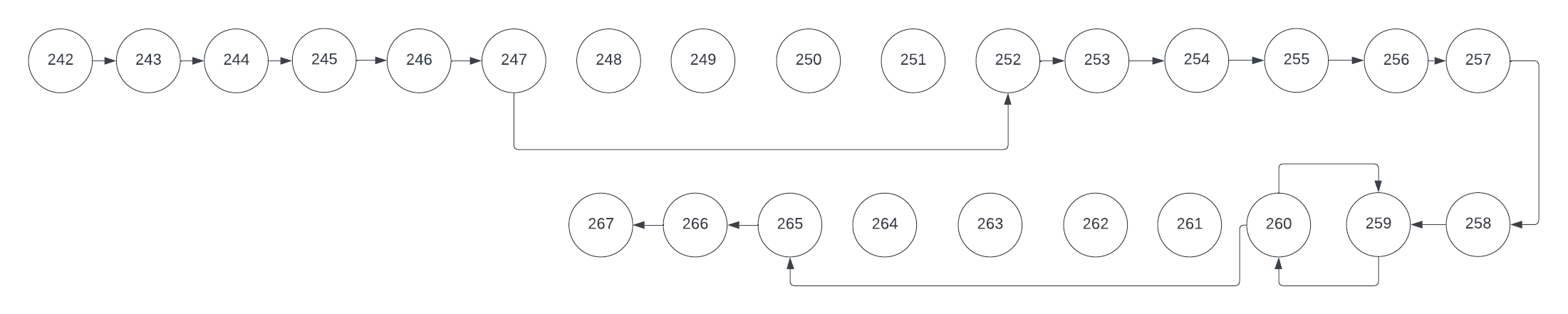
<https://github.com/ehakobya/ser316-spring2023B-ehakobya/commit/bb64d1ab40da80adcafb9cd54939e81f5d9e7c06>

Task 3: Whitebox testing



Control flow graphs





{ < 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267 > } – playAttackerSpeedGreaterThanOpponent()

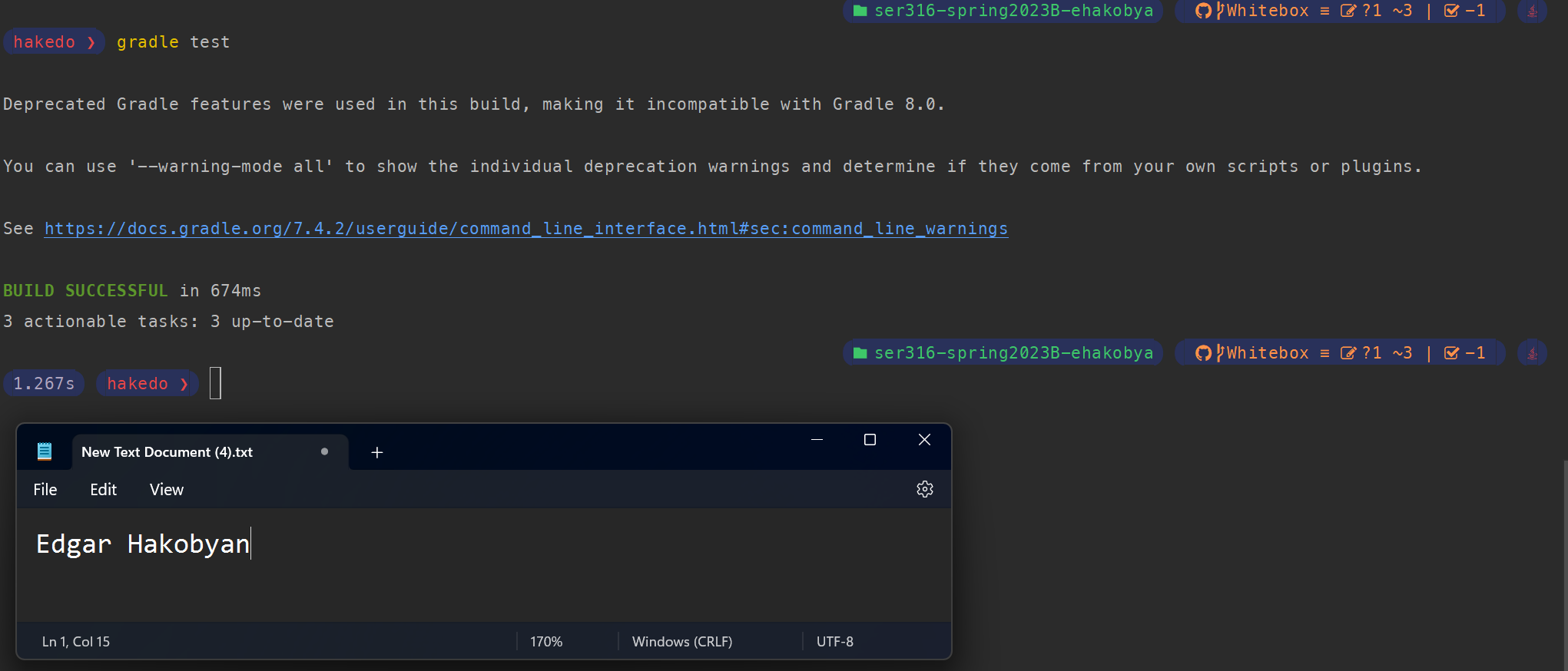
{ < 242, 243, 244, 245, 246, 247, 248, 249, 250, 251, 252, 253, 254, 255, 256, 257, 258, 259, 260, 261, 262, 263, 264, 265, 266, 267 > } – playAttackerSpeedLessThanOpponent()

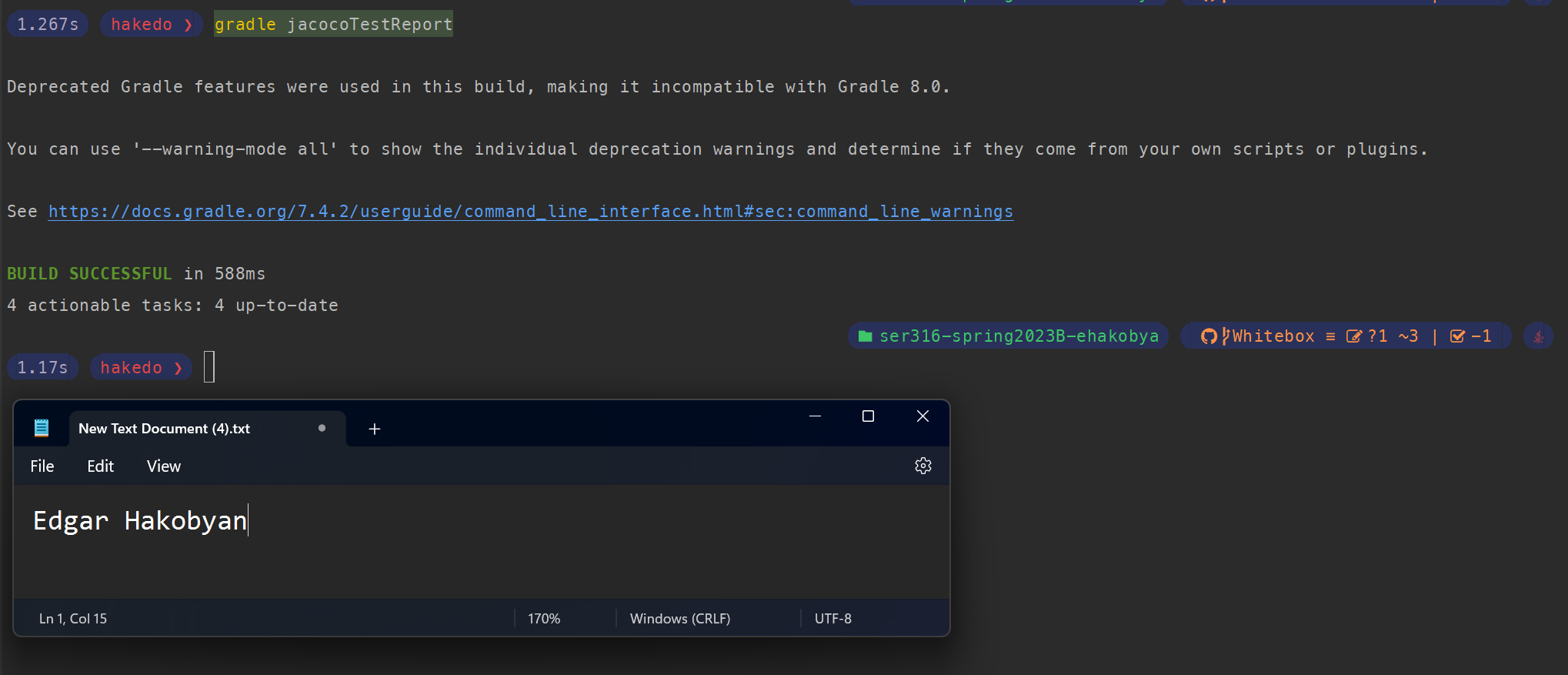
Explanation:

The test cases mentioned above were used to achieve 92% code coverage for the GamePlay class. The methods were tested to ensure they functioned correctly to some extent. The attack(), dealDamage(), and takeDamage() methods were tested using newly added implementations in the TDD class through blackbox testing. All test cases passed with the correct implementations.

Two test cases, playAttackerSpeedGreaterThanOpponent () and playAttackerSpeedLessThanOpponent (), achieved complete node and edge coverage, confirming the expected functionality of the attack(), dealDamage(), and takeDamage() methods. Additional test cases were added to ensure the play() method awarded bonus experience to the character with greater speed as specified. The remaining methods for adding/removing character opponents, as well as levelup(), were also tested with their respective test cases and found to be working correctly.

**Task 4: Gradle**





Test Coverage Results

1. Code coverage not including main is 87%
2. 92% for GamePlay

