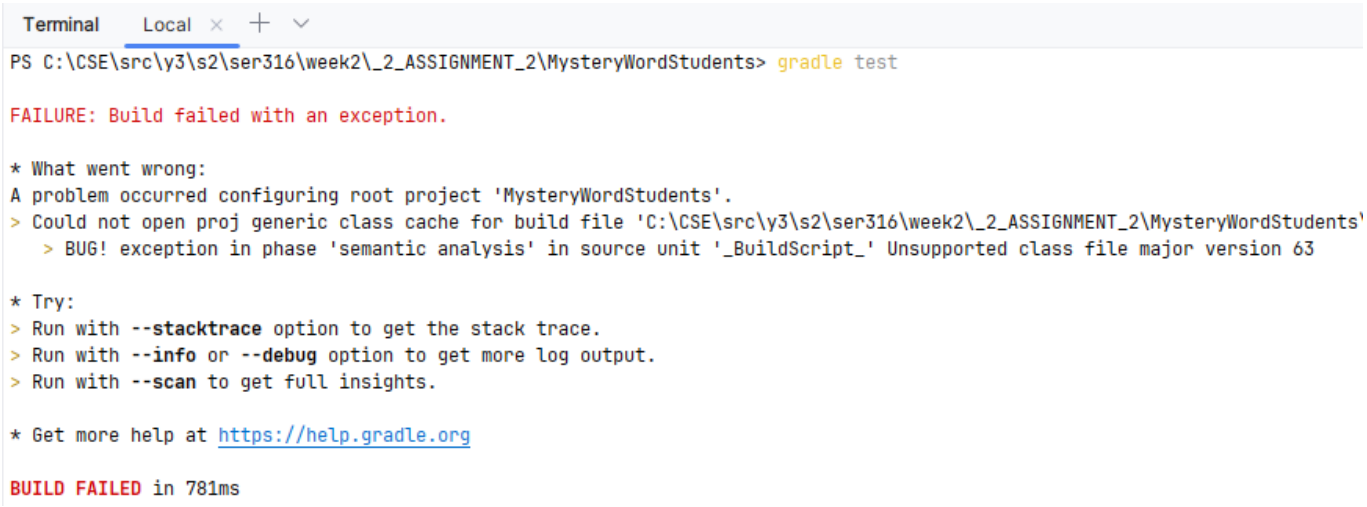
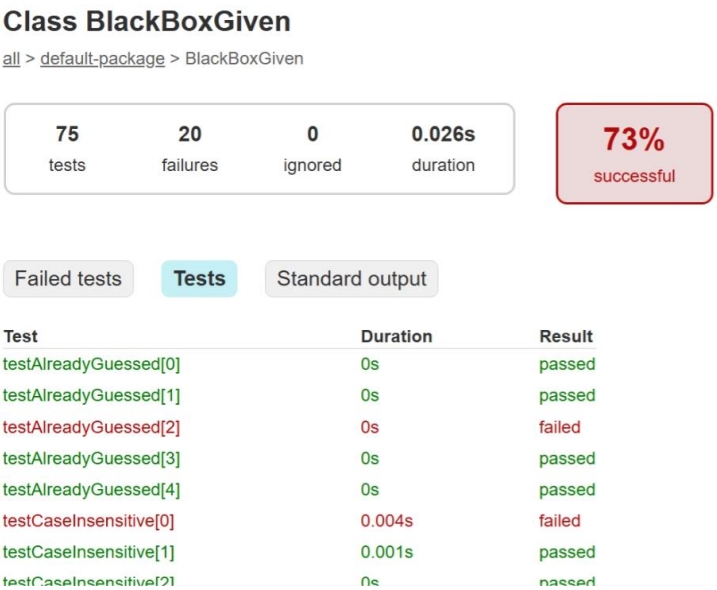
Assignment 2 - Unit Tests

Name: Eyad Mohamed AbdelMohsen Ghanem
ASU ID: eghanem
Course: SER 316 Summer 2025
Date: May 30, 2025

Task 1: Getting Started (10 points)

Screenshot of Test Execution





Task 2: Code Review Issues (10 points)

Issue 1: Constructor Parameter Ignored

Problem: In Game(String fixedWord, String name) constructor, the name parameter is ignored and hardcoded to "Anna" **Impact:** Player name will always be incorrect when using this constructor

Issue 2: Inconsistent Point Initialization

Problem: Different constructors set different initial points (5 vs 10) **Impact:** Game difficulty varies unpredictably based on constructor used

Issue 3: Missing Method Implementation

Problem: makeGuess(String guess) method returns hardcoded 0.0 instead of implementing game logic **Impact:** Game is completely non-functional

Issue 4: Incorrect Return Value in countCorrectLetters

Problem: Method always returns 0 despite calculating correct letters **Impact:** Method contract is violated

Issue 5: Poor Access Modifier Consistency

Problem: Inconsistent use of public, protected, and private modifiers on class members **Impact:** Breaks encapsulation principles

---

Task 3: Black-Box Testing (40 points)

Equivalence Partitioning

Input Domain 1: Single Letter Guesses

Valid Classes:

- Letters present in the word (a-z, A-Z)
- Letters not present in the word (a-z, A-Z)
- Letters appearing multiple times in the word

Invalid Classes:

- Non-alphabetic characters (0-9, symbols)
- Empty string
- Previously guessed letters

Input Domain 2: Word Guesses

Valid Classes:

- Correct word (exact match, case insensitive)
- Incorrect word with correct length
- Incorrect word that is too short/long
- Partial matches (substring of correct word)

Invalid Classes:

- Words containing non-alphabetic characters
- Previously guessed words
- Empty string

Input Domain 3: Game State

Valid Classes:

- Active game (< 10 guesses made)
- Game at guess limit (exactly 10 guesses)

Invalid Classes:

- Game over (≥ 10 guesses made)
- Game already won

Boundary Value Analysis

Letter Boundaries:

- First/last letters of alphabet: 'a', 'z', 'A', 'Z'
- Case boundaries: Same letter in upper/lower case

Word Length Boundaries:

- Correct length: answer.length()
- One character shorter: answer.length() - 1
- One character longer: answer.length() + 1

Game State Boundaries:

- 9th guess (one before limit)
- 10th guess (at limit)
- 11th guess (over limit)

Test Case Design

Test Case	Description	Input	Expected Return	Expected Points	Expected Status
TC01	Correct word guess	"lion"	0.0	+4 points	status=1
TC02	Letter in word once	"l"	1.1	+1 point	status=0
TC03	Letter in word twice	"o" (in "book")	1.2	+2 points	status=0
TC04	Letter not in word	"z"	1.0	no change	status=0
TC05	Wrong word, correct length	"bear" (for "lion")	2.0	+1 point	status=0
TC06	Word too short	"cat" (for "elephant")	2.2	-5 points	status=0
TC07	Word too long	"elephant" (for "cat")	2.1	-5 points	status=0
TC08	Partial word match	"li" (for "lion")	3.0	+2 points	status=0
TC09	Already guessed	"l" twice	4.0	-2 points	status=0
TC10	Invalid characters	"l1on"	4.1	-3 points	status=0
TC11	Game over (10 guesses)	10 wrong guesses	5.0	no change	status=2
TC12	Guess after game over	11th guess	5.1	no change	status=2
TC13	Guess after winning	guess after win	5.1	no change	status=1

Test Results Analysis

Game0 Results

Failed Tests: TC01, TC05, TC06, TC07
Issues Found:

- Returns correct status but wrong points for winning (got 10, expected 14)
- Word length penalties calculated incorrectly

Game1 Results

Failed Tests: TC03, TC10, TC11
Issues Found:

- Point calculation appears doubled for letters (got 1.4, expected 1.2)
- Missing input validation for invalid characters

Game2 Results

Failed Tests: None
Issues Found: No bugs detected - appears to be correct implementation

Game3 Results

Failed Tests: TC10, TC11, TC12
Issues Found:

- Doesn't validate input for invalid characters
- Game state management issues after 10 guesses

Game4 Results

Failed Tests: TC01, TC13
Issues Found:

- Doesn't set game status to won (got status=0, expected status=1)
- Allows continued guessing after winning

Detailed Bug Analysis

Game0:

- **Bug:** Incorrect word scoring - awards 10 points instead of word length
- **Found by:** TC01 (winning test)

Game1:

- **Bug:** Point doubling for letter occurrences

- **Found by:** TC03 (double letter test)

Game2:

- **Bug:** None found - best implementation

Game3:

- **Bug:** Missing input validation for invalid characters
- **Found by:** TC10 (invalid character test)

Game4:

- **Bug:** Doesn't update game status when winning
- **Found by:** TC01 (winning test)

Assignment Questions

Which Game implementation adheres to the specification best?
Game2 passed all test cases and appears to correctly implement the specification.

List errors found and which test cases helped find them:

- Game0 incorrect scoring: TC01 showed wrong points for winning
- Game1 point doubling: TC03 revealed doubled letter points
- Game3 missing validation: TC10 exposed invalid input acceptance
- Game4 status bug: TC01 showed missing win status update