

Comprehending Test Code: An Empirical Study

Chak Shun Yu, Christoph Treude, Maurício Aniche



Program Comprehension

Impact of Code Lexicon

Hofmeister et al., Schankin et al., Fakhoury et al., Kosti et al.

Impact of Code Constructs

Jbara and Feitelson, Melo et al., Crosby et al., Hegarty-Kelly et al.

Impact of Code Smells

Palomba et al., Soh et al.

Using Code Visualization to Enhance Program Comprehension

Bacher et al., Mumtaz et al., Cornelissen et al.

Similarities with Natural Language Comprehension

Busjahn et al., Peitek et al.

Program Comprehension

Impact of Code Lexicon

Hofmeister et al., Schankin et al., Fakhoury et al., Kosti et al.

Impact of Code Constructs

Jbara and Feitelson, Melo et al., Crosby et al., Hegarty-Kelly et al.

Impact of Code Smells

Palomba et al., Soh et al.

Using Code Visualization to Enhance Program Comprehension

Bacher et al., Mumtaz et al., Cornelissen et al.

Similarities with Natural Language Comprehension

Busjahn et al., Peitek et al.

Test Code Comprehension

?

Better Testing Requires Understanding How Developers Comprehend Tests

Better Testing Requires Understanding How Developers Comprehend Tests

RQ1 What factors influence the time that developers spend **reading** test code?

RQ2 What factors influence the ability of developers to identify the **purpose** of a test suite?

RQ3 What factors influence the ability of developers to produce additional test cases to **extend** a test suite?

Method: Study developers comprehending test code



44 participants



132 data points

Method: Study developers comprehending test code



44 participants

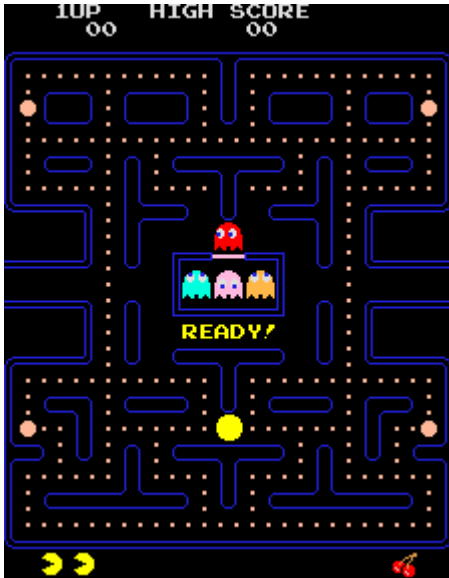
10 participants with prior
knowledge of project

9 female participants



132 data points

Test Code

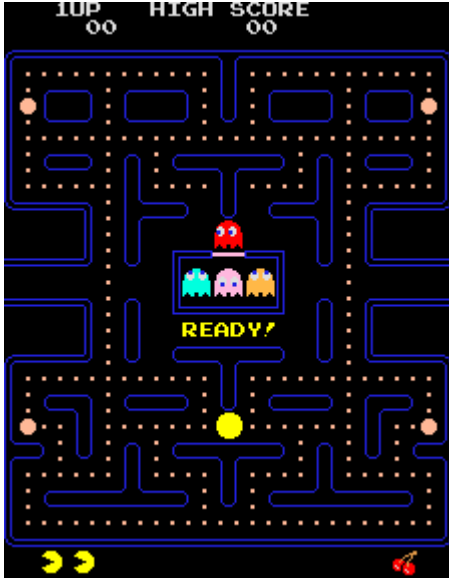


JPacman: Pacman game in Java

Educational software project at TU Delft
to teach testing principles

<https://github.com/SERG-Delft/jpacman-framework>

Test Code



Understandable in an isolated and standalone environment

Well-known domain

Adheres to the AAA testing structure in an unambiguous manner

<https://github.com/SERG-Delft/jpacman-framework>

```
49  /**
50   * R3: A player bumps into a pellet.
51   * The player should eat it.
52   */
53  @Test
54  public void testPlayerPellet() {
55      final int pointsEaten = 42;
56      when(pellet.getValue()).thenReturn(pointsEaten);
57
58      getCmap().collide(player, pellet);
59
60      verify(player).addPoints(pointsEaten);
61      verify(pellet).leaveSquare();
62      verifyZeroInteractions(ghost);
63  }
```

...

Arrange

Act

Assert

**Test
(Case)**

**Test
Suite**

Pre-Study Questionnaire

Age

Gender

Experience

as a developer

with the Java programming language

with using tests

```
4. private Ghost ghost = mock(Ghost.class);  
5. private Pellet pellet = mock(Pellet.class);  
6.  
7. @Test  
8. public void testCase1_1() {
```

Design similar to Hofmeister et al.



```
4. private Ghost ghost = mock(Ghost.class);  
5. private Pellet pellet = mock(Pellet.class);  
6.  
7. @Test  
8. public void testCase1_1() {
```

What is the purpose of the test suite?

Describe the additional test cases that you would write to extend the current test suite.



Variables

Independent Variables

Participant Age

Participant Gender

Participant Experience

- developer

- Java

- test code

Prior Knowledge of Project

Participant UUID (random factor)

Trial Number (random factor)

Variables

Independent Variables

Participant Age

Participant Gender

Participant Experience

- developer

- Java

- test code

Prior Knowledge of Project

Participant UUID (random factor)

Trial Number (random factor)

Dependent Variables

Reading Time

% Arrange

% Act

% Assert

Identification of Testing Purpose

Producing Additional Test Cases

- Basic tests

- Domain tests

- Error tests

- Other tests

Variables

Extension of the provided cases in a limited manner based on the information in the provided cases.

Dependent Variables

Reading Time

% Arrange

% Act

% Assert

Identification of Testing Purpose

Producing Additional Test Cases

- Basic tests
- Domain tests
- Error tests
- Other tests

Variables

Extension of the provided cases in a limited manner based on the information in the provided cases

Extending cases to test valid scenarios and input, but not limited to examples given in the provided cases

Dependent Variables

Reading Time

% Arrange

% Act

% Assert

Identification of Testing Purpose

Producing Additional Test Cases

- Basic tests

- Domain tests

- Error tests

- Other tests

Variables

Extension of the provided cases in a limited manner based on the information in the provided cases

Extending cases to test valid scenarios and input, but not limited to examples given in the provided cases

Additional cases with invalid input or dedicated to fail

Dependent Variables

Reading Time

% Arrange

% Act

% Assert

Identification of Testing Purpose

Producing Additional Test Cases

- Basic tests

- Domain tests

- Error tests

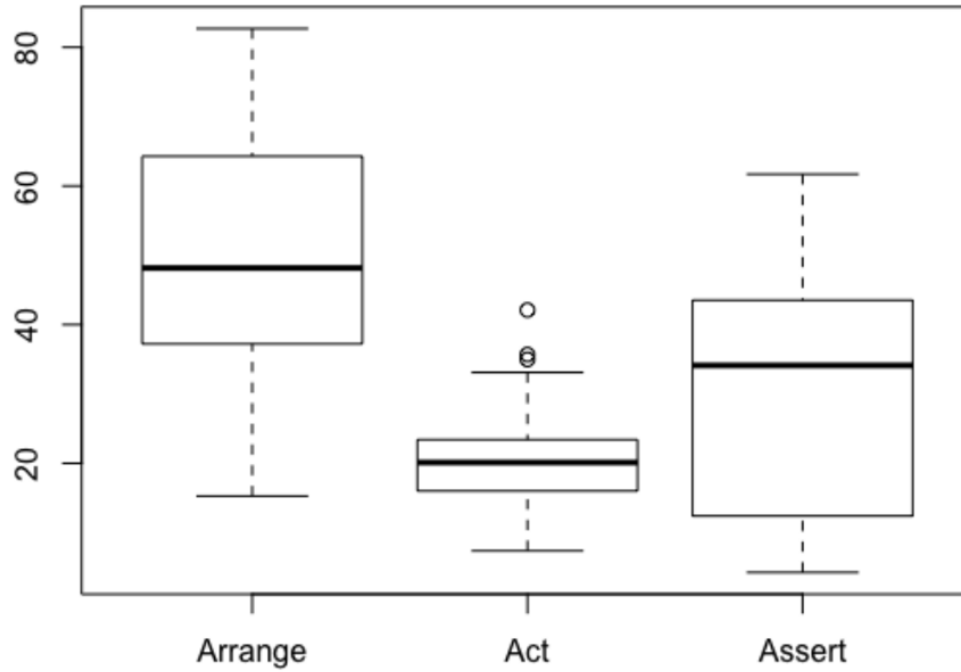
- Other tests

Dependent Variable	Independent Variables	p-value	Effect	$\chi^2(1)$	Marginal R^2	Conditional R^2	McFadden pseudo- R^2	Log Effect
Reading Time	Project Knowl. ($\downarrow$)	0.022	-1.815 ± 1.284 sec.	5.364	0.165	0.726	–	-0.259 ± 0.109
%Arrange	Java ($\uparrow$)	0.009	$+0.867\% \pm 0.328\%$	6.73	0.049	0.586	–	–
%Act	–	–	–	–	0.1	0.298	–	–
%Assert	Java ($\downarrow$)	0.020	$-0.636\% \pm 0.276\%$	4.92	0.021	0.776	–	–
Purpose	Tests ($\uparrow$)	0.008	$+e^{0.272} = 1.313$ times	–	–	–	0.216	$+0.272 \pm 0.103$
Basic Test Cases	–	–	–	–	–	–	0.052	–
Domain Test Cases	Tests ($\uparrow$)	0.002	$+e^{0.339} = 1.404$ times	–	–	–	0.157	$+0.339 \pm 0.108$
	Project Knowl. ($\uparrow$)	0.001	$+e^{1.975} = 7.207$ times	–	–	–	0.157	$+1.975 \pm 0.619$
Error Test Cases	Java ($\uparrow$)	0.023	$+e^{0.345} = 1.411$ times	–	–	–	0.279	$+0.345 \pm 0.152$
Other Test Cases	Tests ($\uparrow$)	0.024	$+e^{0.266} = 1.304$ times	–	–	–	0.153	$+0.266 \pm 0.118$

Having prior knowledge of the software under test significantly reduces the time developers have to spend to comprehend its test code.

No evidence for differences in reading time related to other types of experience.

Dependent Variable	Independent Variables	p-value	Effect	$\chi^2(1)$	Marginal R^2	Conditional R^2	McFadden pseudo- R^2	Log Effect
Reading Time	Project Knowl. ($\downarrow$)	0.022	-1.815 ± 1.284 sec.	5.364	0.165	0.726	–	-0.259 ± 0.109
%Arrange	Java ($\uparrow$)	0.009	$+0.867\% \pm 0.328\%$	6.73	0.049	0.586	–	–
%Act	–	–	–	–	0.1	0.298	–	–
%Assert	Java ($\downarrow$)	0.020	$-0.636\% \pm 0.276\%$	4.92	0.021	0.776	–	–
Purpose	Tests ($\uparrow$)	0.008	$+e^{0.272} = 1.313$ times	–	–	–	0.216	$+0.272 \pm 0.103$
Basic Test Cases	–	–	–	–	–	–	0.052	–
Domain Test Cases	Tests ($\uparrow$)	0.002	$+e^{0.339} = 1.404$ times	–	–	–	0.157	$+0.339 \pm 0.108$
	Project Knowl. ($\uparrow$)	0.001	$+e^{1.975} = 7.207$ times	–	–	–	0.157	$+1.975 \pm 0.619$
Error Test Cases	Java ($\uparrow$)	0.023	$+e^{0.345} = 1.411$ times	–	–	–	0.279	$+0.345 \pm 0.152$
Other Test Cases	Tests ($\uparrow$)	0.024	$+e^{0.266} = 1.304$ times	–	–	–	0.153	$+0.266 \pm 0.118$



Dependent Variable	Independent Variables	p-value	Effect	$\chi^2(1)$	Marginal R^2	Conditional R^2	McFadden pseudo- R^2	Log Effect
Reading Time	Project Knowl. ($\downarrow$)	0.022	-1.815 ± 1.284 sec.	5.364	0.165	0.726	–	-0.259 ± 0.109
%Arrange	Java ($\uparrow$)	0.009	$+0.867\% \pm 0.328\%$	6.73	0.049	0.586	–	–
%Act	–	–	–	–	0.1	0.298	–	–
%Assert	Java ($\downarrow$)	0.020	$-0.636\% \pm 0.276\%$	4.92	0.021	0.776	–	–
Purpose	Tests ($\uparrow$)	0.008	$+e^{0.272} = 1.313$ times	–	–	–	0.216	$+0.272 \pm 0.103$
Basic Test Cases	–	–	–	–	–	–	0.052	–
Domain Test Cases	Tests ($\uparrow$)	0.002	$+e^{0.339} = 1.404$ times	–	–	–	0.157	$+0.339 \pm 0.108$
	Project Knowl. ($\uparrow$)	0.001	$+e^{1.975} = 7.207$ times	–	–	–	0.157	$+1.975 \pm 0.619$
Error Test Cases	Java ($\uparrow$)	0.023	$+e^{0.345} = 1.411$ times	–	–	–	0.279	$+0.345 \pm 0.152$
Other Test Cases	Tests ($\uparrow$)	0.024	$+e^{0.266} = 1.304$ times	–	–	–	0.153	$+0.266 \pm 0.118$

Experience with Java increases the amount of time spent on reading the Arrange section of a test case.

Dependent Variable	Independent Variables	p-value	Effect	$\chi^2(1)$	Marginal R^2	Conditional R^2	McFadden pseudo- R^2	Log Effect
Reading Time	Project Knowl. ($\downarrow$)	0.022	-1.815 ± 1.284 sec.	5.364	0.165	0.726	–	-0.259 ± 0.109
%Arrange	Java ($\uparrow$)	0.009	$+0.867\% \pm 0.328\%$	6.73	0.049	0.586	–	–
%Act	–	–	–	–	0.1	0.298	–	–
%Assert	Java ($\downarrow$)	0.020	$-0.636\% \pm 0.276\%$	4.92	0.021	0.776	–	–
Purpose	Tests ($\uparrow$)	0.008	$+e^{0.272} = 1.313$ times	–	–	–	0.216	$+0.272 \pm 0.103$
Basic Test Cases	–	–	–	–	–	–	0.052	–
Domain Test Cases	Tests ($\uparrow$)	0.002	$+e^{0.339} = 1.404$ times	–	–	–	0.157	$+0.339 \pm 0.108$
	Project Knowl. ($\uparrow$)	0.001	$+e^{1.975} = 7.207$ times	–	–	–	0.157	$+1.975 \pm 0.619$
Error Test Cases	Java ($\uparrow$)	0.023	$+e^{0.345} = 1.411$ times	–	–	–	0.279	$+0.345 \pm 0.152$
Other Test Cases	Tests ($\uparrow$)	0.024	$+e^{0.266} = 1.304$ times	–	–	–	0.153	$+0.266 \pm 0.118$

Experience with Java reduces the amount of time spent on reading test assertions.

Dependent Variable	Independent Variables	p-value	Effect	$\chi^2(1)$	Marginal R^2	Conditional R^2	McFadden pseudo- R^2	Log Effect
Reading Time	Project Knowl. ($\downarrow$)	0.022	-1.815 ± 1.284 sec.	5.364	0.165	0.726	–	-0.259 ± 0.109
%Arrange	Java ($\uparrow$)	0.009	$+0.867\% \pm 0.328\%$	6.73	0.049	0.586	–	–
%Act	–	–	–	–	0.1	0.298	–	–
%Assert	Java ($\downarrow$)	0.020	$-0.636\% \pm 0.276\%$	4.92	0.021	0.776	–	–
Purpose	Tests ($\uparrow$)	0.008	$+e^{0.272} = 1.313$ times	–	–	–	0.216	$+0.272 \pm 0.103$
Basic Test Cases	–	–	–	–	–	–	0.052	–
Domain Test Cases	Tests ($\uparrow$)	0.002	$+e^{0.339} = 1.404$ times	–	–	–	0.157	$+0.339 \pm 0.108$
	Project Knowl. ($\uparrow$)	0.001	$+e^{1.975} = 7.207$ times	–	–	–	0.157	$+1.975 \pm 0.619$
Error Test Cases	Java ($\uparrow$)	0.023	$+e^{0.345} = 1.411$ times	–	–	–	0.279	$+0.345 \pm 0.152$
Other Test Cases	Tests ($\uparrow$)	0.024	$+e^{0.266} = 1.304$ times	–	–	–	0.153	$+0.266 \pm 0.118$

Experience with tests significantly improves test code comprehension.

Dependent Variable	Independent Variables	p-value	Effect	$\chi^2(1)$	Marginal R^2	Conditional R^2	McFadden pseudo- R^2	Log Effect
Reading Time	Project Knowl. ($\downarrow$)	0.022	-1.815 ± 1.284 sec.	5.364	0.165	0.726	–	-0.259 ± 0.109
%Arrange	Java ($\uparrow$)	0.009	$+0.867\% \pm 0.328\%$	6.73	0.049	0.586	–	–
%Act	–	–	–	–	0.1	0.298	–	–
%Assert	Java ($\downarrow$)	0.020	$-0.636\% \pm 0.276\%$	4.92	0.021	0.776	–	–
Purpose	Tests ($\uparrow$)	0.008	$+e^{0.272} = 1.313$ times	–	–	–	0.216	$+0.272 \pm 0.103$
Basic Test Cases	–	–	–	–	–	–	0.052	–
Domain Test Cases	Tests ($\uparrow$)	0.002	$+e^{0.339} = 1.404$ times	–	–	–	0.157	$+0.339 \pm 0.108$
	Project Knowl. ($\uparrow$)	0.001	$+e^{1.975} = 7.207$ times	–	–	–	0.157	$+1.975 \pm 0.619$
Error Test Cases	Java ($\uparrow$)	0.023	$+e^{0.345} = 1.411$ times	–	–	–	0.279	$+0.345 \pm 0.152$
Other Test Cases	Tests ($\uparrow$)	0.024	$+e^{0.266} = 1.304$ times	–	–	–	0.153	$+0.266 \pm 0.118$

Prior knowledge of the software project is fundamental for domain-related test cases (even in domains with which most people are already somewhat familiar).

We have no evidence to suggest that the ability of developers to produce basic test cases is correlated to any type of experience.

Dependent Variable	Independent Variables	p-value	Effect	$\chi^2(1)$	Marginal R^2	Conditional R^2	McFadden pseudo- R^2	Log Effect
Reading Time	Project Knowl. ($\downarrow$)	0.022	-1.815 ± 1.284 sec.	5.364	0.165	0.726	–	-0.259 ± 0.109
%Arrange	Java ($\uparrow$)	0.009	$+0.867\% \pm 0.328\%$	6.73	0.049	0.586	–	–
%Act	–	–	–	–	0.1	0.298	–	–
%Assert	Java ($\downarrow$)	0.020	$-0.636\% \pm 0.276\%$	4.92	0.021	0.776	–	–
Purpose	Tests ($\uparrow$)	0.008	$+e^{0.272} = 1.313$ times	–	–	–	0.216	$+0.272 \pm 0.103$
Basic Test Cases	–	–	–	–	–	–	0.052	–
Domain Test Cases	Tests ($\uparrow$)	0.002	$+e^{0.339} = 1.404$ times	–	–	–	0.157	$+0.339 \pm 0.108$
	Project Knowl. ($\uparrow$)	0.001	$+e^{1.975} = 7.207$ times	–	–	–	0.157	$+1.975 \pm 0.619$
Error Test Cases	Java ($\uparrow$)	0.023	$+e^{0.345} = 1.411$ times	–	–	–	0.279	$+0.345 \pm 0.152$
Other Test Cases	Tests ($\uparrow$)	0.024	$+e^{0.266} = 1.304$ times	–	–	–	0.153	$+0.266 \pm 0.118$

Experience with programming language significantly impacts the amount of error test cases a developer can think of.

Dependent Variable	Independent Variables	p-value	Effect	$\chi^2(1)$	Marginal R^2	Conditional R^2	McFadden pseudo- R^2	Log Effect
Reading Time	Project Knowl. ($\downarrow$)	0.022	-1.815 ± 1.284 sec.	5.364	0.165	0.726	–	-0.259 ± 0.109
%Arrange	Java ($\uparrow$)	0.009	$+0.867\% \pm 0.328\%$	6.73	0.049	0.586	–	–
%Act	–	–	–	–	0.1	0.298	–	–
%Assert	Java ($\downarrow$)	0.020	$-0.636\% \pm 0.276\%$	4.92	0.021	0.776	–	–
Purpose	Tests ($\uparrow$)	0.008	$+e^{0.272} = 1.313$ times	–	–	–	0.216	$+0.272 \pm 0.103$
Basic Test Cases	–	–	–	–	–	–	0.052	–
Domain Test Cases	Tests ($\uparrow$)	0.002	$+e^{0.339} = 1.404$ times	–	–	–	0.157	$+0.339 \pm 0.108$
	Project Knowl. ($\uparrow$)	0.001	$+e^{1.975} = 7.207$ times	–	–	–	0.157	$+1.975 \pm 0.619$
Error Test Cases	Java ($\uparrow$)	0.023	$+e^{0.345} = 1.411$ times	–	–	–	0.279	$+0.345 \pm 0.152$
Other Test Cases	Tests ($\uparrow$)	0.024	$+e^{0.266} = 1.304$ times	–	–	–	0.153	$+0.266 \pm 0.118$

Different types of experience influence different activities related to test code comprehension.

Dependent Variable	Independent Variables	p-value	Effect	$\chi^2(1)$	Marginal R^2	Conditional R^2	McFadden pseudo- R^2	Log Effect
Reading Time	Project Knowl. ($\downarrow$)	0.022	-1.815 ± 1.284 sec.	5.364	0.165	0.726	–	-0.259 ± 0.109
%Arrange	Java ($\uparrow$)	0.009	$+0.867\% \pm 0.328\%$	6.73	0.049	0.586	–	–
%Act	–	–	–	–	0.1	0.298	–	–
%Assert	Java ($\downarrow$)	0.020	$-0.636\% \pm 0.276\%$	4.92	0.021	0.776	–	–
Purpose	Tests ($\uparrow$)	0.008	$+e^{0.272} = 1.313$ times	–	–	–	0.216	$+0.272 \pm 0.103$
Basic Test Cases	–	–	–	–	–	–	0.052	–
Domain Test Cases	Tests ($\uparrow$)	0.002	$+e^{0.339} = 1.404$ times	–	–	–	0.157	$+0.339 \pm 0.108$
	Project Knowl. ($\uparrow$)	0.001	$+e^{1.975} = 7.207$ times	–	–	–	0.157	$+1.975 \pm 0.619$
Error Test Cases	Java ($\uparrow$)	0.023	$+e^{0.345} = 1.411$ times	–	–	–	0.279	$+0.345 \pm 0.152$
Other Test Cases	Tests ($\uparrow$)	0.024	$+e^{0.266} = 1.304$ times	–	–	–	0.153	$+0.266 \pm 0.118$

Practical Recommendations

become **familiar** with source code since source code familiarity is related to faster test code comprehension

ensure that **Arrange** part of tests are clear, concise and self-explanatory since developers spend much time reading it

ask experienced developers to **review** test code from novices, as odds are that they will not test domain cases (and error cases, in case of language novices)

Threats to Validity

time restriction

- no participant used entire time

no access to production code

- necessary to study test code comprehension

evaluated purpose and additional cases manually

- raw data available, new metrics

5-line window

- similar to related work

generalisability

- first study on test code comprehension

Comprehending Test Code

prior knowledge of project

reading time ↓

Java experience

time spent on Arrange ↑, time spent on Assert ↓

Java experience, prior knowledge of project

specific kinds of additional test cases ↑

most positive influential factor towards understanding
and extending a test suite: **experience with using tests**

Comprehending Test Code

prior knowledge of project

reading time ↓

Java experience

time spent on Arrange ↑, time spent on Assert ↓

Java experience, prior knowledge of project

specific kinds of additional test cases ↑

most positive influential factor towards understanding
and extending a test suite: **experience with using tests**

Thanks!

christoph.treude@adelaide.edu.au