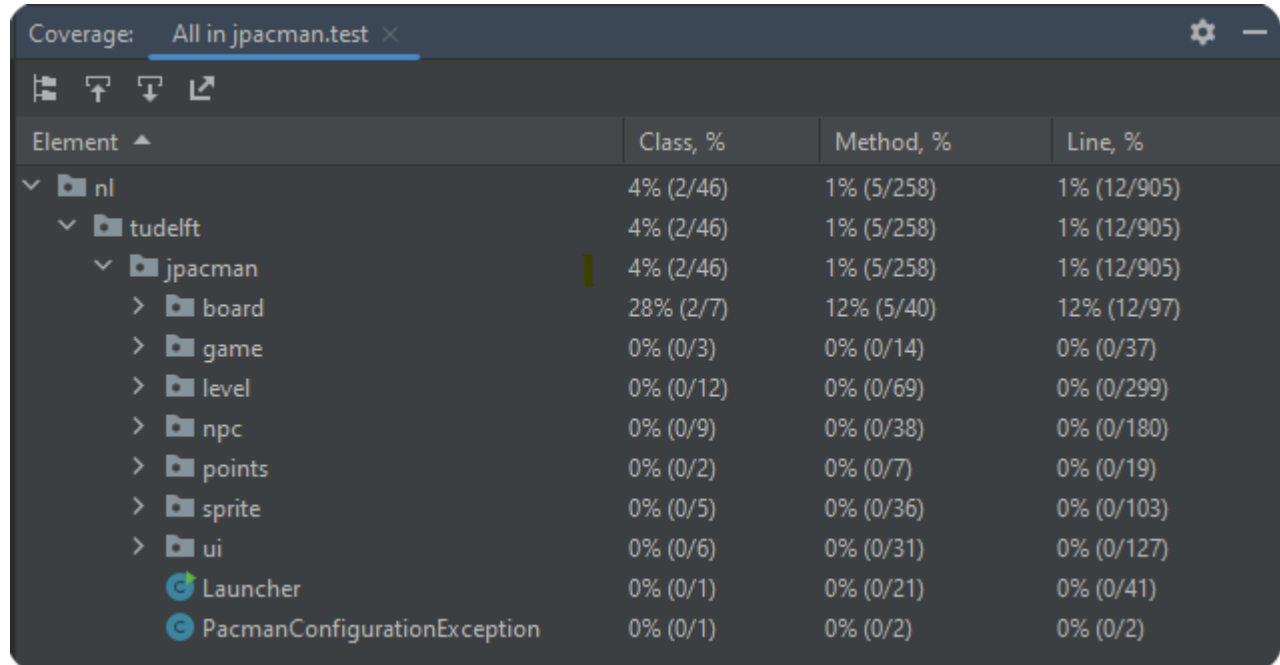


Lab 2: Dynamic Analysis and Unit Testing

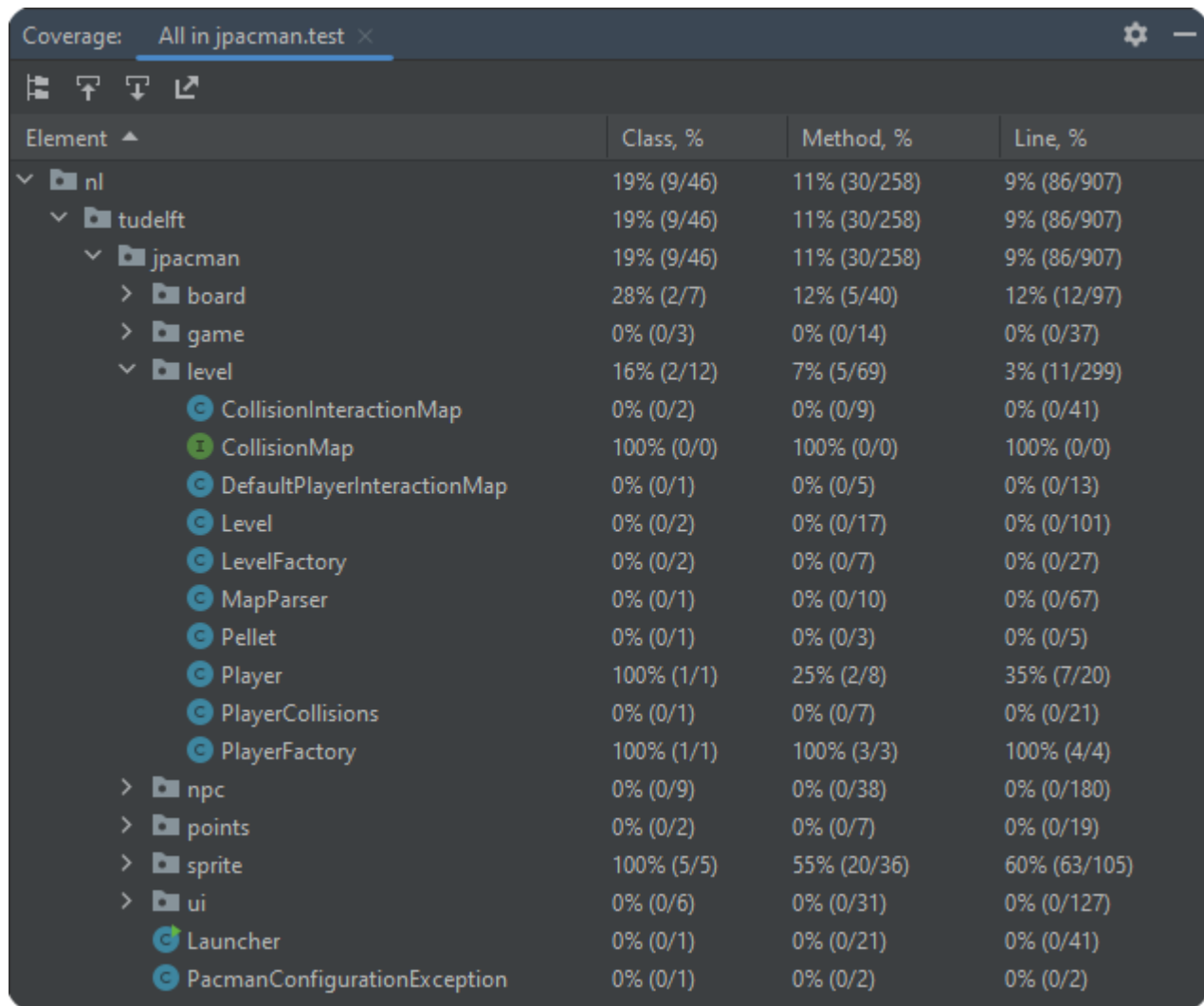
Task 1

A screenshot of the IntelliJ IDEA coverage report window. The title bar says 'Coverage: All in jpacman.test'. The window contains a table with four columns: 'Element', 'Class, %', 'Method, %', and 'Line, %'. The 'Element' column shows a tree structure of packages and classes. The 'Class, %' column shows the percentage of class coverage with the number of classes in parentheses. The 'Method, %' column shows the percentage of method coverage with the number of methods in parentheses. The 'Line, %' column shows the percentage of line coverage with the number of lines in parentheses. The packages 'nl', 'tudelft', and 'jpacman' all have 4% class coverage. The 'board' package has 28% class coverage. The 'game', 'level', 'npc', 'points', 'sprite', and 'ui' packages all have 0% class coverage. The 'Launcher' and 'PacmanConfigurationException' classes also have 0% class coverage.

Element	Class, %	Method, %	Line, %
nl	4% (2/46)	1% (5/258)	1% (12/905)
tudelft	4% (2/46)	1% (5/258)	1% (12/905)
jpacman	4% (2/46)	1% (5/258)	1% (12/905)
board	28% (2/7)	12% (5/40)	12% (12/97)
game	0% (0/3)	0% (0/14)	0% (0/37)
level	0% (0/12)	0% (0/69)	0% (0/299)
npc	0% (0/9)	0% (0/38)	0% (0/180)
points	0% (0/2)	0% (0/7)	0% (0/19)
sprite	0% (0/5)	0% (0/36)	0% (0/103)
ui	0% (0/6)	0% (0/31)	0% (0/127)
Launcher	0% (0/1)	0% (0/21)	0% (0/41)
PacmanConfigurationException	0% (0/1)	0% (0/2)	0% (0/2)

Preliminary test coverage results after running `jpacman.test` (shown above). Results shows lacking test coverage with the majority of the packages having zero to less than five percent test coverage.

Task 2



Coverage: All in jpacman.test			
Element	Class, %	Method, %	Line, %
nl	19% (9/46)	11% (30/258)	9% (86/907)
tudelft	19% (9/46)	11% (30/258)	9% (86/907)
jpacman	19% (9/46)	11% (30/258)	9% (86/907)
board	28% (2/7)	12% (5/40)	12% (12/97)
game	0% (0/3)	0% (0/14)	0% (0/37)
level	16% (2/12)	7% (5/69)	3% (11/299)
CollisionInteractionMap	0% (0/2)	0% (0/9)	0% (0/41)
CollisionMap	100% (0/0)	100% (0/0)	100% (0/0)
DefaultPlayerInteractionMap	0% (0/1)	0% (0/5)	0% (0/13)
Level	0% (0/2)	0% (0/17)	0% (0/101)
LevelFactory	0% (0/2)	0% (0/7)	0% (0/27)
MapParser	0% (0/1)	0% (0/10)	0% (0/67)
Pellet	0% (0/1)	0% (0/3)	0% (0/5)
Player	100% (1/1)	25% (2/8)	35% (7/20)
PlayerCollisions	0% (0/1)	0% (0/7)	0% (0/21)
PlayerFactory	100% (1/1)	100% (3/3)	100% (4/4)
npc	0% (0/9)	0% (0/38)	0% (0/180)
points	0% (0/2)	0% (0/7)	0% (0/19)
sprite	100% (5/5)	55% (20/36)	60% (63/105)
ui	0% (0/6)	0% (0/31)	0% (0/127)
Launcher	0% (0/1)	0% (0/21)	0% (0/41)
PacmanConfigurationException	0% (0/1)	0% (0/2)	0% (0/2)

Test coverage results after adding `PlayerTest.java`. Notice the test coverage increase in the `level` package to 16 percent, up from zero percent.

Task 2.1

Test for `split()` method in `EmptySprite.java`.

Coverage: All in jpacman.test

Element	Class, %	Method, %	Line, %
nl	19% (9/46)	11% (30/258)	9% (86/907)
tudelft	19% (9/46)	11% (30/258)	9% (86/907)
jpacman	19% (9/46)	11% (30/258)	9% (86/907)
board	28% (2/7)	12% (5/40)	12% (12/97)
game	0% (0/3)	0% (0/14)	0% (0/37)
level	16% (2/12)	7% (5/69)	3% (11/299)
npc	0% (0/9)	0% (0/38)	0% (0/180)
points	0% (0/2)	0% (0/7)	0% (0/19)
sprite	100% (5/5)	55% (20/36)	60% (63/105)
AnimatedSprite	100% (1/1)	36% (4/11)	31% (12/38)
EmptySprite	100% (1/1)	0% (0/4)	20% (1/5)
ImageSprite	100% (1/1)	85% (6/7)	80% (12/15)
PacmanSprites	100% (1/1)	55% (5/9)	68% (17/25)
Sprite	100% (0/0)	100% (0/0)	100% (0/0)
SpriteStore	100% (1/1)	100% (5/5)	95% (21/22)
ui	0% (0/6)	0% (0/31)	0% (0/127)
Launcher	0% (0/1)	0% (0/21)	0% (0/41)
PacmanConfigurationException	0% (0/1)	0% (0/2)	0% (0/2)

Coverage: All in jpacman.test (3)

Element	Class, %	Method, %	Line, %
nl	19% (9/46)	12% (33/258)	9% (90/907)
tudelft	19% (9/46)	12% (33/258)	9% (90/907)
jpacman	19% (9/46)	12% (33/258)	9% (90/907)
board	28% (2/7)	12% (5/40)	12% (12/97)
game	0% (0/3)	0% (0/14)	0% (0/37)
level	16% (2/12)	7% (5/69)	3% (11/299)
npc	0% (0/9)	0% (0/38)	0% (0/180)
points	0% (0/2)	0% (0/7)	0% (0/19)
sprite	100% (5/5)	63% (23/36)	63% (67/105)
AnimatedSprite	100% (1/1)	36% (4/11)	31% (12/38)
EmptySprite	100% (1/1)	75% (3/4)	80% (4/5)
ImageSprite	100% (1/1)	85% (6/7)	80% (12/15)
PacmanSprites	100% (1/1)	55% (5/9)	68% (17/25)
Sprite	100% (0/0)	100% (0/0)	100% (0/0)
SpriteStore	100% (1/1)	100% (5/5)	100% (22/22)
ui	0% (0/6)	0% (0/31)	0% (0/127)
Launcher	0% (0/1)	0% (0/21)	0% (0/41)
PacmanConfigurationException	0% (0/1)	0% (0/2)	0% (0/2)

Code for test, `SplitTest.java`:

```
1 package nl.tudelft.jpacman.sprite;
2 import org.junit.jupiter.api.Test;
3 import static org.assertj.core.api.Assertions.assertThat;
4
5 no usages
6 public class SplitTest {
7
8   no usages
9   @Test
10   void testSplit() {
11     EmptySprite sprite = new EmptySprite();
12     Sprite empty = sprite.split(x: 1, y: 1, width: 1, height: 1);
13     assertThat(empty.getWidth()).isEqualTo(expected: 0);
14     assertThat(empty.getHeight()).isEqualTo(expected: 0);
15   }
16 }
```

Test for `pacmanMoved()` method in `DefaultPointCalculator.java`.

Coverage: All in jpacman.test (2) ×

Element	Class, %	Method, %	Line, %
nl	32% (15/46)	19% (50/258)	14% (128/908)
tudelft	32% (15/46)	19% (50/258)	14% (128/908)
jpacman	32% (15/46)	19% (50/258)	14% (128/908)
board	28% (2/7)	12% (5/40)	12% (12/97)
game	0% (0/3)	0% (0/14)	0% (0/37)
level	25% (3/12)	18% (13/69)	9% (28/299)
npc	44% (4/9)	15% (6/38)	7% (14/180)
points	50% (1/2)	14% (1/7)	10% (2/20)
DefaultPointCalculator	100% (1/1)	33% (1/3)	50% (2/4)
PointCalculator	100% (0/0)	100% (0/0)	100% (0/0)
PointCalculatorLoader	0% (0/1)	0% (0/4)	0% (0/16)
sprite	100% (5/5)	69% (25/36)	68% (72/105)
ui	0% (0/6)	0% (0/31)	0% (0/127)
Launcher	0% (0/1)	0% (0/21)	0% (0/41)
PacmanConfigurationException	0% (0/1)	0% (0/2)	0% (0/2)

Coverage: All in jpacman.test (2) ×















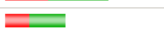










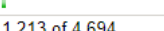
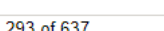

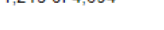
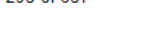
Element	Class, %	Method, %	Line, %
nl	32% (15/46)	19% (51/258)	14% (129/908)
tudelft	32% (15/46)	19% (51/258)	14% (129/908)
jpacman	32% (15/46)	19% (51/258)	14% (129/908)
board	28% (2/7)	12% (5/40)	12% (12/97)
game	0% (0/3)	0% (0/14)	0% (0/37)
level	25% (3/12)	18% (13/69)	9% (28/299)
npc	44% (4/9)	15% (6/38)	7% (14/180)
points	50% (1/2)	28% (2/7)	15% (3/20)
DefaultPointCalculator	100% (1/1)	66% (2/3)	75% (3/4)
PointCalculator	100% (0/0)	100% (0/0)	100% (0/0)
PointCalculatorLoader	0% (0/1)	0% (0/4)	0% (0/16)
sprite	100% (5/5)	69% (25/36)	68% (72/105)
ui	0% (0/6)	0% (0/31)	0% (0/127)
Launcher	0% (0/1)	0% (0/21)	0% (0/41)
PacmanConfigurationException	0% (0/1)	0% (0/2)	0% (0/2)

Code for test, `PacmanMovedTest.java`:

```
1 package nl.tudelft.jpacman.points;
2 import nl.tudelft.jpacman.board.Direction;
3 import nl.tudelft.jpacman.level.Player;
4 import nl.tudelft.jpacman.level.PlayerFactory;
5 import nl.tudelft.jpacman.sprite.PacManSprites;
6 import org.junit.jupiter.api.Test;
7 import static org.assertj.core.api.Assertions.assertThat;
8
9 no usages
10 public class PacmanMovedTest {
11     1 usage
12     private static final PacManSprites SPRITES = new PacManSprites();
13     1 usage
14     private final PlayerFactory factory = new PlayerFactory(SPRITES);
15     3 usages
16     private final Player pacman = factory.createPacMan();
17     1 usage
18     private final PointCalculator score = new DefaultPointCalculator();
19     1 usage
20     private final Direction move = Direction.NORTH;
21
22     no usages
23     @Test
24     void testPacmanMoved() {
25         int points = pacman.getScore();
26         score.pacmanMoved(pacman, move);
27         assertThat(actual: pacman.getScore() == points);
28     }
29 }
```

Task 3

jpacman

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods	Missed	Classes
 nl.tudelft.jpacman.level		67%		57%	74	155	104	344	21	69	4	12
 nl.tudelft.jpacman.npc.ghost		71%		55%	56	105	43	181	5	34	0	8
 nl.tudelft.jpacman.ui		77%		47%	54	86	21	144	7	31	0	6
 default		0%		0%	12	12	21	21	5	5	1	1
 nl.tudelft.jpacman.board		86%		58%	44	93	2	110	0	40	0	7
 nl.tudelft.jpacman.sprite		86%		59%	30	70	11	113	5	38	0	5
 nl.tudelft.jpacman		69%		25%	12	30	18	52	6	24	1	2
 nl.tudelft.jpacman.points		60%		75%	1	11	5	21	0	9	0	2
 nl.tudelft.jpacman.game		87%		60%	10	24	4	45	2	14	0	3
 nl.tudelft.jpacman.npc		100%		n/a	0	4	0	8	0	4	0	1
Total	1,213 of 4,694	74%	293 of 637	54%	293	590	229	1,039	51	268	6	47