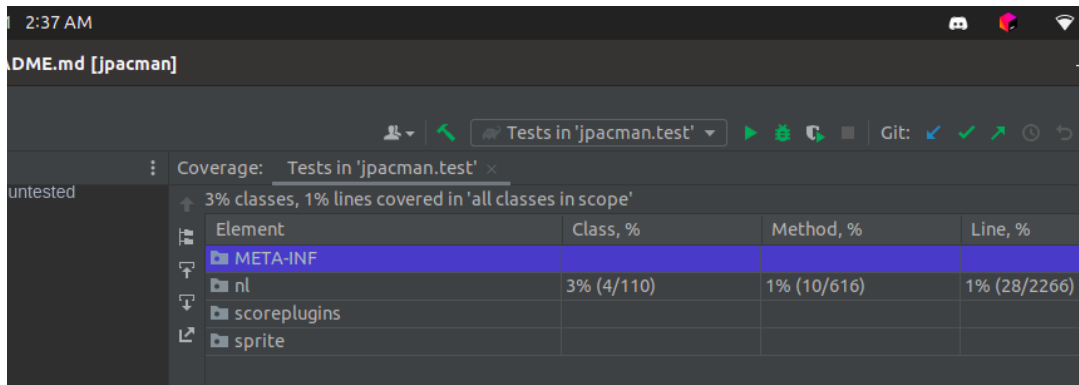


Erick Serrano

02/1/2023

Github Link: <https://github.com/erickserr125/cs472project-erick>

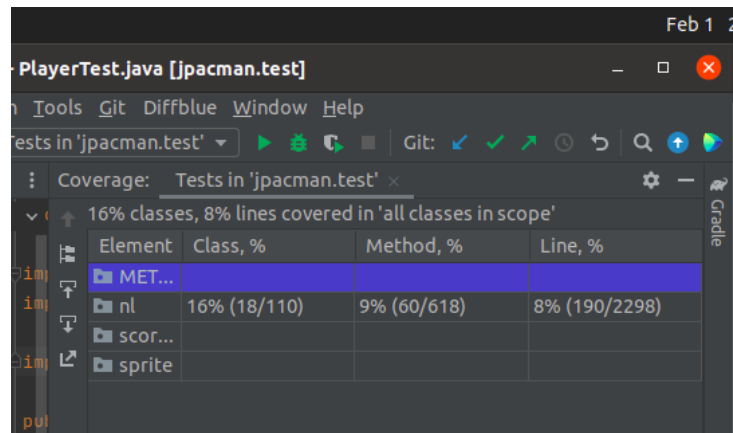
In this laboratory assignment, I focused on developing tests for the **jpacman repository**. The jpacman repository simulates the game “Pacman” using several Java files. With the help of the IntelliJ IDE, we first autogenerated test files and ran the test against the jpacman repository files. Our initial test coverage was relatively low, which is shown below:



The screenshot shows the IntelliJ IDE interface with a test run configuration named 'Tests in 'jpacman.test''. The coverage report indicates that only 3% of classes and 1% of lines are covered. The table below represents the data shown in the coverage report.

Element	Class, %	Method, %	Line, %
META-INF			
nl	3% (4/110)	1% (10/616)	1% (28/2266)
scoreplugins			
sprite			

Clearly, the default tests did not cover the code very well. I sought to improve it by adding additional testing files. In particular, I first added a test which addressed the “isAlive()” method in the player file. By doing so, we dramatically increased the coverage, which is shown below:



The screenshot shows the IntelliJ IDE interface with the same test run configuration. The coverage report now indicates that 16% of classes and 8% of lines are covered, a significant improvement. The table below represents the data shown in the coverage report.

Element	Class, %	Method, %	Line, %
META-INF			
nl	16% (18/110)	9% (60/618)	8% (190/2298)
score...			
sprite			

Clearly, adding crucial tests allows us to further verify the functionality of any piece of code. I then went on to add even more tests, including one for the **Pellet file, which I called PelletTest.java**, one for the **imageSprite class, which I called imageSpriteTest.java**, and one for the **Unit class constructor which I tested in the unitConstructorTest.java file**. The code and results are provided below:

```

1  package nl.tudelft.jpacman.level;
2
3  import ...
4
5
6
7
8  public class pelletTest {
9      //Sprite is an abstract class, checking for the null instance:
10     private static int points = 3;
11     private Pellet pellet= new Pellet(points,null);
12     @Test
13     void testGetSprite(){
14         assertThat(pellet.getSprite() instanceof Sprite);
15         assertThat(pellet.getSprite() == null);
16     }
17 }

```

man.test]

Tests in 'jpacman.test'

Coverage: Tests in 'jpacman.test' x

18% classes, 8% lines covered in 'all classes in scope'

Element	Class, %	Method, %	Line, %
META-INF			
nl	18% (20/110)	10% (64/618)	8% (200/2300)
scoreplugins			
sprite			

Above: Pellet Test Coverage Code & Results

```

1  package nl.tudelft.jpacman.sprite;
2
3  import ...
4
5
6
7
8
9  public class imageSpriteTest {
10     private static final Image im = null;
11     private ImageSprite im_s = new ImageSprite(im);
12     @Test
13     void testNullImageWidth() { assertThat(im_s.getWidth() == 0); }
14
15     @Test
16     void testNullImageHeight() { assertThat(im_s.getHeight() == 0); }
17 }

```

Tests in 'jpacman.test'

Coverage: Tests in 'jpacman.test' x

18% classes, 8% lines covered in 'all classes in scope'

Element	Class, %	Method, %	Line, %
META-INF			
nl	18% (20/110)	10% (66/618)	8% (202/2300)
scoreplug...			
sprite			

Above: Image Sprite Code & Test Coverage Results

Project JDK is not defined

```

1  package nl.tudelft.jpacman.board;
2
3  import ...
4
5
6
7
8
9  public class unitConstructorTest {
10     private static final PacManSprites SPRITE_STORE = new PacManSprites();
11     private PlayerFactory Factory = new PlayerFactory(SPRITE_STORE);
12     private Player ThePlayer = Factory.createPacMan();
13     @Test
14     void testUnitConstructor(){assertThat(ThePlayer.getDirection() == Direction.EAST);}
15 }
16

```

Tests in 'jpacman.test'

Coverage: Tests in 'jpacman.test' x

18% classes, 8% lines covered in 'all classes in scope'

Element	Class, %	Method, %	Line, %
META-INF			
nl	18% (20/110)	10% (66/618)	8% (202/2300)
scoreplugins			
sprite			

Above: Unit Constructor Test Coverage

The result of the tests are described below. Only 2 tests failed, where I checked the ImageSprite width and height.

Test Summary

9	2	0	0.230s
tests	failures	ignored	duration

77%
successful

Failed tests

Packages

Classes

imageSpriteTest. testNullImageHeight().
imageSpriteTest. testNullImageWidth().

Generated by [Gradle 5.3](#) at Feb 1, 2023, 4:47:45 AM