

testSetAlive()

Coverage Tests in 'jpacman.test'				
Element	Class, %	Method, %	Line, %	Branch, %
nl.tudelft.jpacman	14% (8/55)	9% (30/312)	8% (93/1141)	4% (22/539)
sprite	66% (4/6)	44% (20/45)	51% (66/128)	31% (21/66)
board	20% (2/10)	9% (5/53)	9% (14/141)	1% (1/96)
level	15% (2/13)	6% (5/78)	3% (13/348)	0% (0/167)
npc	0% (0/10)	0% (0/47)	0% (0/233)	0% (0/116)
ui	0% (0/6)	0% (0/31)	0% (0/123)	0% (0/60)
Launcher	0% (0/1)	0% (0/21)	0% (0/41)	0% (0/6)
game	0% (0/3)	0% (0/14)	0% (0/37)	0% (0/14)
points	0% (0/2)	0% (0/7)	0% (0/19)	0% (0/4)
fuzzer	0% (0/1)	0% (0/6)	0% (0/32)	0% (0/8)
integration	0% (0/1)	0% (0/4)	0% (0/6)	100% (0/0)
LauncherSmokeTes	0% (0/1)	0% (0/4)	0% (0/29)	0% (0/2)
PacmanConfiguratic	0% (0/1)	0% (0/2)	0% (0/4)	100% (0/0)

Element	Class, %	Method, %	Line, %	Branch, %
nl.tudelft.jpacman	14% (8/55)	10% (32/312)	9% (104/1141)	5% (27/539)
sprite	66% (4/6)	46% (21/45)	54% (70/128)	33% (22/66)
board	20% (2/10)	9% (5/53)	9% (14/141)	1% (1/96)
level	15% (2/13)	7% (6/78)	5% (20/348)	2% (4/167)
npc	0% (0/10)	0% (0/47)	0% (0/233)	0% (0/116)
ui	0% (0/6)	0% (0/31)	0% (0/123)	0% (0/60)
Launcher	0% (0/1)	0% (0/21)	0% (0/41)	0% (0/6)
game	0% (0/3)	0% (0/14)	0% (0/37)	0% (0/14)
points	0% (0/2)	0% (0/7)	0% (0/19)	0% (0/4)
fuzzer	0% (0/1)	0% (0/6)	0% (0/32)	0% (0/8)
integration	0% (0/1)	0% (0/4)	0% (0/6)	100% (0/0)
LauncherSmokeTes	0% (0/1)	0% (0/4)	0% (0/29)	0% (0/2)
PacmanConfiguratic	0% (0/1)	0% (0/2)	0% (0/4)	100% (0/0)

testScore()

Element	Class, %	Method, %	Line, %	Branch, %
nl.tudelft.jpacman	14% (8/55)	10% (32/312)	9% (104/1141)	5% (27/539)
sprite	66% (4/6)	46% (21/45)	54% (70/128)	33% (22/66)
board	20% (2/10)	9% (5/53)	9% (14/141)	1% (1/96)
level	15% (2/13)	7% (6/78)	5% (20/348)	2% (4/167)
npc	0% (0/10)	0% (0/47)	0% (0/233)	0% (0/116)
ui	0% (0/6)	0% (0/31)	0% (0/123)	0% (0/60)
Launcher	0% (0/1)	0% (0/21)	0% (0/41)	0% (0/6)
game	0% (0/3)	0% (0/14)	0% (0/37)	0% (0/14)
points	0% (0/2)	0% (0/7)	0% (0/19)	0% (0/4)
fuzzer	0% (0/1)	0% (0/6)	0% (0/32)	0% (0/8)
integration	0% (0/1)	0% (0/4)	0% (0/6)	100% (0/0)
LauncherSmokeTes	0% (0/1)	0% (0/4)	0% (0/29)	0% (0/2)
PacmanConfiguratic	0% (0/1)	0% (0/2)	0% (0/4)	100% (0/0)

Element	Class, %	Method, %	Line, %	Branch, %
nl.tudelft.jpacman	14% (8/55)	10% (33/312)	9% (105/1141)	5% (27/539)
sprite	66% (4/6)	46% (21/45)	54% (70/128)	33% (22/66)
board	20% (2/10)	9% (5/53)	9% (14/141)	1% (1/96)
level	15% (2/13)	8% (7/78)	6% (21/348)	2% (4/167)
npc	0% (0/10)	0% (0/47)	0% (0/233)	0% (0/116)
ui	0% (0/6)	0% (0/31)	0% (0/123)	0% (0/60)
Launcher	0% (0/1)	0% (0/21)	0% (0/41)	0% (0/6)
game	0% (0/3)	0% (0/14)	0% (0/37)	0% (0/14)
points	0% (0/2)	0% (0/7)	0% (0/19)	0% (0/4)
fuzzer	0% (0/1)	0% (0/6)	0% (0/32)	0% (0/8)
integration	0% (0/1)	0% (0/4)	0% (0/6)	100% (0/0)
LauncherSmokeTes	0% (0/1)	0% (0/4)	0% (0/29)	0% (0/2)
PacmanConfiguratio	0% (0/1)	0% (0/2)	0% (0/4)	100% (0/0)

testSetScore()

Element	Class, %	Method...	Line, %	Branch, %
nl.tudelft.jpacman	14% (8/55)	10% (33/312)	9% (105/1141)	5% (27/539)
sprite	66% (4/6)	46% (21/45)	54% (70/128)	33% (22/66)
board	20% (2/10)	9% (5/53)	9% (14/141)	1% (1/96)
level	15% (2/13)	8% (7/78)	6% (21/348)	2% (4/167)
npc	0% (0/10)	0% (0/47)	0% (0/233)	0% (0/116)
ui	0% (0/6)	0% (0/31)	0% (0/123)	0% (0/60)
Launcher	0% (0/1)	0% (0/21)	0% (0/41)	0% (0/6)
game	0% (0/3)	0% (0/14)	0% (0/37)	0% (0/14)
points	0% (0/2)	0% (0/7)	0% (0/19)	0% (0/4)
fuzzer	0% (0/1)	0% (0/6)	0% (0/32)	0% (0/8)
integration	0% (0/1)	0% (0/4)	0% (0/6)	100% (0/0)
LauncherSmokeTe	0% (0/1)	0% (0/4)	0% (0/29)	0% (0/2)
PacmanConfigurat	0% (0/1)	0% (0/2)	0% (0/4)	100% (0/0)

Element	Class, %	Method...	Line, %	Branch, %
nl.tudelft.jpacman	14% (8/55)	10% (34/312)	9% (107/1141)	5% (27/539)
sprite	66% (4/6)	46% (21/45)	54% (70/128)	33% (22/66)
level	15% (2/13)	10% (8/78)	6% (23/348)	2% (4/167)
board	20% (2/10)	9% (5/53)	9% (14/141)	1% (1/96)
npc	0% (0/10)	0% (0/47)	0% (0/233)	0% (0/116)
ui	0% (0/6)	0% (0/31)	0% (0/123)	0% (0/60)
Launcher	0% (0/1)	0% (0/21)	0% (0/41)	0% (0/6)
game	0% (0/3)	0% (0/14)	0% (0/37)	0% (0/14)
points	0% (0/2)	0% (0/7)	0% (0/19)	0% (0/4)
fuzzer	0% (0/1)	0% (0/6)	0% (0/32)	0% (0/8)
integration	0% (0/1)	0% (0/4)	0% (0/6)	100% (0/0)
LauncherSmokeTe	0% (0/1)	0% (0/4)	0% (0/29)	0% (0/2)
PacmanConfigurat	0% (0/1)	0% (0/2)	0% (0/4)	100% (0/0)

```

19      @Test  ⚙ David
20      void testSetAlive() {
21          player.setAlive(true);
22          assertThat(player.isAlive()).isEqualTo( expected: true);
23
24          player.setAlive(false);
25          assertThat(player.isAlive()).isEqualTo( expected: false);
26      }
27
28      @Test  ⚙ David
29      void testScore() {
30          // Upon initialization, score should be 0
31          assertThat(player.getScore()).isEqualTo( expected: 0);
32      }
33
34      @Test  ⚙ David
35      void testSetScore() {
36          int points = player.getScore();
37          player.addPoints(1);
38
39          assertThat(player.getScore()).isEqualTo( expected: points + 1);
40      }
41  }

```

Method testing functions above.

- Are the coverage results from JaCoCo similar to the ones you got from IntelliJ in the last task? Why so or why not?
 - JaCoCo and the IntelliJ coverage test are very similar however, it appears that IntelliJ recognized a few files missing testing that JaCoCo did not. Notable, these files appear to only be test files; for example, within the “~jpacman.sprite” JaCoCo has all except for “SpriteTest.” It can be said that either JaCoCo is inferior to IntelliJ for not recognizing this file or that it is smarter because it recognized that this is a test file.
- Did you find helpful the source code visualization from JaCoCo on uncovered branches?
 - It was very nice to see the red and green bar as the test filled up; however, IntelliJ’s UI is not bad either. They are both very helpful in conveying information with no real preference for one or the other.
- Which visualization did you prefer and why? IntelliJ’s coverage window or JaCoCo’s report?

- For mere convenience, I believe IntelliJ's coverage window is more convenient because the window is within the same screen. It may be a little more difficult to see some information like the exact count of files counted for in testing and total files but it is more compact.