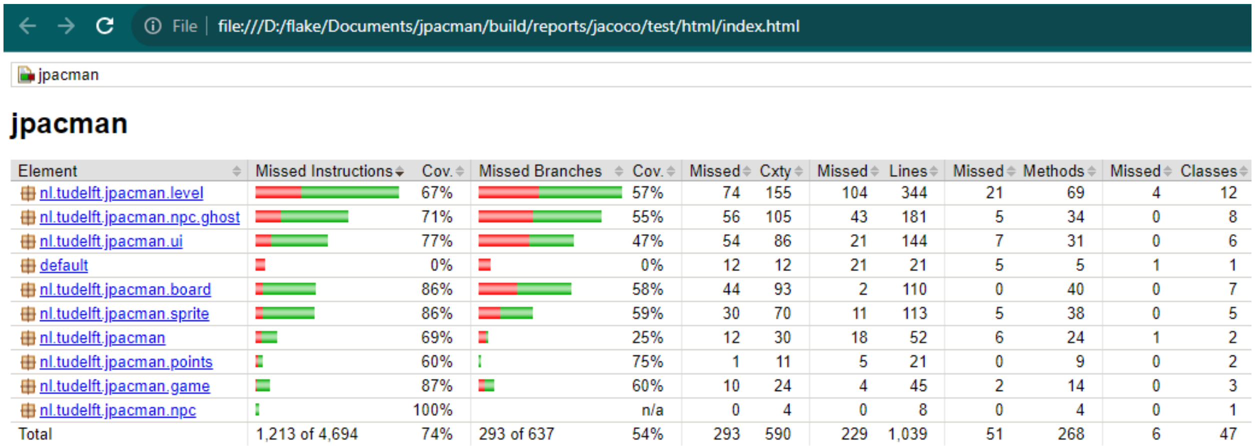
Finnick Lakeotes
CS 472
26 September 2023
Testing – Assignment 2

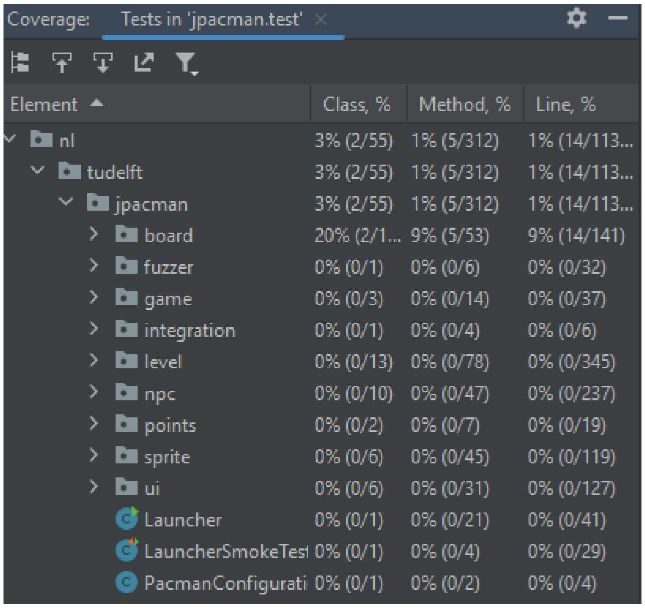
Task 1 – JPacman Test Coverage



File
file:///D:/flake/Documents/jpacman/build/reports/jacoco/test/html/index.html
jpacman

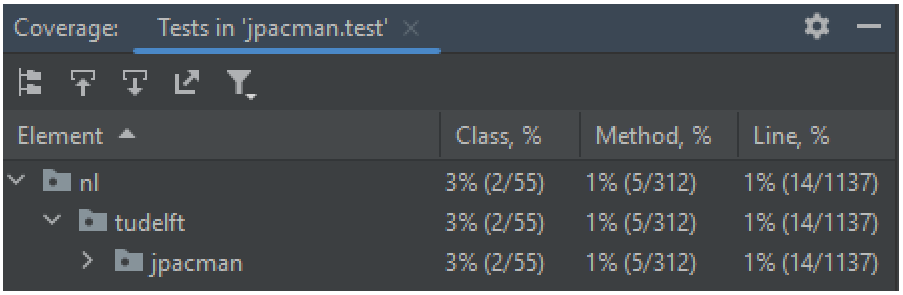
jpacman

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



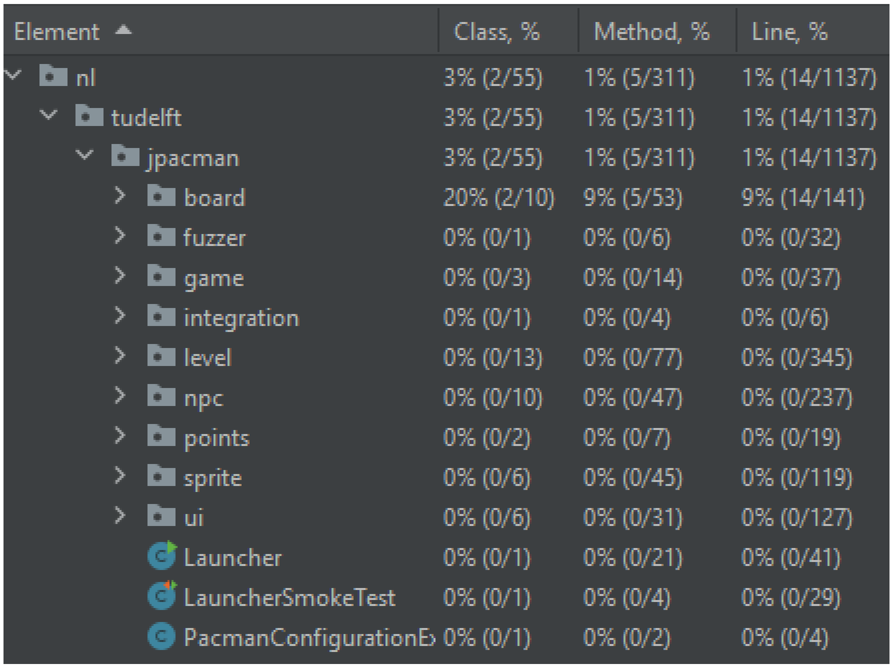
Element	Class, %	Method, %	Line, %
nl	3% (2/55)	1% (5/312)	1% (14/113...
tudelft	3% (2/55)	1% (5/312)	1% (14/113...
jpacman	3% (2/55)	1% (5/312)	1% (14/113...
board	20% (2/1...	9% (5/53)	9% (14/141)
fuzzer	0% (0/1)	0% (0/6)	0% (0/32)
game	0% (0/3)	0% (0/14)	0% (0/37)
integration	0% (0/1)	0% (0/4)	0% (0/6)
level	0% (0/13)	0% (0/78)	0% (0/345)
npc	0% (0/10)	0% (0/47)	0% (0/237)
points	0% (0/2)	0% (0/7)	0% (0/19)
sprite	0% (0/6)	0% (0/45)	0% (0/119)
ui	0% (0/6)	0% (0/31)	0% (0/127)
Launcher	0% (0/1)	0% (0/21)	0% (0/41)
LauncherSmokeTesı	0% (0/1)	0% (0/4)	0% (0/29)
PacmanConfigurati	0% (0/1)	0% (0/2)	0% (0/4)

Task 2 – Increasing Coverage on JPacman



Element	Class, %	Method, %	Line, %
nl	3% (2/55)	1% (5/312)	1% (14/1137)
tudelft	3% (2/55)	1% (5/312)	1% (14/1137)
jpacman	3% (2/55)	1% (5/312)	1% (14/1137)

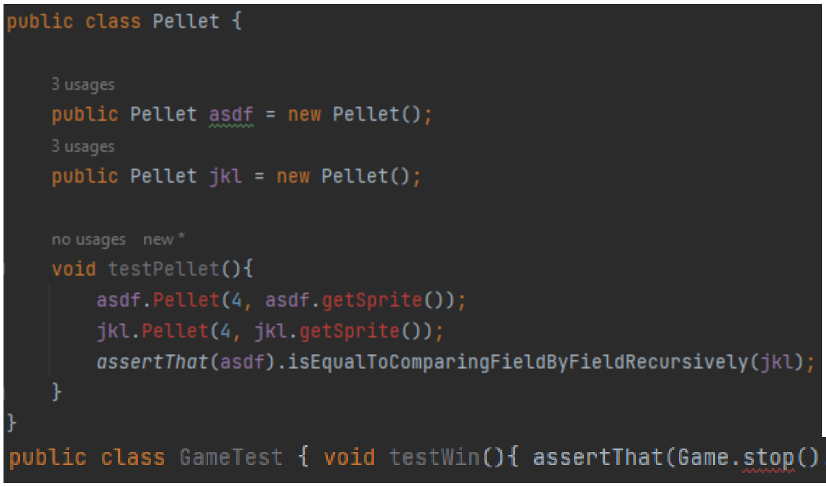
Task 2.1



Element	Class, %	Method, %	Line, %
nl	3% (2/55)	1% (5/311)	1% (14/1137)
tudelft	3% (2/55)	1% (5/311)	1% (14/1137)
jpacman	3% (2/55)	1% (5/311)	1% (14/1137)
board	20% (2/10)	9% (5/53)	9% (14/141)
fuzzer	0% (0/1)	0% (0/6)	0% (0/32)
game	0% (0/3)	0% (0/14)	0% (0/37)
integration	0% (0/1)	0% (0/4)	0% (0/6)
level	0% (0/13)	0% (0/77)	0% (0/345)
npc	0% (0/10)	0% (0/47)	0% (0/237)
points	0% (0/2)	0% (0/7)	0% (0/19)
sprite	0% (0/6)	0% (0/45)	0% (0/119)
ui	0% (0/6)	0% (0/31)	0% (0/127)
Launcher	0% (0/1)	0% (0/21)	0% (0/41)
LauncherSmokeTest	0% (0/1)	0% (0/4)	0% (0/29)
PacmanConfigurationE	0% (0/1)	0% (0/2)	0% (0/4)

Tried writing tests and reading through the code base for 3 hours, was never taught Java in the CSBS degree path so I don’t fully understand it and can’t do this in the time allotted. UNLV CS program needs to consider not all students are able to explore computer languages outside of the degree path. Every day in this degree I become more frustrated by what I have to try and teach myself and not what the degree path is literally supposed to do.

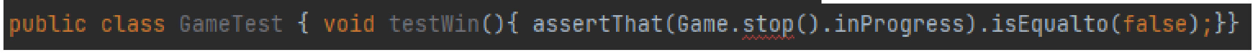
I attempted /level/Pellet, /level/MapParser, and /game/GameTest.



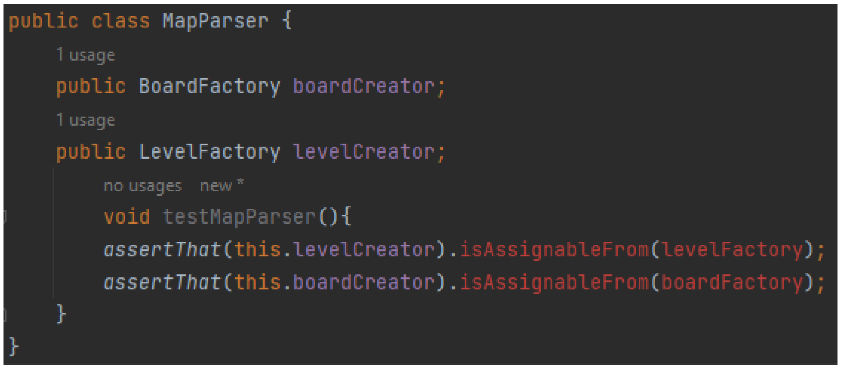
```
public class Pellet {

    3 usages
    public Pellet asdf = new Pellet();
    3 usages
    public Pellet jkl = new Pellet();

    no usages   new *
    void testPellet(){
        asdf.Pellet(4, asdf.getSprite());
        jkl.Pellet(4, jkl.getSprite());
        assertThat(asdf).isEqualToComparingFieldByFieldRecursively(jkl);
    }
}
```



```
public class GameTest { void testWin(){ assertThat(Game.stop().inProgress).isEqualto(false);}}
```



```
public class MapParser {
    1 usage
    public BoardFactory boardCreator;
    1 usage
    public LevelFactory levelCreator;
        no usages   new *
        void testMapParser(){
        assertThat(this.levelCreator).isAssignableFrom(levelFactory);
        assertThat(this.boardCreator).isAssignableFrom(boardFactory);
    }
}
```

Task 3 – JaCoCo Report on JPacman

Are the coverage results from JaCoCo similar to the ones you got from IntelliJ in the last task? Why so or why not?

No, they are not similar. The JaCoCo results are significantly higher than the IntelliJ results. It appears that JaCoCo averages the results from each category, ignoring some files and 100% on others, creating a higher overall result.

Did you find helpful the source code visualization from JaCoCo on uncovered branches?

Yes I did find this helpful. It makes it easy to see coverage.

Which visualization did you prefer and why? IntelliJ's coverage window or JaCoCo's report?

I preferred JaCoCo's display as the visual aid was helpful.

Task 4 & 5

I don't know what to say again. In class Dr. Businge literally said there was no python in this week's assignment after another student complained with my same sentiment. I have had brief exposure to Java but absolutely no prior experience with python. I have no clue how to do this and there is no reasonable way I can get up to speed with my full-time course load while having a family to care for and personal obligations. This degree path is inaccessible to those who aren't 19-year-olds supported by their parents.

Link to fork:
https://github.com/flakeotes/jpacman