Infinite-tux Game

Report
SE401

Test Members

1. Fouad Majd Alkadri |218110075

2.Hisham Adnan | 218110087

3. Abdulaziz Alowain |219110119

4. Serry Sibaee | 218110246

5.Abdullah Abdul Mohimen |218110141

Tool used:

Mokito library:

Mockito is a mocking framework, JAVA-based library that is used for effective unit testing of JAVA applications. Mockito is used to mock interfaces so that a dummy functionality can be added to a mock interface that can be used in unit testing and integration testing.

Source: https://mvnrepository.com/artifact/org.mockito/mockito-core.

Junit 5:

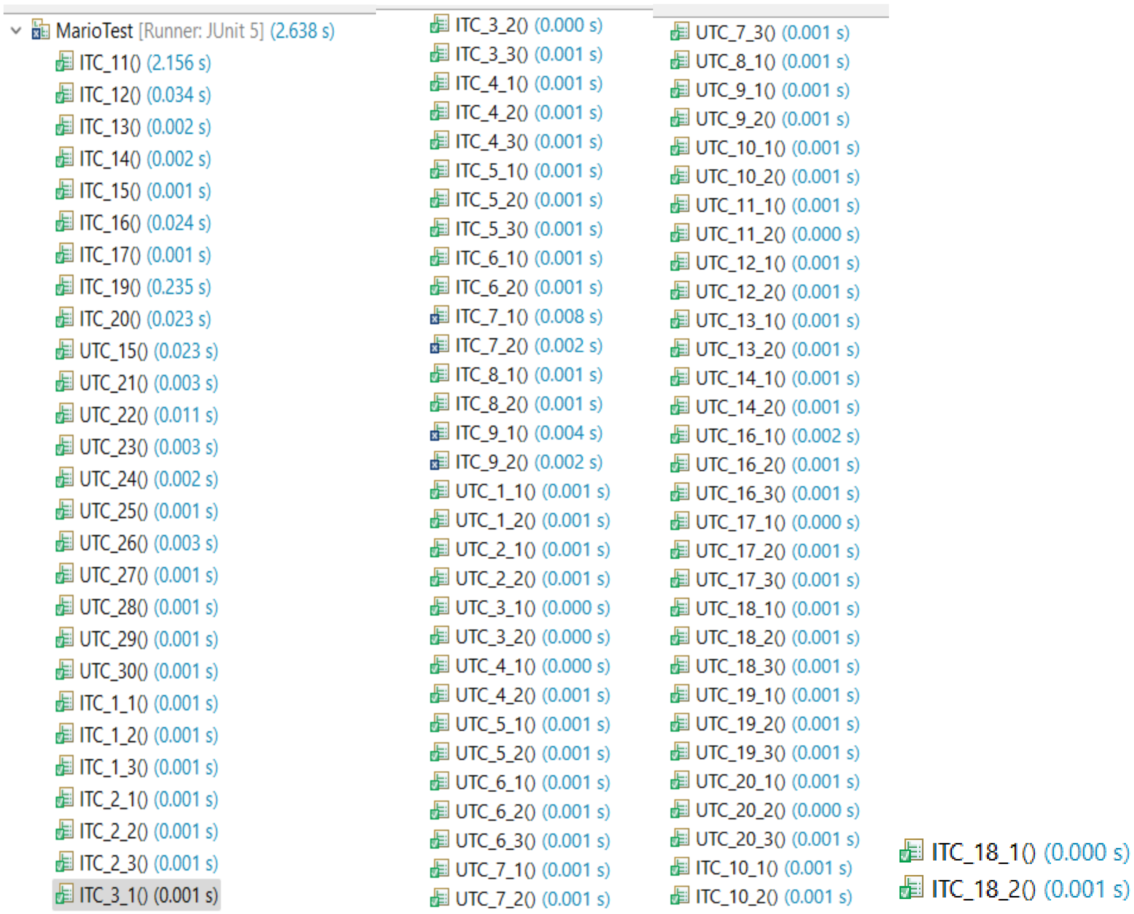
The JUnit Platform serves as a foundation for launching testing frameworks on the JVM.

Source: https://junit.org/junit5/.

CodeMR Complexity Testing:

CodeMR is an architectural software quality and static code analysis tool.
Source: https://www.codemr.co.uk/.

Junit Results:



```
MarioTest [Runner: JUnit 5] (2.638 s)
    ITC_11() (2.156 s)
    ITC_12() (0.034 s)
    ITC_13() (0.002 s)
    ITC_14() (0.002 s)
    ITC_15() (0.001 s)
    ITC_16() (0.024 s)
    ITC_17() (0.001 s)
    ITC_19() (0.235 s)
    ITC_20() (0.023 s)
    UTC_15() (0.023 s)
    UTC_21() (0.003 s)
    UTC_22() (0.011 s)
    UTC_23() (0.003 s)
    UTC_24() (0.002 s)
    UTC_25() (0.001 s)
    UTC_26() (0.003 s)
    UTC_27() (0.001 s)
    UTC_28() (0.001 s)
    UTC_29() (0.001 s)
    UTC_30() (0.001 s)
    ITC_1_1() (0.001 s)
    ITC_1_2() (0.001 s)
    ITC_1_3() (0.001 s)
    ITC_2_1() (0.001 s)
    ITC_2_2() (0.001 s)
    ITC_2_3() (0.001 s)
    ITC_3_1() (0.001 s)
```

```
ITC_3_2() (0.000 s)
ITC_3_3() (0.001 s)
ITC_4_1() (0.001 s)
ITC_4_2() (0.001 s)
ITC_4_3() (0.001 s)
ITC_5_1() (0.001 s)
ITC_5_2() (0.001 s)
ITC_5_3() (0.001 s)
ITC_6_1() (0.001 s)
ITC_6_2() (0.001 s)
ITC_7_1() (0.008 s)
ITC_7_2() (0.002 s)
ITC_8_1() (0.001 s)
ITC_8_2() (0.001 s)
ITC_9_1() (0.004 s)
ITC_9_2() (0.002 s)
UTC_1_1() (0.001 s)
UTC_1_2() (0.001 s)
UTC_2_1() (0.001 s)
UTC_2_2() (0.001 s)
UTC_3_1() (0.000 s)
UTC_3_2() (0.000 s)
UTC_4_1() (0.000 s)
UTC_4_2() (0.001 s)
UTC_5_1() (0.001 s)
UTC_5_2() (0.001 s)
UTC_6_1() (0.001 s)
UTC_6_2() (0.001 s)
UTC_6_3() (0.001 s)
UTC_7_1() (0.001 s)
UTC_7_2() (0.001 s)
```

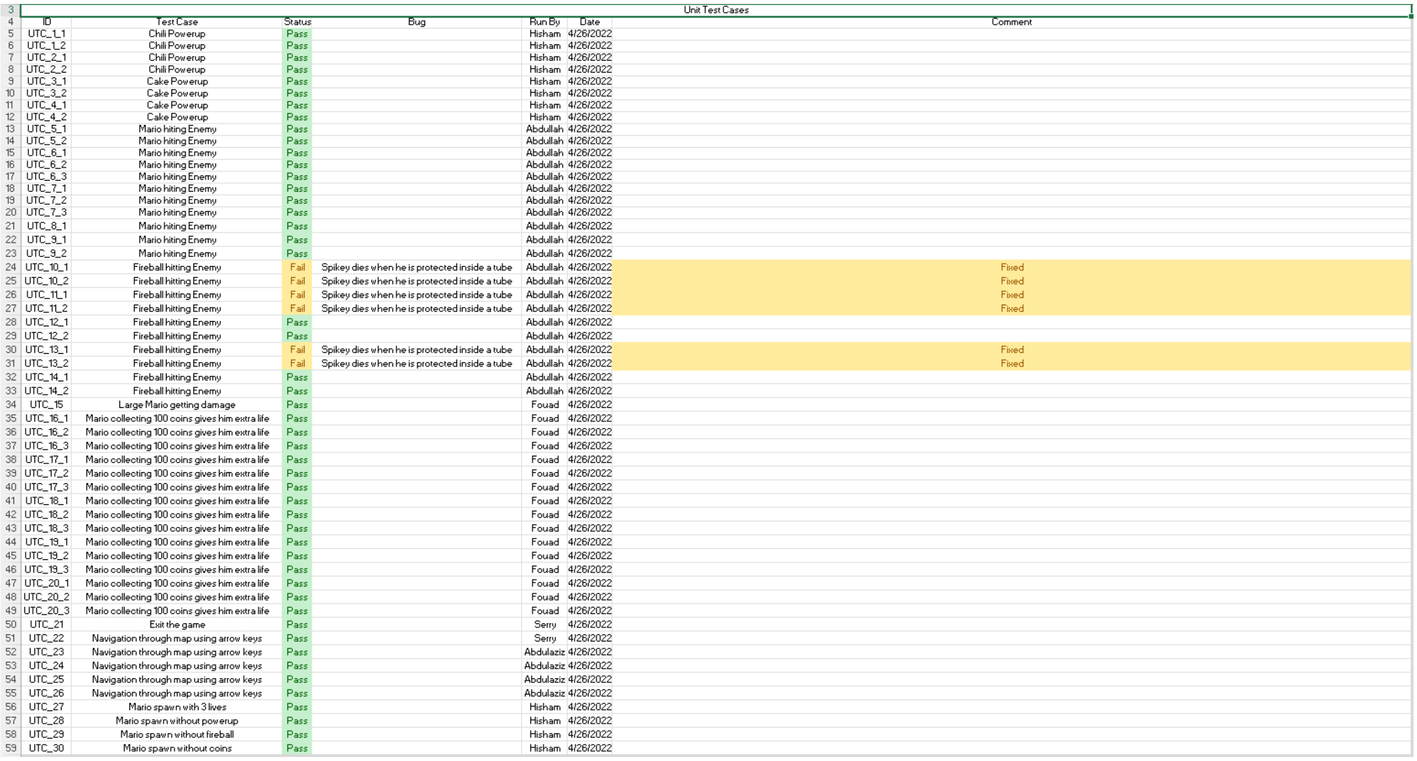
```
UTC_7_3() (0.001 s)
UTC_8_1() (0.001 s)
UTC_9_1() (0.001 s)
UTC_9_2() (0.001 s)
UTC_10_1() (0.001 s)
UTC_10_2() (0.001 s)
UTC_11_1() (0.001 s)
UTC_11_2() (0.000 s)
UTC_12_1() (0.001 s)
UTC_12_2() (0.001 s)
UTC_13_1() (0.001 s)
UTC_13_2() (0.001 s)
UTC_14_1() (0.001 s)
UTC_14_2() (0.001 s)
UTC_16_1() (0.002 s)
UTC_16_2() (0.001 s)
UTC_16_3() (0.001 s)
UTC_17_1() (0.000 s)
UTC_17_2() (0.001 s)
UTC_17_3() (0.001 s)
UTC_18_1() (0.001 s)
UTC_18_2() (0.001 s)
UTC_18_3() (0.001 s)
UTC_19_1() (0.001 s)
UTC_19_2() (0.001 s)
UTC_19_3() (0.001 s)
UTC_20_1() (0.001 s)
UTC_20_2() (0.000 s)
UTC_20_3() (0.001 s)
ITC_10_1() (0.001 s)
ITC_10_2() (0.001 s)
```

```
ITC_18_1() (0.000 s)
ITC_18_2() (0.001 s)
```

Summary Report:

Attached with submission (excel file).

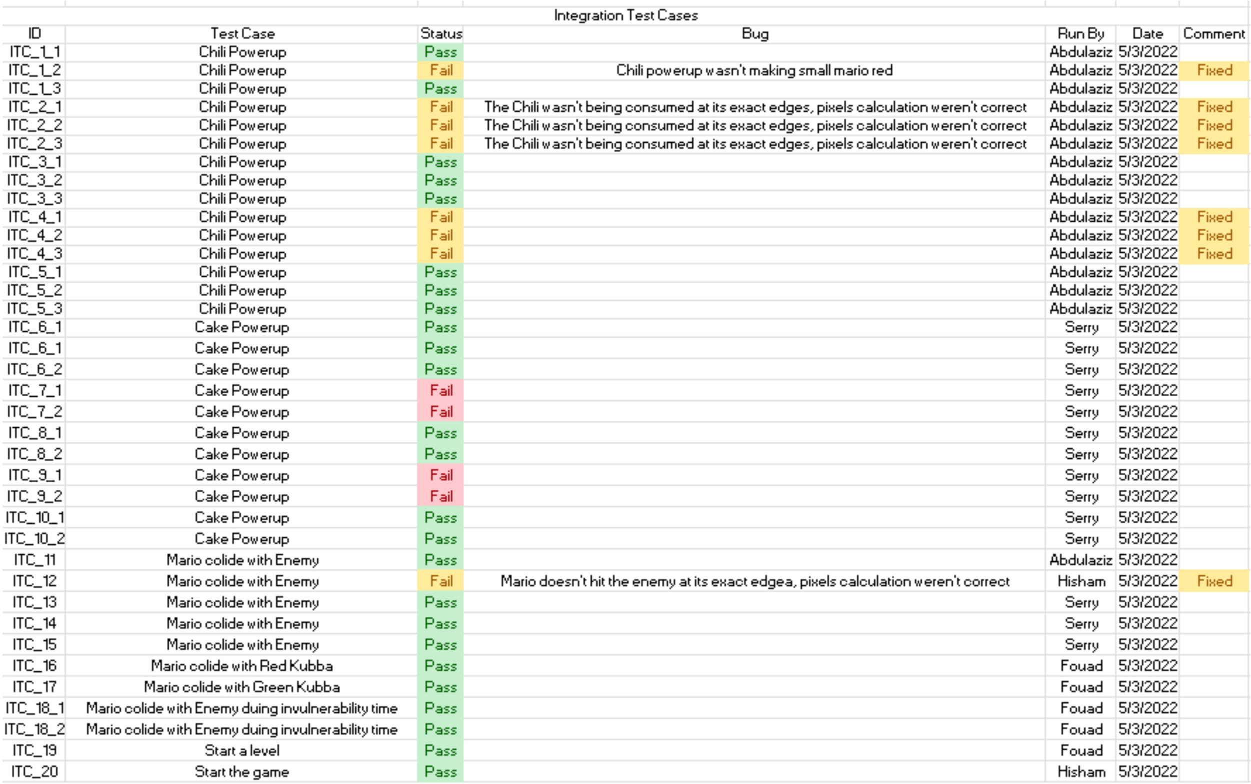
Unit Test Cases:



3 Unit Test Cases

4	ID	Test Case	Status	Bug	Run By	Date	Comment
5	UTC_1_1	Chili Powerup	Pass		Hisham	4/26/2022	
6	UTC_1_2	Chili Powerup	Pass		Hisham	4/26/2022	
7	UTC_2_1	Chili Powerup	Pass		Hisham	4/26/2022	
8	UTC_2_2	Chili Powerup	Pass		Hisham	4/26/2022	
9	UTC_3_1	Cake Powerup	Pass		Hisham	4/26/2022	
10	UTC_3_2	Cake Powerup	Pass		Hisham	4/26/2022	
11	UTC_4_1	Cake Powerup	Pass		Hisham	4/26/2022	
12	UTC_4_2	Cake Powerup	Pass		Hisham	4/26/2022	
13	UTC_5_1	Mario hiting Enemy	Pass		Abdullah	4/26/2022	
14	UTC_5_2	Mario hiting Enemy	Pass		Abdullah	4/26/2022	
15	UTC_6_1	Mario hiting Enemy	Pass		Abdullah	4/26/2022	
16	UTC_6_2	Mario hiting Enemy	Pass		Abdullah	4/26/2022	
17	UTC_6_3	Mario hiting Enemy	Pass		Abdullah	4/26/2022	
18	UTC_7_1	Mario hiting Enemy	Pass		Abdullah	4/26/2022	
19	UTC_7_2	Mario hiting Enemy	Pass		Abdullah	4/26/2022	
20	UTC_7_3	Mario hiting Enemy	Pass		Abdullah	4/26/2022	
21	UTC_8_1	Mario hiting Enemy	Pass		Abdullah	4/26/2022	
22	UTC_9_1	Mario hiting Enemy	Pass		Abdullah	4/26/2022	
23	UTC_9_2	Mario hiting Enemy	Pass		Abdullah	4/26/2022	
24	UTC_10_1	Fireball hitting Enemy	Fail	Spikey dies when he is protected inside a tube	Abdullah	4/26/2022	Fixed
25	UTC_10_2	Fireball hitting Enemy	Fail	Spikey dies when he is protected inside a tube	Abdullah	4/26/2022	Fixed
26	UTC_11_1	Fireball hitting Enemy	Fail	Spikey dies when he is protected inside a tube	Abdullah	4/26/2022	Fixed
27	UTC_11_2	Fireball hitting Enemy	Fail	Spikey dies when he is protected inside a tube	Abdullah	4/26/2022	Fixed
28	UTC_12_1	Fireball hitting Enemy	Pass		Abdullah	4/26/2022	
29	UTC_12_2	Fireball hitting Enemy	Pass		Abdullah	4/26/2022	
30	UTC_13_1	Fireball hitting Enemy	Fail	Spikey dies when he is protected inside a tube	Abdullah	4/26/2022	Fixed
31	UTC_13_2	Fireball hitting Enemy	Fail	Spikey dies when he is protected inside a tube	Abdullah	4/26/2022	Fixed
32	UTC_14_1	Fireball hitting Enemy	Pass		Abdullah	4/26/2022	
33	UTC_14_2	Fireball hitting Enemy	Pass		Abdullah	4/26/2022	
34	UTC_15	Large Mario getting damage	Pass		Fouad	4/26/2022	
35	UTC_16_1	Mario collecting 100 coins gives him extra life	Pass		Fouad	4/26/2022	
36	UTC_16_2	Mario collecting 100 coins gives him extra life	Pass		Fouad	4/26/2022	
37	UTC_16_3	Mario collecting 100 coins gives him extra life	Pass		Fouad	4/26/2022	
38	UTC_17_1	Mario collecting 100 coins gives him extra life	Pass		Fouad	4/26/2022	
39	UTC_17_2	Mario collecting 100 coins gives him extra life	Pass		Fouad	4/26/2022	
40	UTC_17_3	Mario collecting 100 coins gives him extra life	Pass		Fouad	4/26/2022	
41	UTC_18_1	Mario collecting 100 coins gives him extra life	Pass		Fouad	4/26/2022	
42	UTC_18_2	Mario collecting 100 coins gives him extra life	Pass		Fouad	4/26/2022	
43	UTC_18_3	Mario collecting 100 coins gives him extra life	Pass		Fouad	4/26/2022	
44	UTC_19_1	Mario collecting 100 coins gives him extra life	Pass		Fouad	4/26/2022	
45	UTC_19_2	Mario collecting 100 coins gives him extra life	Pass		Fouad	4/26/2022	
46	UTC_19_3	Mario collecting 100 coins gives him extra life	Pass		Fouad	4/26/2022	
47	UTC_20_1	Mario collecting 100 coins gives him extra life	Pass		Fouad	4/26/2022	
48	UTC_20_2	Mario collecting 100 coins gives him extra life	Pass		Fouad	4/26/2022	
49	UTC_20_3	Mario collecting 100 coins gives him extra life	Pass		Fouad	4/26/2022	
50	UTC_21	Exit the game	Pass		Serry	4/26/2022	
51	UTC_22	Navigation through map using arrow keys	Pass		Serry	4/26/2022	
52	UTC_23	Navigation through map using arrow keys	Pass		Abdulaziz	4/26/2022	
53	UTC_24	Navigation through map using arrow keys	Pass		Abdulaziz	4/26/2022	
54	UTC_25	Navigation through map using arrow keys	Pass		Abdulaziz	4/26/2022	
55	UTC_26	Navigation through map using arrow keys	Pass		Abdulaziz	4/26/2022	
56	UTC_27	Mario spawn with 3 lives	Pass		Hisham	4/26/2022	
57	UTC_28	Mario spawn without powerup	Pass		Hisham	4/26/2022	
58	UTC_29	Mario spawn without fireball	Pass		Hisham	4/26/2022	
59	UTC_30	Mario spawn without coins	Pass		Hisham	4/26/2022	

Integration Test Cases:



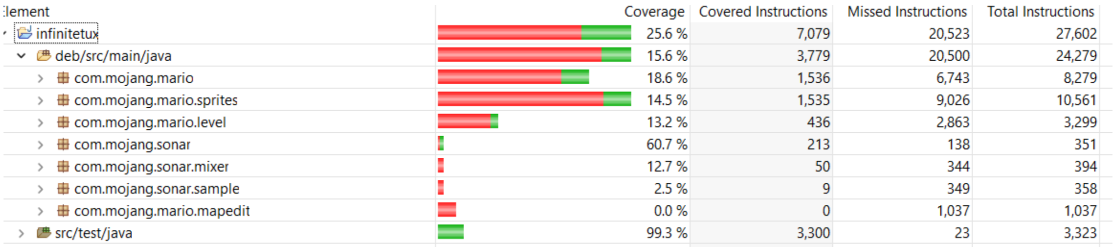
Integration Test Cases

ID	Test Case	Status	Bug	Run By	Date	Comment
ITC_1_1	Chili Powerup	Pass		Abdulaziz	5/3/2022	
ITC_1_2	Chili Powerup	Fail	Chili powerup wasn't making small mario red	Abdulaziz	5/3/2022	Fixed
ITC_1_3	Chili Powerup	Pass		Abdulaziz	5/3/2022	
ITC_2_1	Chili Powerup	Fail	The Chili wasn't being consumed at its exact edges, pixels calculation weren't correct	Abdulaziz	5/3/2022	Fixed
ITC_2_2	Chili Powerup	Fail	The Chili wasn't being consumed at its exact edges, pixels calculation weren't correct	Abdulaziz	5/3/2022	Fixed
ITC_2_3	Chili Powerup	Fail	The Chili wasn't being consumed at its exact edges, pixels calculation weren't correct	Abdulaziz	5/3/2022	Fixed
ITC_3_1	Chili Powerup	Pass		Abdulaziz	5/3/2022	
ITC_3_2	Chili Powerup	Pass		Abdulaziz	5/3/2022	
ITC_3_3	Chili Powerup	Pass		Abdulaziz	5/3/2022	
ITC_4_1	Chili Powerup	Fail		Abdulaziz	5/3/2022	Fixed
ITC_4_2	Chili Powerup	Fail		Abdulaziz	5/3/2022	Fixed
ITC_4_3	Chili Powerup	Fail		Abdulaziz	5/3/2022	Fixed
ITC_5_1	Chili Powerup	Pass		Abdulaziz	5/3/2022	
ITC_5_2	Chili Powerup	Pass		Abdulaziz	5/3/2022	
ITC_5_3	Chili Powerup	Pass		Abdulaziz	5/3/2022	
ITC_6_1	Cake Powerup	Pass		Serry	5/3/2022	
ITC_6_1	Cake Powerup	Pass		Serry	5/3/2022	
ITC_6_2	Cake Powerup	Pass		Serry	5/3/2022	
ITC_7_1	Cake Powerup	Fail		Serry	5/3/2022	
ITC_7_2	Cake Powerup	Fail		Serry	5/3/2022	
ITC_8_1	Cake Powerup	Pass		Serry	5/3/2022	
ITC_8_2	Cake Powerup	Pass		Serry	5/3/2022	
ITC_9_1	Cake Powerup	Fail		Serry	5/3/2022	
ITC_9_2	Cake Powerup	Fail		Serry	5/3/2022	
ITC_10_1	Cake Powerup	Pass		Serry	5/3/2022	
ITC_10_2	Cake Powerup	Pass		Serry	5/3/2022	
ITC_11	Mario colide with Enemy	Pass		Abdulaziz	5/3/2022	
ITC_12	Mario colide with Enemy	Fail	Mario doesn't hit the enemy at its exact edgea, pixels calculation weren't correct	Hisham	5/3/2022	Fixed
ITC_13	Mario colide with Enemy	Pass		Serry	5/3/2022	
ITC_14	Mario colide with Enemy	Pass		Serry	5/3/2022	
ITC_15	Mario colide with Enemy	Pass		Serry	5/3/2022	
ITC_16	Mario colide with Red Kubba	Pass		Fouad	5/3/2022	
ITC_17	Mario colide with Green Kubba	Pass		Fouad	5/3/2022	
ITC_18_1	Mario colide with Enemy duing invulnerability time	Pass		Fouad	5/3/2022	
ITC_18_2	Mario colide with Enemy duing invulnerability time	Pass		Fouad	5/3/2022	
ITC_19	Start a level	Pass		Fouad	5/3/2022	
ITC_20	Start the game	Pass		Hisham	5/3/2022	

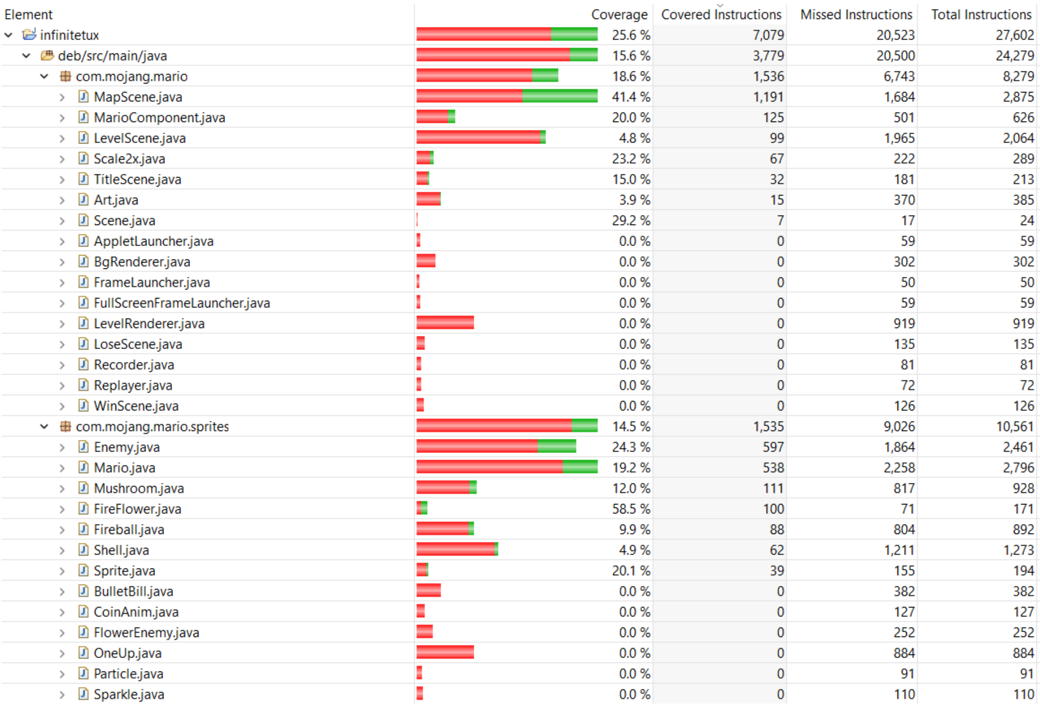
General Coverage results:

White-box testing:

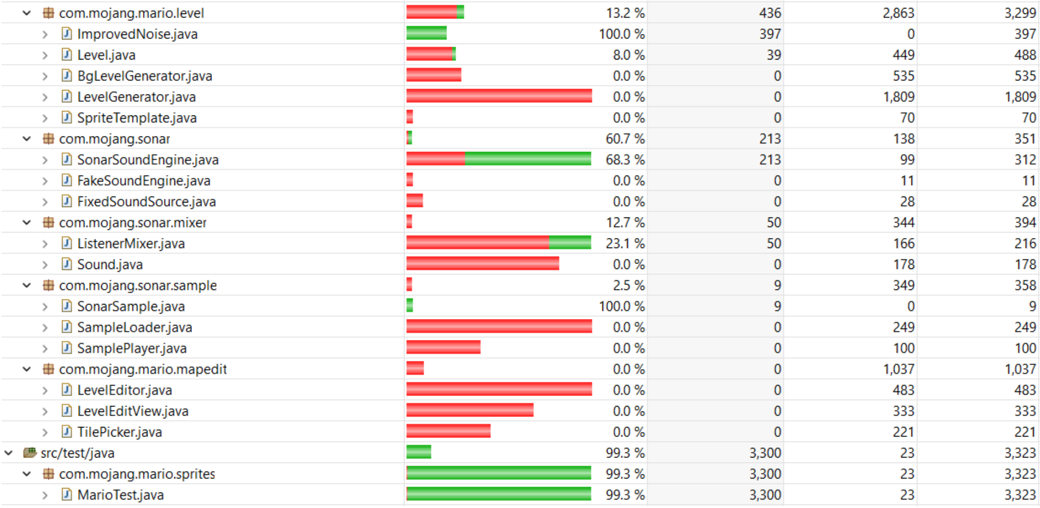
Infinite_tux_master Package:



lement	Coverage	Covered Instructions	Missed Instructions	Total Instructions
infinitetux	25.6 %	7,079	20,523	27,602
deb/src/main/java	15.6 %	3,779	20,500	24,279
com.mojang.mario	18.6 %	1,536	6,743	8,279
com.mojang.mario.sprites	14.5 %	1,535	9,026	10,561
com.mojang.mario.level	13.2 %	436	2,863	3,299
com.mojang.sonar	60.7 %	213	138	351
com.mojang.sonar.mixer	12.7 %	50	344	394
com.mojang.sonar.sample	2.5 %	9	349	358
com.mojang.mario.mapedit	0.0 %	0	1,037	1,037
src/test/java	99.3 %	3,300	23	3,323

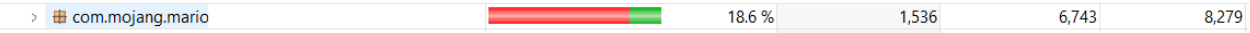


Element	Coverage	Covered Instructions	Missed Instructions	Total Instructions
infinitetux	25.6 %	7,079	20,523	27,602
deb/src/main/java	15.6 %	3,779	20,500	24,279
com.mojang.mario	18.6 %	1,536	6,743	8,279
MapScene.java	41.4 %	1,191	1,684	2,875
MarioComponent.java	20.0 %	125	501	626
LevelScene.java	4.8 %	99	1,965	2,064
Scale2x.java	23.2 %	67	222	289
TitleScene.java	15.0 %	32	181	213
Art.java	3.9 %	15	370	385
Scene.java	29.2 %	7	17	24
AppletLauncher.java	0.0 %	0	59	59
BgRenderer.java	0.0 %	0	302	302
FrameLauncher.java	0.0 %	0	50	50
FullScreenFrameLauncher.java	0.0 %	0	59	59
LevelRenderer.java	0.0 %	0	919	919
LoseScene.java	0.0 %	0	135	135
Recorder.java	0.0 %	0	81	81
Replayer.java	0.0 %	0	72	72
WinScene.java	0.0 %	0	126	126
com.mojang.mario.sprites	14.5 %	1,535	9,026	10,561
Enemy.java	24.3 %	597	1,864	2,461
Mario.java	19.2 %	538	2,258	2,796
Mushroom.java	12.0 %	111	817	928
FireFlower.java	58.5 %	100	71	171
Fireball.java	9.9 %	88	804	892
Shell.java	4.9 %	62	1,211	1,273
Sprite.java	20.1 %	39	155	194
BulletBill.java	0.0 %	0	382	382
CoinAnim.java	0.0 %	0	127	127
FlowerEnemy.java	0.0 %	0	252	252
OneUp.java	0.0 %	0	884	884
Particle.java	0.0 %	0	91	91
Sparkle.java	0.0 %	0	110	110



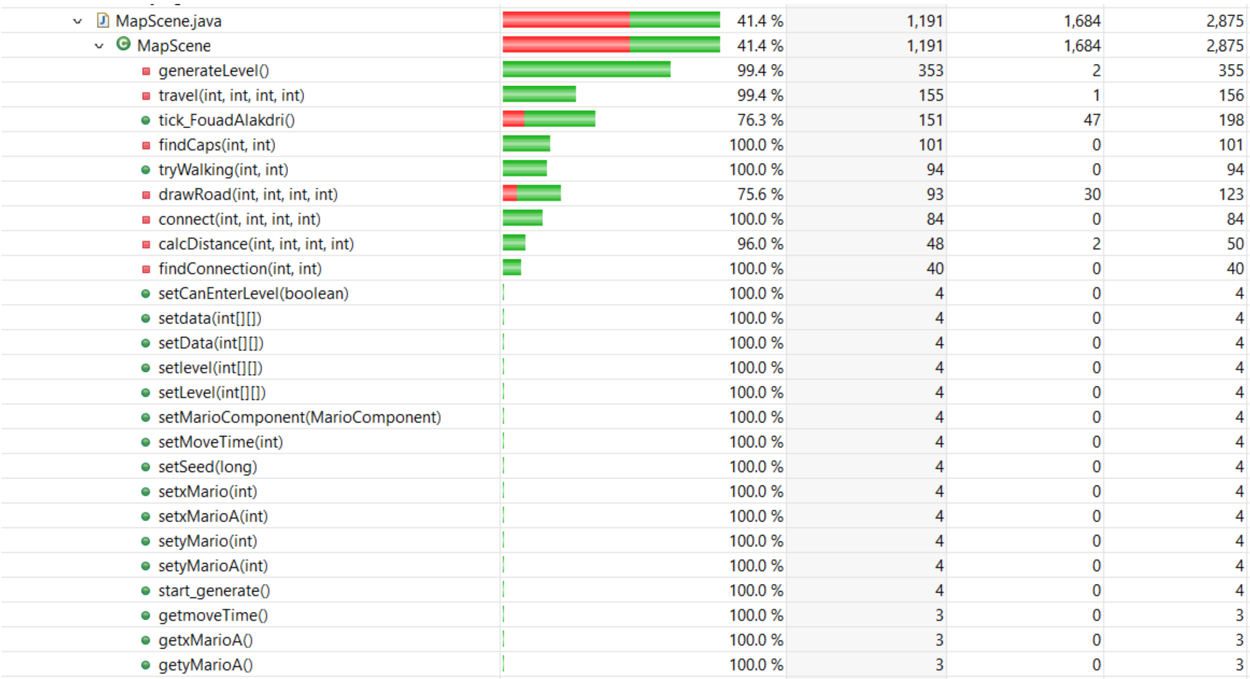
com.mojang.mario.level	13.2 %	436	2,863	3,299
ImprovedNoise.java	100.0 %	397	0	397
Level.java	8.0 %	39	449	488
BgLevelGenerator.java	0.0 %	0	535	535
LevelGenerator.java	0.0 %	0	1,809	1,809
SpriteTemplate.java	0.0 %	0	70	70
com.mojang.sonar	60.7 %	213	138	351
SonarSoundEngine.java	68.3 %	213	99	312
FakeSoundEngine.java	0.0 %	0	11	11
FixedSoundSource.java	0.0 %	0	28	28
com.mojang.sonar.mixer	12.7 %	50	344	394
ListenerMixer.java	23.1 %	50	166	216
Sound.java	0.0 %	0	178	178
com.mojang.sonar.sample	2.5 %	9	349	358
SonarSample.java	100.0 %	9	0	9
SampleLoader.java	0.0 %	0	249	249
SamplePlayer.java	0.0 %	0	100	100
com.mojang.mario.mapedit	0.0 %	0	1,037	1,037
LevelEditor.java	0.0 %	0	483	483
LevelEditView.java	0.0 %	0	333	333
TilePicker.java	0.0 %	0	221	221
src/test/java	99.3 %	3,300	23	3,323
com.mojang.mario.sprites	99.3 %	3,300	23	3,323
MarioTest.java	99.3 %	3,300	23	3,323

Package com.mojang.mario:



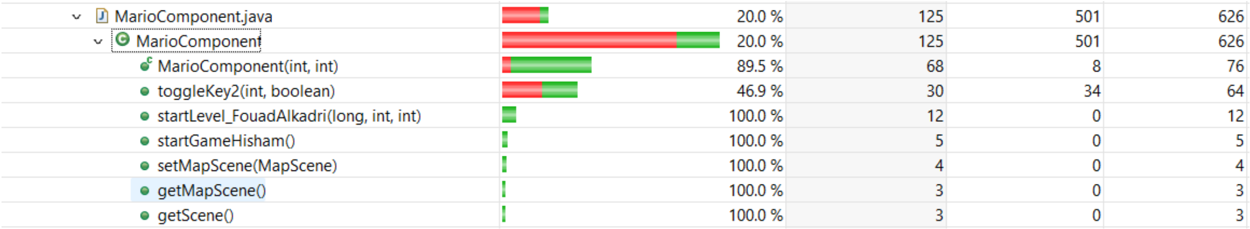
com.mojang.mario	18.6 %	1,536	6,743	8,279

MapScene Class:



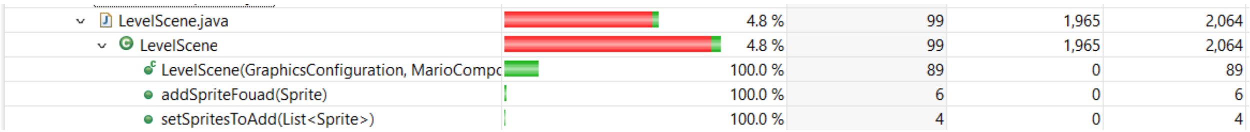
MapScene.java	41.4 %	1,191	1,684	2,875
MapScene	41.4 %	1,191	1,684	2,875
generateLevel()	99.4 %	353	2	355
travel(int, int, int, int)	99.4 %	155	1	156
tick_FouadAlakdri()	76.3 %	151	47	198
findCaps(int, int)	100.0 %	101	0	101
tryWalking(int, int)	100.0 %	94	0	94
drawRoad(int, int, int, int)	75.6 %	93	30	123
connect(int, int, int, int)	100.0 %	84	0	84
calcDistance(int, int, int, int)	96.0 %	48	2	50
findConnection(int, int)	100.0 %	40	0	40
setCanEnterLevel(boolean)	100.0 %	4	0	4
setdata(int[][])	100.0 %	4	0	4
setData(int[][])	100.0 %	4	0	4
setlevel(int[][])	100.0 %	4	0	4
setLevel(int[][])	100.0 %	4	0	4
setMarioComponent(MarioComponent)	100.0 %	4	0	4
setMoveTime(int)	100.0 %	4	0	4
setSeed(long)	100.0 %	4	0	4
setxMario(int)	100.0 %	4	0	4
setxMarioA(int)	100.0 %	4	0	4
setyMario(int)	100.0 %	4	0	4
setyMarioA(int)	100.0 %	4	0	4
start_generate()	100.0 %	4	0	4
getmoveTime()	100.0 %	3	0	3
getxMarioA()	100.0 %	3	0	3
getyMarioA()	100.0 %	3	0	3

MarioComponent Class:



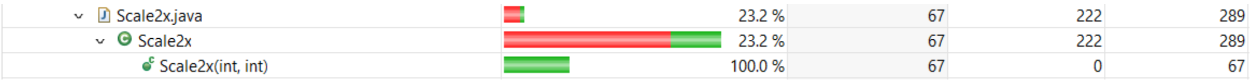
MarioComponent.java	20.0 %	125	501	626
MarioComponent	20.0 %	125	501	626
MarioComponent(int, int)	89.5 %	68	8	76
toggleKey2(int, boolean)	46.9 %	30	34	64
startLevel_FouadAlkadri(long, int, int)	100.0 %	12	0	12
startGameHisham()	100.0 %	5	0	5
setMapScene(MapScene)	100.0 %	4	0	4
getMapScene()	100.0 %	3	0	3
getScene()	100.0 %	3	0	3

LevelScene Class:



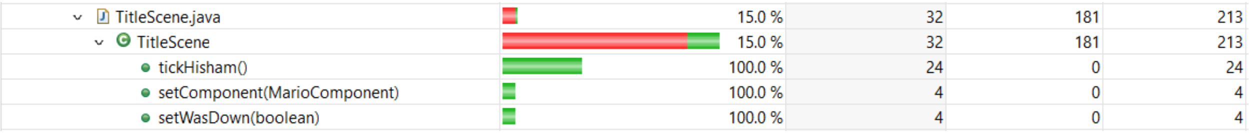
LevelScene.java	4.8 %	99	1,965	2,064
LevelScene	4.8 %	99	1,965	2,064
LevelScene(GraphicsConfiguration, MarioCompo	100.0 %	89	0	89
addSpriteFouad(Sprite)	100.0 %	6	0	6
setSpritesToAdd(List<Sprite>)	100.0 %	4	0	4

Scale2x Class:



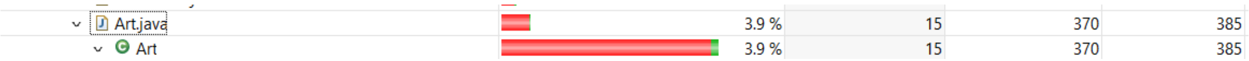
Scale2x.java	23.2 %	67	222	289
Scale2x	23.2 %	67	222	289
Scale2x(int, int)	100.0 %	67	0	67

Title Class:



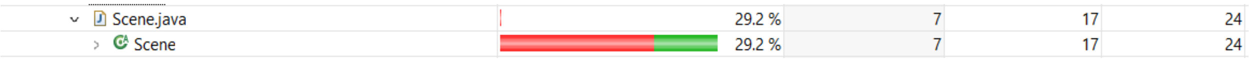
TitleScene.java	15.0 %	32	181	213
TitleScene	15.0 %	32	181	213
tickHisham()	100.0 %	24	0	24
setComponent(MarioComponent)	100.0 %	4	0	4
setWasDown(boolean)	100.0 %	4	0	4

Art Class:



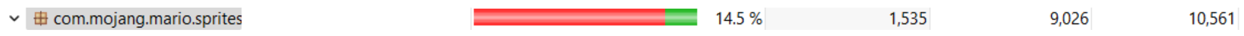
Art.java	3.9 %	15	370	385
Art	3.9 %	15	370	385

Scene Class:



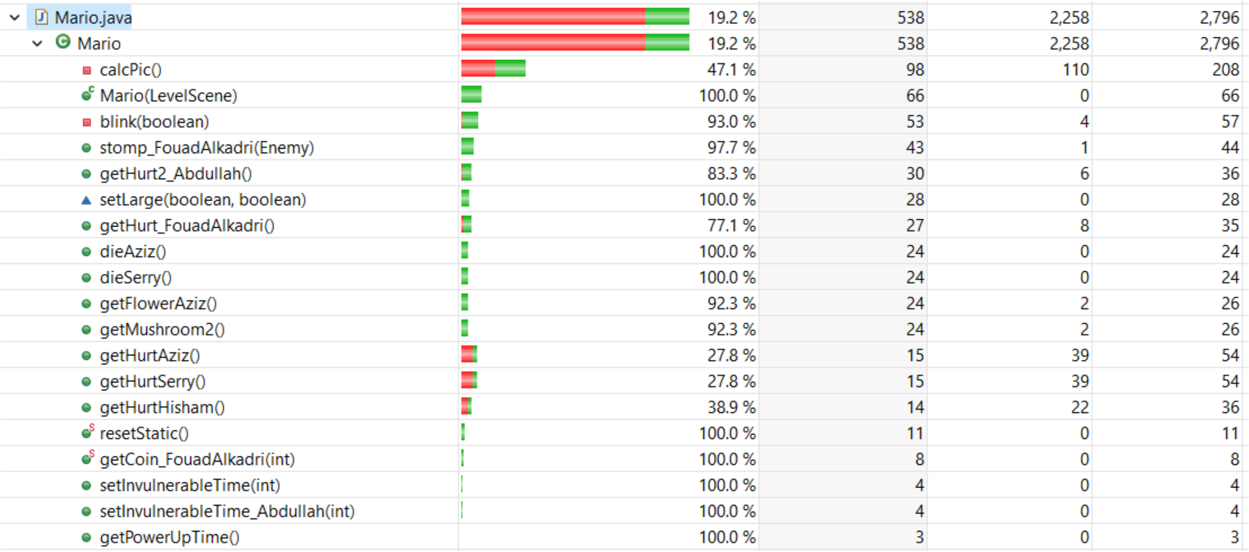
Scene.java	29.2 %	7	17	24
Scene	29.2 %	7	17	24

Package com.mojang.mario.sprites:



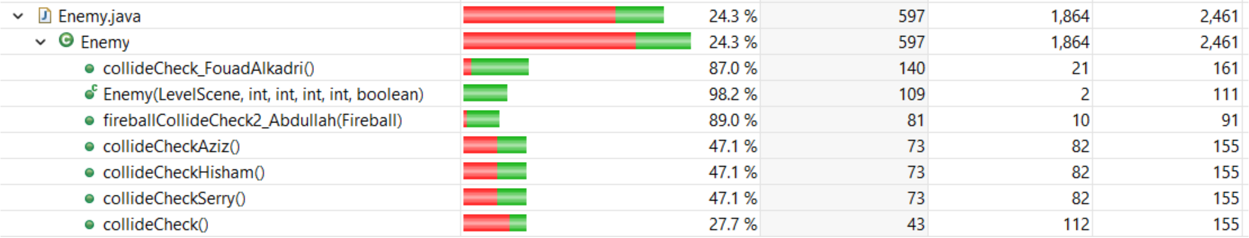
com.mojang.mario.sprites	14.5 %	1,535	9,026	10,561

Mario Class:



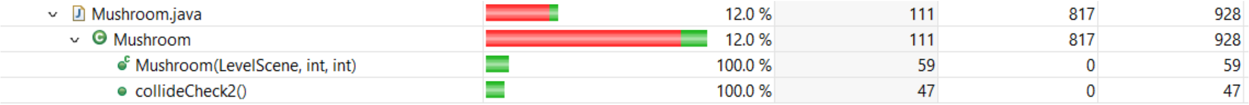
Mario.java	19.2 %	538	2,258	2,796
Mario	19.2 %	538	2,258	2,796
calcPic()	47.1 %	98	110	208
Mario(LevelScene)	100.0 %	66	0	66
blink(boolean)	93.0 %	53	4	57
stomp_FouadAlkadri(Enemy)	97.7 %	43	1	44
getHurt2_Abdullah()	83.3 %	30	6	36
setLarge(boolean, boolean)	100.0 %	28	0	28
getHurt_FouadAlkadri()	77.1 %	27	8	35
dieAziz()	100.0 %	24	0	24
dieSerry()	100.0 %	24	0	24
getFlowerAziz()	92.3 %	24	2	26
getMushroom2()	92.3 %	24	2	26
getHurtAziz()	27.8 %	15	39	54
getHurtSerry()	27.8 %	15	39	54
getHurtHisham()	38.9 %	14	22	36
resetStatic()	100.0 %	11	0	11
getCoin_FouadAlkadri(int)	100.0 %	8	0	8
setInvulnerableTime(int)	100.0 %	4	0	4
setInvulnerableTime_Abdullah(int)	100.0 %	4	0	4
getPowerUpTime()	100.0 %	3	0	3

Enemy Class:



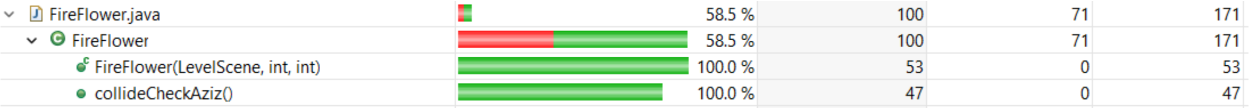
Enemy.java	24.3 %	597	1,864	2,461
Enemy	24.3 %	597	1,864	2,461
collideCheck_FouadAlkadri()	87.0 %	140	21	161
Enemy(LevelScene, int, int, int, int, boolean)	98.2 %	109	2	111
fireballCollideCheck2_Abdullah(Fireball)	89.0 %	81	10	91
collideCheckAziz()	47.1 %	73	82	155
collideCheckHisham()	47.1 %	73	82	155
collideCheckSerry()	47.1 %	73	82	155
collideCheck()	27.7 %	43	112	155

Mushroom Class:



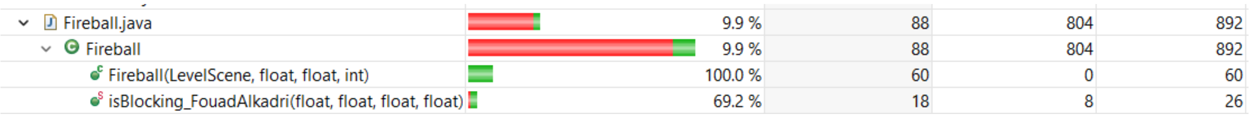
Mushroom.java	12.0 %	111	817	928
Mushroom	12.0 %	111	817	928
Mushroom(LevelScene, int, int)	100.0 %	59	0	59
collideCheck2()	100.0 %	47	0	47

FireFlower Class:



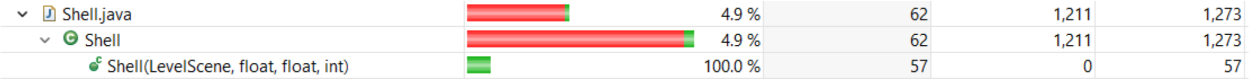
FireFlower.java	58.5 %	100	71	171
FireFlower	58.5 %	100	71	171
FireFlower(LevelScene, int, int)	100.0 %	53	0	53
collideCheckAziz()	100.0 %	47	0	47

Fireball Class:



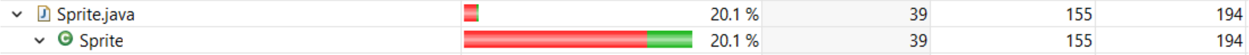
Fireball.java	9.9 %	88	804	892
Fireball	9.9 %	88	804	892
Fireball(LevelScene, float, float, int)	100.0 %	60	0	60
isBlocking_FouadAlkadri(float, float, float, float)	69.2 %	18	8	26

Shell Class:



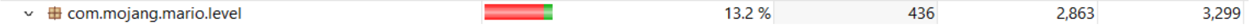
Shell.java	4.9 %	62	1,211	1,273
Shell	4.9 %	62	1,211	1,273
Shell(LevelScene, float, float, int)	100.0 %	57	0	57

Sprite Class:



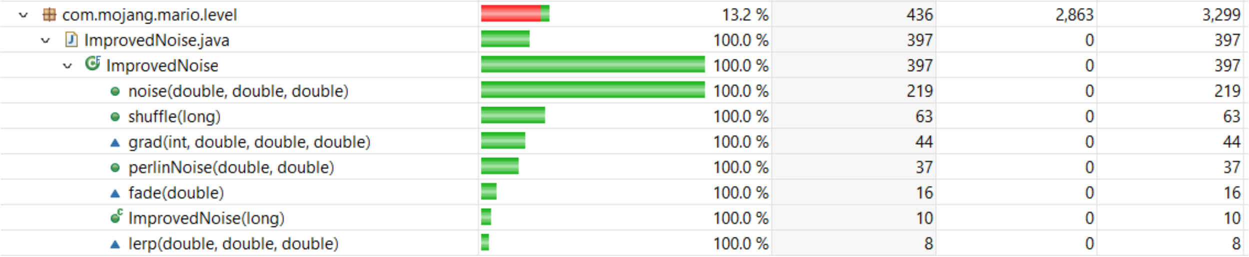
Sprite.java	20.1 %	39	155	194
Sprite	20.1 %	39	155	194

Package com.mojang.mario.level:



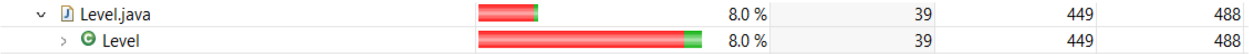
com.mojang.mario.level	13.2 %	436	2,863	3,299

ImprovedNoise Class:



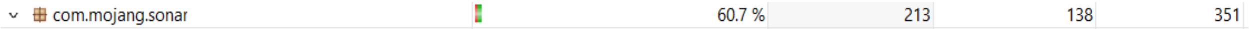
com.mojang.mario.level	13.2 %	436	2,863	3,299
ImprovedNoise.java	100.0 %	397	0	397
ImprovedNoise	100.0 %	397	0	397
noise(double, double, double)	100.0 %	219	0	219
shuffle(long)	100.0 %	63	0	63
grad(int, double, double, double)	100.0 %	44	0	44
perlinNoise(double, double)	100.0 %	37	0	37
fade(double)	100.0 %	16	0	16
ImprovedNoise(long)	100.0 %	10	0	10
lerp(double, double, double)	100.0 %	8	0	8

Level Class:



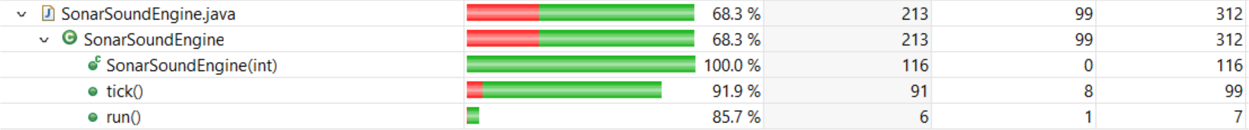
Level.java	8.0 %	39	449	488
Level	8.0 %	39	449	488

Package com.mojang.sonar:



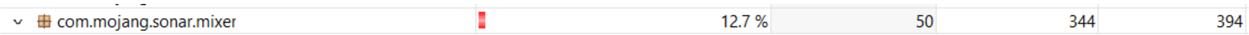
com.mojang.sonar	60.7 %	213	138	351

SonarSoundEngine Class:



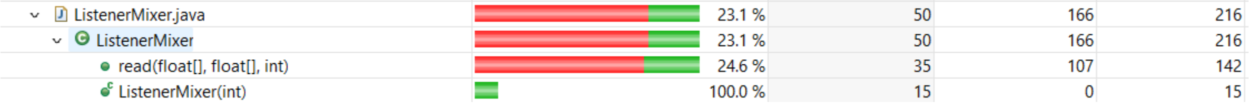
SonarSoundEngine.java	68.3 %	213	99	312
SonarSoundEngine	68.3 %	213	99	312
SonarSoundEngine(int)	100.0 %	116	0	116
tick()	91.9 %	91	8	99
run()	85.7 %	6	1	7

Package com.mojang.sonar.mixer:



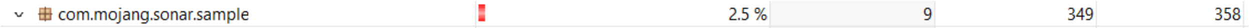
com.mojang.sonar.mixer	12.7 %	50	344	394

ListenerMixer Class;



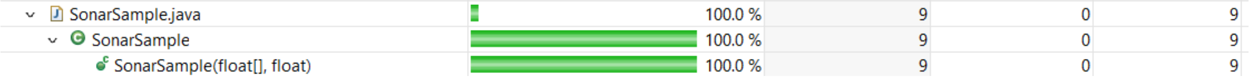
ListenerMixer.java	23.1 %	50	166	216
ListenerMixer	23.1 %	50	166	216
read(float[], float[], int)	24.6 %	35	107	142
ListenerMixer(int)	100.0 %	15	0	15

Package com.mojang.sonar.sample:



com.mojang.sonar.sample	2.5 %	9	349	358

SonarSample Class:



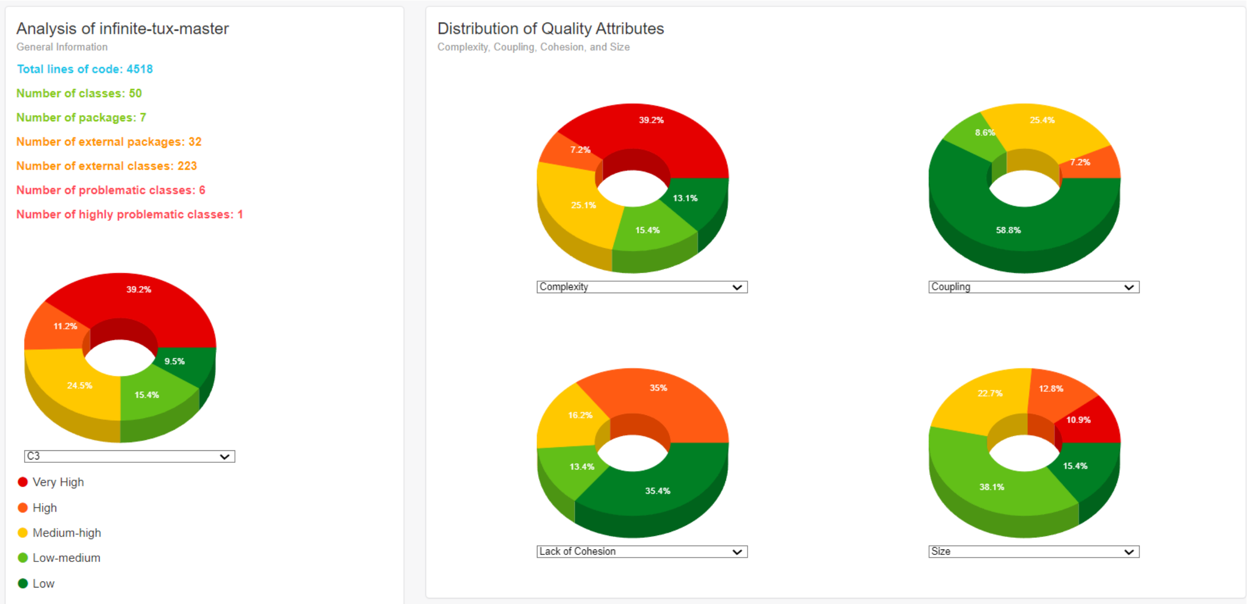
SonarSample.java	100.0 %	9	0	9
SonarSample	100.0 %	9	0	9
SonarSample(float[], float)	100.0 %	9	0	9

Black-Box Testing coverage:

Requirement	Covered by test cases
4.1.1	Integration test case: 20
4.2.1	Unit test cases: 22,23,24,25,26
4.3.8	Unit test case: 27
4.3.9	Unit test case: 28
4.3.10	Unit test case: 29
4.3.11	Unit test case: 30
4.3.13	Unit test cases: 1,2
	Integration test Cases: 1,2,3,4,5
4.3.14	Unit test cases: 1,2
	Integration test Cases: 1,2,3,4,5
4.3.15	Unit test cases: 3,4
	Integration test Cases: 6,7,8,9,10
4.3.16	Unit test cases: 3,4
	Integration test Cases: 6,7,8,9,10
4.3.17	Integration test Cases: 11,12,13,14,15

4.3.18	**Unit test cases: 5,6,7,8,9**
4.3.21	**Unit test cases: 10,11,12,13,14**
4.3.22	**Integration test cases: 16**
4.3.23	**Integration test cases: 17**
4.3.25	**Integration test cases: 18**
4.3.27	**Unit test cases: 16,17,18,19,20**

CodeMR Complexity Testing:



Analysis of infinite-tux-master

General Information

Total lines of code: 4518

Number of classes: 50

Number of packages: 7

Number of external packages: 32

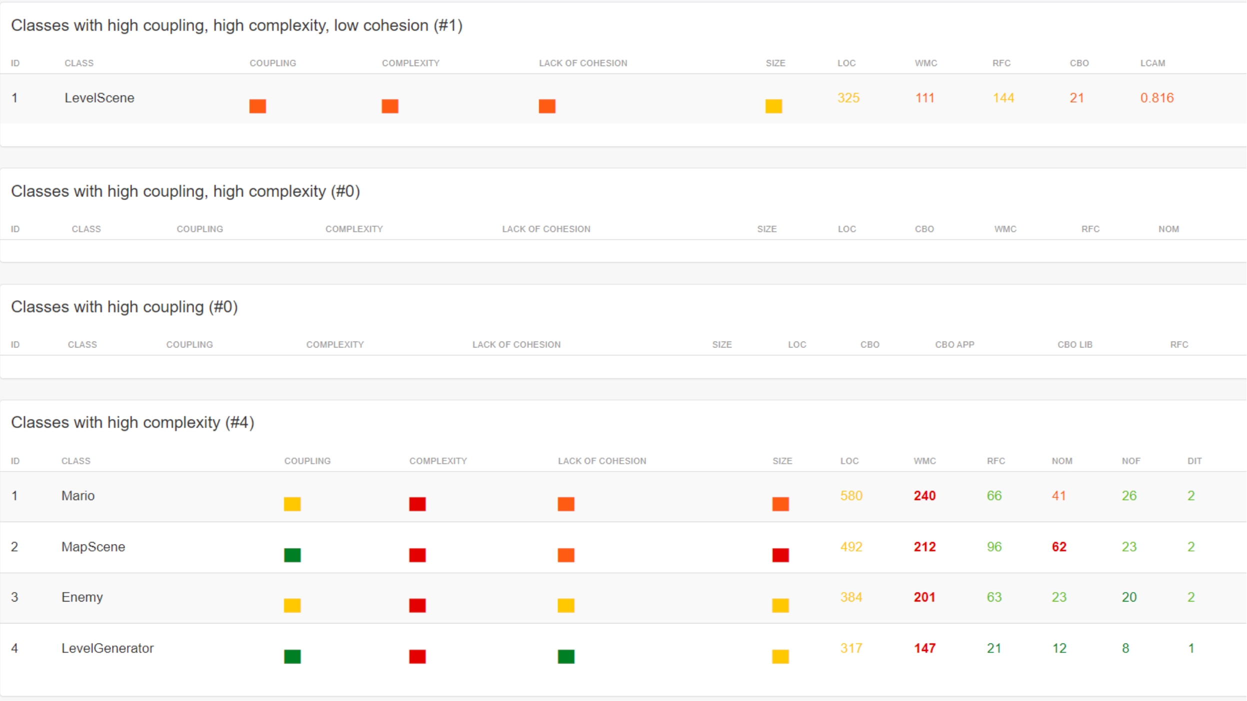
Number of external classes: 223

Number of problematic classes: 6

Number of highly problematic classes: 1

Distribution of Quality Attributes

Complexity, Coupling, Cohesion, and Size



Classes with high coupling, high complexity, low cohesion (#1)

ID	CLASS	COUPLING	COMPLEXITY	LACK OF COHESION	SIZE	LOC	WMC	RFC	CBO	LCAM
1	LevelScene	🟧	🟧	🟧	🟨	325	111	144	21	0.816

Classes with high coupling, high complexity (#0)

ID	CLASS	COUPLING	COMPLEXITY	LACK OF COHESION	SIZE	LOC	CBO	WMC	RFC	NOM

Classes with high coupling (#0)

ID	CLASS	COUPLING	COMPLEXITY	LACK OF COHESION	SIZE	LOC	CBO	CBO APP	CBO LIB	RFC

Classes with high complexity (#4)

ID	CLASS	COUPLING	COMPLEXITY	LACK OF COHESION	SIZE	LOC	WMC	RFC	NOM	NOF	DIT
1	Mario	🟨	🟥	🟧	🟧	580	240	66	41	26	2
2	MapScene	🟩	🟥	🟧	🟥	492	212	96	62	23	2
3	Enemy	🟨	🟥	🟨	🟨	384	201	63	23	20	2
4	LevelGenerator	🟩	🟥	🟩	🟨	317	147	21	12	8	1

More details are attached with the Submission HTML Webpage: