Report

500498752

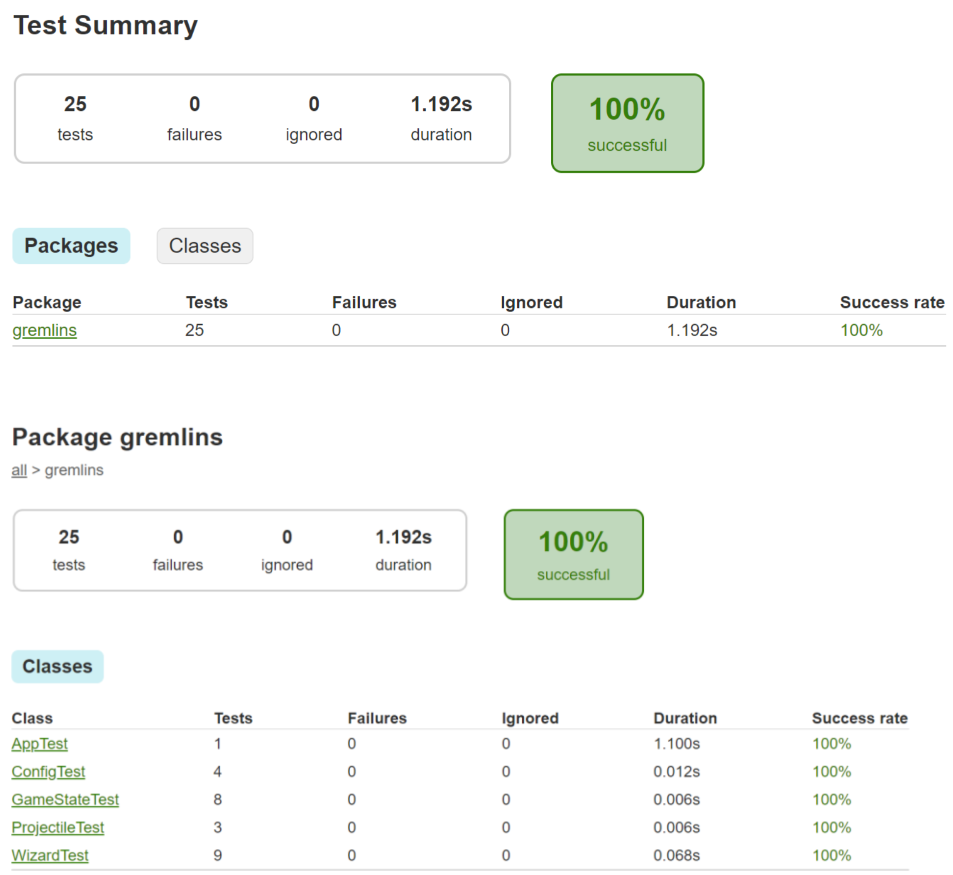
The design for the wizard game implements inheritance and encapsulation. The abstract class GameObject refers to all objects that are present in the game (such as tiles, player, enemy, etc…). Thus, it is the parent class of all game objects that follow. It is an abstract class as it will never be initialized. A game object has the attributes, x-axis, y-axis, and a sprite. It also has an abstract method tick, and a method draw (and various getter setter methods). This is because all game objects require these attributes and methods. The game object parent class has three direct subclasses, Tiles, MovingObject and PowerUp. Tiles and MovingObject are abstract classes as they do not need to be initialized. Tiles has four subclasses of its own, ExitDoor, Brick, Air and Stone. These represent the tiles on the map. Air and Stone classes do nothing but are present for functionality. The exitDoor class is used to determine if the player reaches an instance of that class. The brick class is used to determine if a fireball has hit an instance of that class and if so to perform the brick destruction animation. The moving object abstract class has attributes alive, collide, direction and x/y-velocity. This is because all moving objects require these attributes. It also has methods such as checkCollision and checkAlive which is used for collision detection. It has four subclasses, Gremlin, Wizard, Slime and Freeze. The wizard class is used for anything in the game that directly involves the wizard. This includes wizard movement, powerup, collision with slimes and gremlins, and shooting. The gremlin class is used for anything in the game that directly involves the gremlins. This includes gremlin movement, collision with fireball and freezeball, and shooting. The slime, fireball and freeze classes are used for the shooting of their respective projectiles.

As most of the game objects require references to other game objects, each subclass has attributes and methods that allow game objects to store references to other game objects.

The config class is used to read the config file. The loadlevel class is used to load the specific level onto the game. The states of the game are controlled through an enum class called gamestates. There are 4 gamestates, LevelOneState, LevelTwoState, WinGameState and LoseGameState. These are used in app to determine the state of the game.

The extension implemented for this game is another projectile which when colliding with the gremlins causes them to freeze for 3 seconds. The key that is pressed to shoot the projectile is the shift key. The powerup implemented for this game is a speed increase of the wizard. When the wizard lands on the powerup (a lightning sprite), the wizards speed doubles for a duration of 10 seconds.

The figures below show the gradle Test and gradle jacocoTestReport results. For some reason, running the tests commands work sometimes and do not work other times, so see below for reference.



Test Summary

25	0	0	1.192s
tests	failures	ignored	duration

100%
successful

Packages | Classes

Package	Tests	Failures	Ignored	Duration	Success rate
gremlins	25	0	0	1.192s	100%

Package gremlins

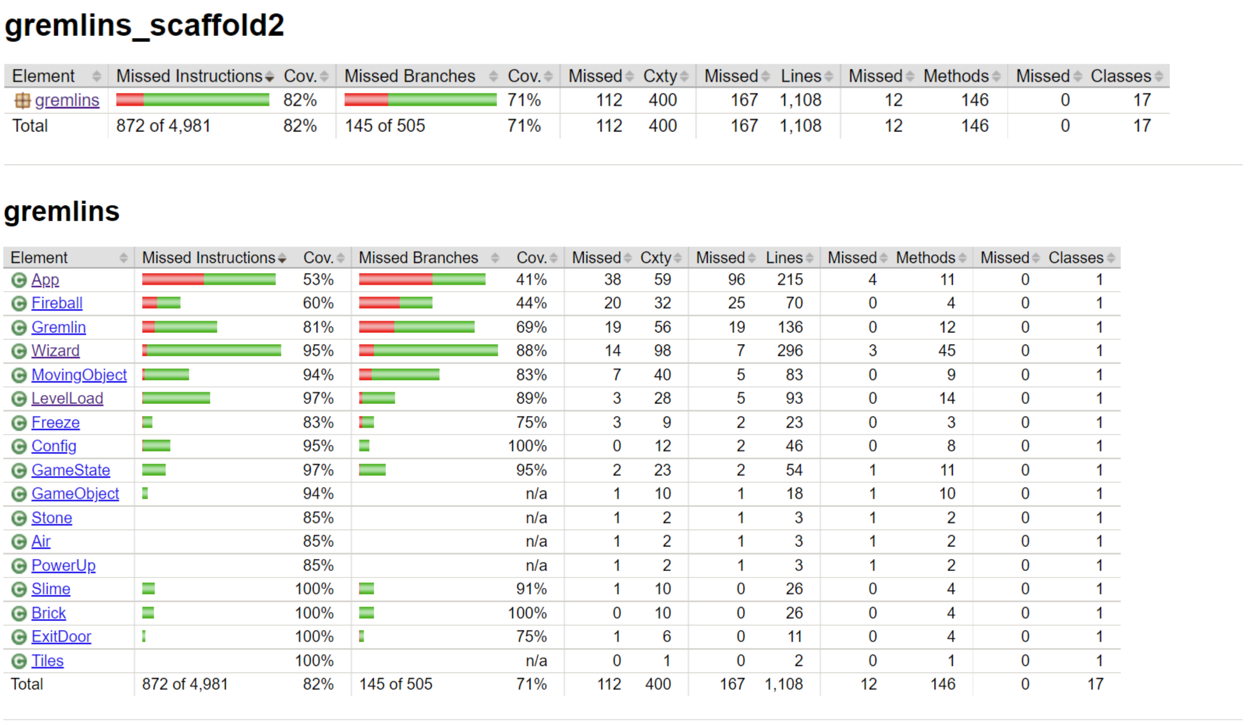
all > gremlins

25	0	0	1.192s
tests	failures	ignored	duration

100%
successful

Classes

Class	Tests	Failures	Ignored	Duration	Success rate
AppTest	1	0	0	1.100s	100%
ConfigTest	4	0	0	0.012s	100%
GameStateTest	8	0	0	0.006s	100%
ProjectileTest	3	0	0	0.006s	100%
WizardTest	9	0	0	0.068s	100%



gremlins_scaffold2

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods	Missed	Classes
gremlins		82%		71%	112	400	167	1,108	12	146	0	17
Total	872 of 4,981	82%	145 of 505	71%	112	400	167	1,108	12	146	0	17

gremlins

Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed	Cxty	Missed	Lines	Missed	Methods	Missed	Classes
App		53%		41%	38	59	96	215	4	11	0	1
Fireball		60%		44%	20	32	25	70	0	4	0	1
Gremlin		81%		69%	19	56	19	136	0	12	0	1
Wizard		95%		88%	14	98	7	296	3	45	0	1
MovingObject		94%		83%	7	40	5	83	0	9	0	1
LevelLoad		97%		89%	3	28	5	93	0	14	0	1
Freeze		83%		75%	3	9	2	23	0	3	0	1
Config		95%		100%	0	12	2	46	0	8	0	1
GameState		97%		95%	2	23	2	54	1	11	0	1
GameObject		94%		n/a	1	10	1	18	1	10	0	1
Stone		85%		n/a	1	2	1	3	1	2	0	1
Air		85%		n/a	1	2	1	3	1	2	0	1
PowerUp		85%		n/a	1	2	1	3	1	2	0	1
Slime		100%		91%	1	10	0	26	0	4	0	1
Brick		100%		100%	0	10	0	26	0	4	0	1
ExitDoor		100%		75%	1	6	0	11	0	4	0	1
Tiles		100%		n/a	0	1	0	2	0	1	0	1
Total	872 of 4,981	82%	145 of 505	71%	112	400	167	1,108	12	146	0	17