

Lab 1 Report

Name: 許哲維

Student ID: 111598066

Date: 2023.03.08

1 Test Plan

1.1 Test requirements

The Lab 1 requires to (1) select **20 methods** from **6 classes** of the SUT (GeoProject), (2) design Unit test cases based on the experience or intuition for the selected methods, (3) develop test scripts to implement the test cases, (4) execute the test script on the selected methods, and (5) report the test results.

In particular, based on the statement coverage criterion, the **test requirements** for Lab 1 are to design test cases for each selected method so that *“each statement of the method will be covered by at least one test case and the minimum statement coverage is **50%**”*.

1.2 Strategy

To satisfy the test requirements listed in Section 1, a proposed strategy is to

- (1) select those public methods that are easy to understand and have primitive types of input and output parameters (if possible).
- (2) set the objective of the minimum statement coverage to be 50% initially and (if necessary) adjust the objective based on the time available.
- (3) learn the necessary skills and tools as soon as possible.
- (4) design the test cases for those selected methods by considering
 - i. the possible **valid values** and **combinations** of the input parameters.
 - ii. the **boundary values** of the input parameters.

1.3 Test activities

To implement the proposed strategy, the following activities are planned to perform.

No.	Activity Name	Plan hours	Schedule Date
1	Study GeoProject	3	2023.03.01
2	Learn JUnit	3	2023.03.02
3	Design test cases for the selected methods	3	2023.03.02
4	Implement test cases	4.5	2023.03.06
5	Perform test	4.5	2023.03.07
6	Complete Lab1 report	2	2023.03.08

1.4 Success criteria

All test cases designed for the selected methods must pass (or "85% of all test cases must pass) and the statement coverage should have achieved at least 50%.

2 Test Design

To fulfill the test requirements listed in section 1.1, the following methods are selected and corresponding test cases are designed.

No .	Class	Method	Test Objective	Inputs	Expected Outputs
1	Base32	encodeBase32()		75324, 4	29jw
2	Base32	encodeBase32()		75324, -4	-29jw
3	Base32	encodeBase32()		4	000000000004
4	Base32	decodeBase32()		29jw	75324
5	Base32	getCharIndex()		0	0
6	Base32	padLeftWithZerosToLength()		Lucy, 5	0Lucy
7	Base32	padLeftWithZerosToLength()		JenJen, 6	JenJen
8	Coverage	getHashes()	Coverage		Katayama, Jenjen, not, Chinchin.
9	Coverage	getRatio()	Coverage		4.0
10	Coverage	getHashLength()	Coverage		9
11	Coverage	toString()	Coverage		Coverage [hashes=[Chinchin., not, Katayama, Jenjen], ratio=4.0]
12	CoverageLongs	getHashLength()	CoverageLongs		3
13	CoverageLongs	getHashLength()	CoverageLongs		0
14	CoverageLongs	toString()	CoverageLongs		"Coverage [hashes="+c.getHashes()+", ratio="+c.getRatio()+"]"
15	CoverageLongs	getCount()	CoverageLongs		1

16	Direction	opposite()		Direction.RIGHT	Direction.LEFT
17	Direction	opposite()		Direction.LEFT	Direction.RIGHT
18	Direction	opposite()		Direction.TOP	Direction.BOTTOM
19	Direction	opposite()		Direction.BOTTOM	Direction.TOP
20	GeoHash	adjacentHash()		"29jw", Direction.BOTTOM, 1	"29jt"
21	GeoHash	neighbours()		"29jw"	"29jq"
22	GeoHash	hashLengthToCoverBoundingBox()		2,2,1,1	2
23	GeoHash	coverBoundingBoxLongs()		0.1,0.1,0.1,0.1,1	1
24	LatLong	getLat()	LatLong		6
25	LatLong	toString()	LatLong		"LatLong [lat="+lat+"lon="+lon+""]"

3 Test Implementation

The design of test cases specified in Section 2 was implemented using JUnit 4. The test scripts of 3 selected test cases are given below. The rest of test script implementations can be found in the [link](#) (or JUnit files).

No.	Test method	Source code
1	encodeBase32()	<pre> no usages 許哲維 +1 @Test public void encodeBase32() throws Exception { String encode = Base32.encodeBase32(75324, length: 4); assertEquals(expected: "29jw", encode); encode = Base32.encodeBase32(-75324, length: 4); assertEquals(expected: "-29jw", encode); encode = Base32.encodeBase32(4); assertEquals(expected: "000000000004", encode); } </pre>
2	TesttoString()	<pre> no usages 許哲維 @Test public void TesttoString() throws Exception { Set<String> word = new HashSet<String>(); word.add("Katayama"); word.add("Jenjen"); word.add("not"); word.add("Chinchin."); Coverage c = new Coverage(word, ratio: 4); String line = c.toString(); assertEquals(expected: "Coverage [hashes=[Chinchin., not, Katayama, Jenjen], ratio=4.0]", line); } </pre>

3	opposite()	<pre> no usages 許哲維 @Test public void opposite() throws Exception { //Direction.BOTTOM Direction direction = Direction.RIGHT; direction = direction.opposite(); assertEquals(Direction.LEFT, direction); direction = direction.opposite(); assertEquals(Direction.RIGHT, direction); direction = Direction.TOP; direction = direction.opposite(); assertEquals(Direction.BOTTOM, direction); direction = direction.opposite(); assertEquals(Direction.TOP, direction); } </pre>
---	------------	--

4 Test Results

4.1 JUnit test result snapshot

Test Results	24 ms
> ✓ com.github.davidmoten.geo.Base32Test	4 ms
> ✓ com.github.davidmoten.geo.CoverageLongsTest	1 ms
> ✓ com.github.davidmoten.geo.CoverageTest	9 ms
> ✓ com.github.davidmoten.geo.DirectionTest	0 ms
> ✓ com.github.davidmoten.geo.GeoHashTest	10 ms
> ✓ com.github.davidmoten.geo.LatLongTest	0 ms

Test Summary

21

tests

0

failures

0

ignored

0.062s

duration

100%

successful

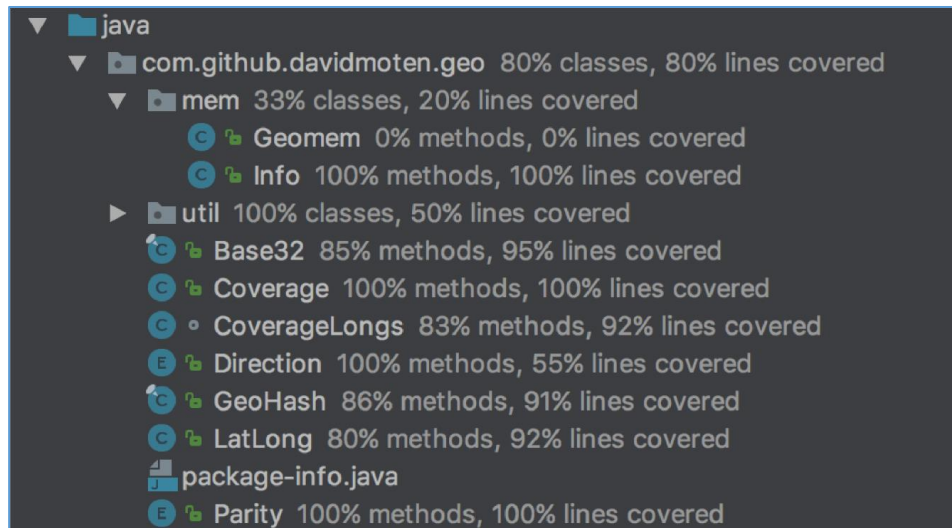
Packages

Classes

Package	Tests	Failures	Ignored	Duration	Success rate
com.github.davidmoten.geo	21	0	0	0.062s	100%

4.2 Code coverage snapshot

- Coverage of each selected method

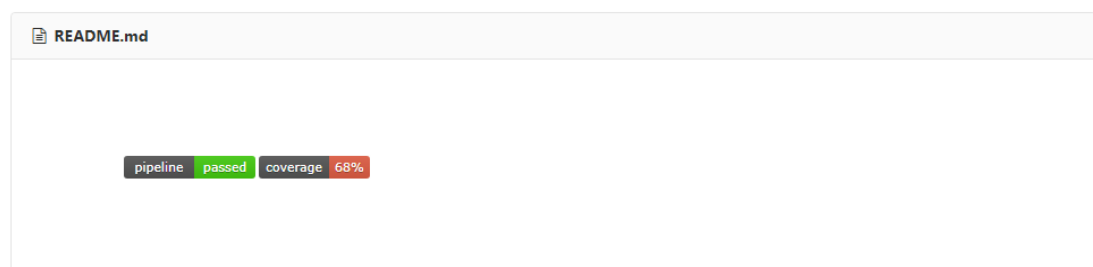


● Total coverage

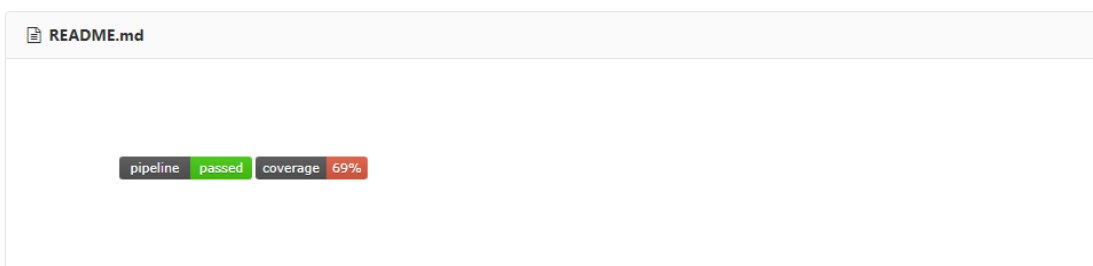
geo										
geo										
Element	Missed Instructions	Cov.	Missed Branches	Cov.	Missed Cxty	Missed Lines	Missed Methods	Missed Classes		
com.github.davidmoten.geo	<div><div></div></div>	82%	<div><div></div></div>	70%	51 149	57 348	15 68	0 10		
com.github.davidmoten.geo.mem	<div><div></div></div>	0%	<div><div></div></div>	0%	30 30	61 61	20 20	3 3		
com.github.davidmoten.geo.util	<div><div></div></div>	68%	<div><div></div></div>	75%	1 4	1 6	0 2	0 1		
Total	675 of 2,326	70%	69 of 186	62%	82 183	119 415	35 90	3 14		

4.3 CI result snapshot (3 iterations for CI)

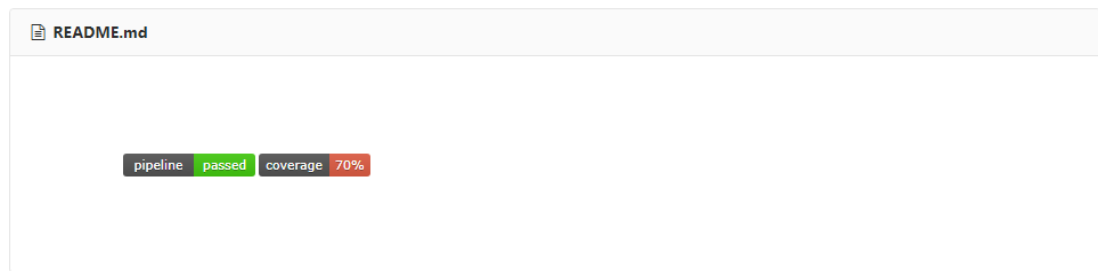
● CI#1



● CI#2



● CI#3



● CI Pipeline

GeoProject

- Overview
- Repository
- Issues
- Merge Requests
- CI / CD
- Pipelines**
- Jobs
- Schedules
- Environments
- Charts
- Cluster

GeoProject Pipelines

Run Pipeline CI Link

Status	Pipeline	Commit	Stages	Duration	Time
passed	#3840 by Lab 1 c13	P master -> 35747d3e	✓ ✓	00:01:15	2 minutes ago
passed	#3839 by Lab 1 c12	P master -> fc661b44	✓ ✓	00:01:18	5 minutes ago
passed	#3838 by Lab 1 c1	P master -> a0e854d2	✓ ✓	00:01:24	21 minutes ago
passed	#3650 by Update README.md	P master -> 0e4d39ef	✓ ✓	00:04:26	a week ago

5 Summary

In Lab 1, **25** test cases have been designed and implemented using JUnit. The test is conducted in **3** CI and the execution results of the **25** test methods are **all passed**. The total statement coverage of the test is **70%**. Thus, the test requirements described in Section 1 are satisfied.