ZiFeng Xu (001045590)

Program Structures & Algorithms

Fall 2021

Assignment No. 1

- **Task (List down the tasks performed in the Assignment)**

  Radom Walk

- **Relationship Conclusion:** $d = \sqrt{n}$

- **$Evidence to support the conclusion:**

1. **Output (Snapshot of Code output in the terminal)**

```
<terminated> RandomWalk [Java Application] /Library/Java/JavaVirtualMachines/jdk1.8.0_162.jdk/Contents/Home/bin/j
We walked 25 steps in 60 experiments and the mean distance is  4.4156944084039695
We walked 100 steps in 60 experiments and the mean distance is  8.941403827977977
We walked 1600 steps in 60 experiments and the mean distance is  35.72083205934917
We walked 160000 steps in 60 experiments and the mean distance is  384.60981318580787
We walked 250000 steps in 60 experiments and the mean distance is  430.45897689985526
We walked 1000000 steps in 60 experiments and the mean distance is  962.2042879063238
```

2. **Graphical Representation(Observations from experiments should be tabulated and analyzed by plotting graphs(usually in excel) to arrive on the relationship conclusion)**

$$d_1 = \sqrt{(x_1 - 0)^2 + (y_1 - 0)^2} d^2$$

$$= \frac{1}{p}\left(\sum_{i=1}^{p} (x_{i1} + x_{i2} + x_{i3} + \cdots + x_{in})^2 + (y_{i1} + y_{i2} + y_{i3} + \cdots + y_{in})^2\right)$$

$$d1 = \frac{1}{p}\sum_{i=1}^{p} ({x_{i1}}^2 + x_{i1}x_{i2} + x_{i1}x_{i3} + \cdots + {x_{i2}}^2 + x_{i1}x_{i2} + \cdots {x_{in}}^2 + {y_{i1}}^2 + y_{i1}y_{i2} + y_{i1}y_{i3} + \cdots + {y_{i2}}^2 + y_{i1}y_{i2} + \cdots {y_{in}}^2)$$

$$d^2 = \frac{1}{p}\sum_{i=1}^{p} ({x_{i1}}^2 + x_{i1}x_{i2} + x_{i1}x_{i3} + \cdots + {x_{i2}}^2 + x_{i1}x_{i2} + \cdots {x_{in}}^2 + {y_{i1}}^2 + y_{i1}y_{i2} + y_{i1}y_{i3} + \cdots + {y_{i2}}^2 + y_{i1}y_{i2} + \cdots {y_{in}}^2)$$

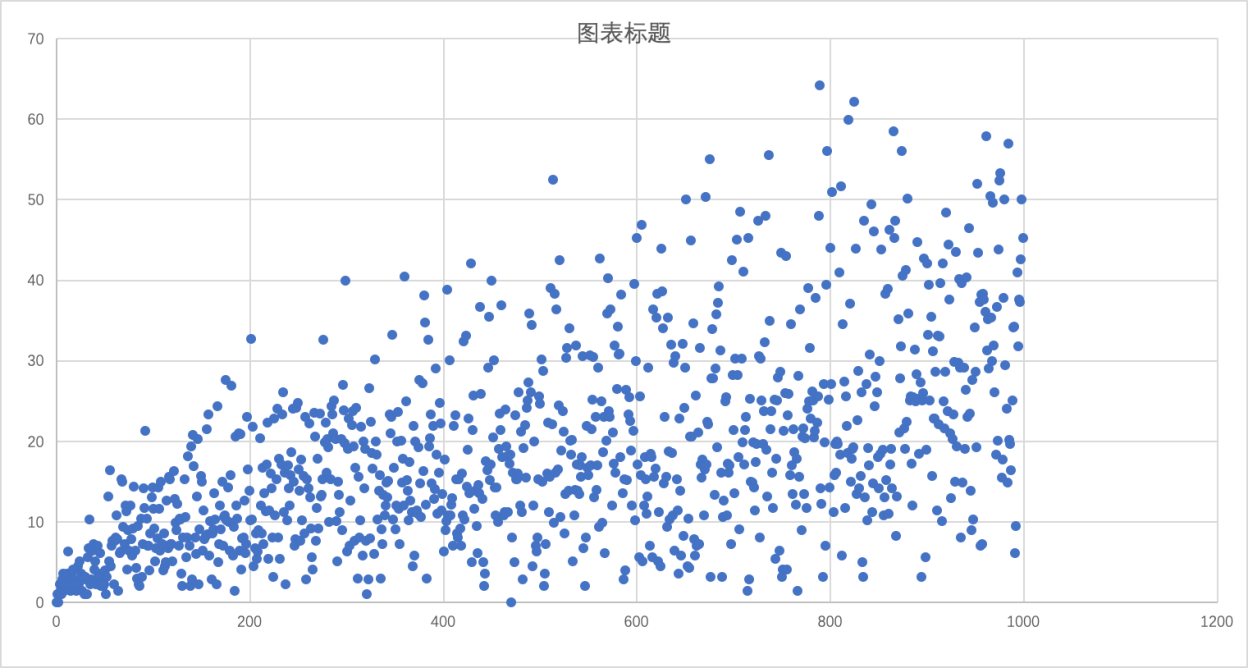
These X and Y values are both positive and negative uniformly distributed, so when the number of experiments p is sufficiently large, except for those square terms, everything else is cancelled out.

$$d^2 = \frac{1}{p}\sum_{i=1}^{p} ({x_{i1}}^2 + {y_{i1}}^2 + {x_{i2}}^2 + {y_{i2}}^2 + \cdots {x_{in}}^2 + {y_{in}}^2)$$

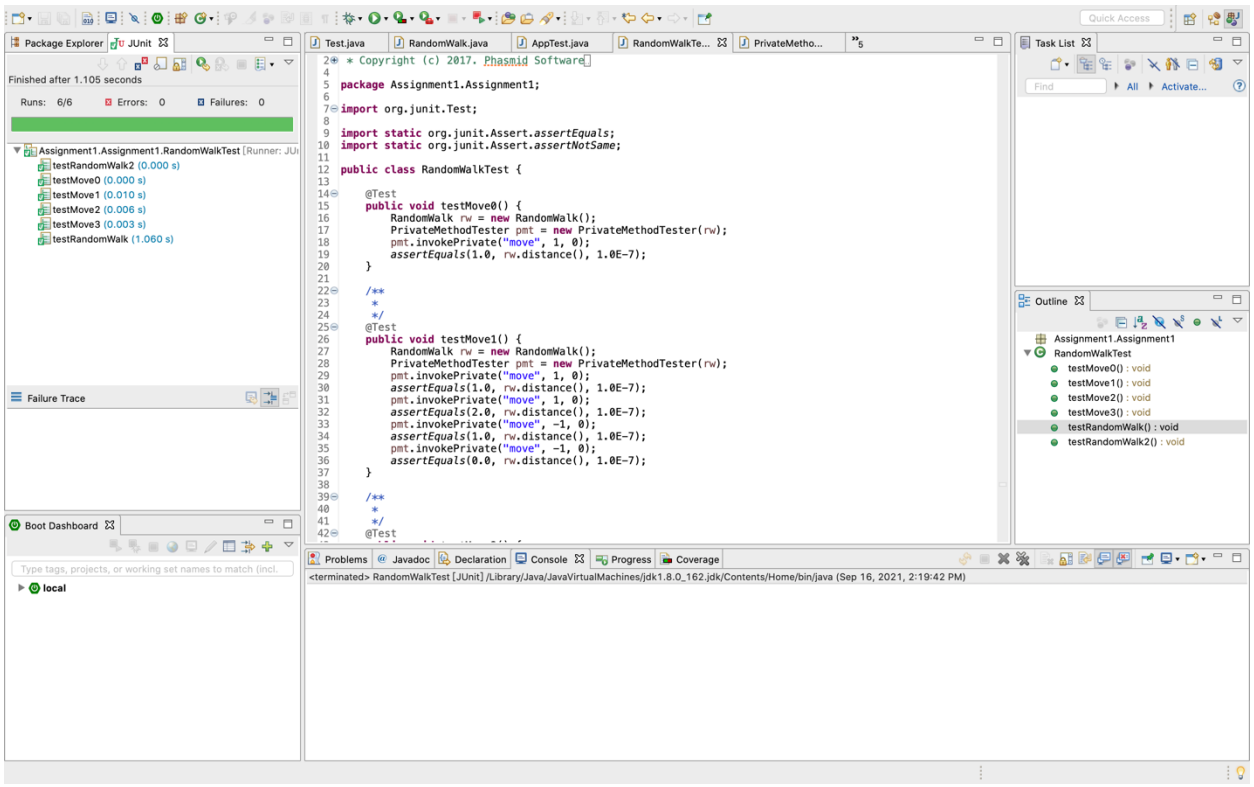
$$d^2 = \frac{1}{p}\sum_{i=1}^{p} \quad (n)$$

$$d^2 = n$$

$$d = \sqrt{n}$$



- **Unit tests result:(Snapshot of successful unit test run)**


Package Explorer
JUnit
Finished after 1.105 seconds
Runs: 6/6
Errors: 0
Failures: 0
Assignment1.Assignment1.RandomWalkTest [Runner: JU
testRandomWalk2 (0.000 s)
testMove0 (0.000 s)
testMove1 (0.010 s)
testMove2 (0.006 s)
testMove3 (0.003 s)
testRandomWalk (1.060 s)
Failure Trace
Boot Dashboard
Type tags, projects, or working set names to match (incl.
local
Test.java
RandomWalk.java
AppTest.java
RandomWalkTe...
PrivateMetho...
5
2 * Copyright (c) 2017. Phasmid Software
4
5 package Assignment1.Assignment1;
6
7 import org.junit.Test;
8
9 import static org.junit.Assert.assertEquals;
10 import static org.junit.Assert.assertNotSame;
11
12 public class RandomWalkTest {
13
14 @Test
15 public void testMove0() {
16 RandomWalk rw = new RandomWalk();
17 PrivateMethodTester pmt = new PrivateMethodTester(rw);
18 pmt.invokePrivate("move", 1, 0);
19 assertEquals(1.0, rw.distance(), 1.0E-7);
20 }
21
22 /**
23 *
24 */
25 @Test
26 public void testMove1() {
27 RandomWalk rw = new RandomWalk();
28 PrivateMethodTester pmt = new PrivateMethodTester(rw);
29 pmt.invokePrivate("move", 1, 0);
30 assertEquals(1.0, rw.distance(), 1.0E-7);
31 pmt.invokePrivate("move", 1, 0);
32 assertEquals(2.0, rw.distance(), 1.0E-7);
33 pmt.invokePrivate("move", -1, 0);
34 assertEquals(1.0, rw.distance(), 1.0E-7);
35 pmt.invokePrivate("move", -1, 0);
36 assertEquals(0.0, rw.distance(), 1.0E-7);
37 }
38
39 /**
40 *
41 */
42 @Test
Task List
Find
All
Activate...
Outline
Assignment1.Assignment1
RandomWalkTest
testMove0() : void
testMove1() : void
testMove2() : void
testMove3() : void
testRandomWalk() : void
testRandomWalk2() : void
Problems
Javadoc
Declaration
Console
Progress
Coverage
<terminated> RandomWalkTest [JUnit] /Library/Java/JavaVirtualMachines/jdk1.8.0_162.jdk/Contents/Home/bin/java (Sep 16, 2021, 2:19:42 PM)