CONCLUSION

The relationship between the number of steps (m) and the Euclidean distance of the man from the lamp post (d) is as follows:

$$d = O(\sqrt{m})$$

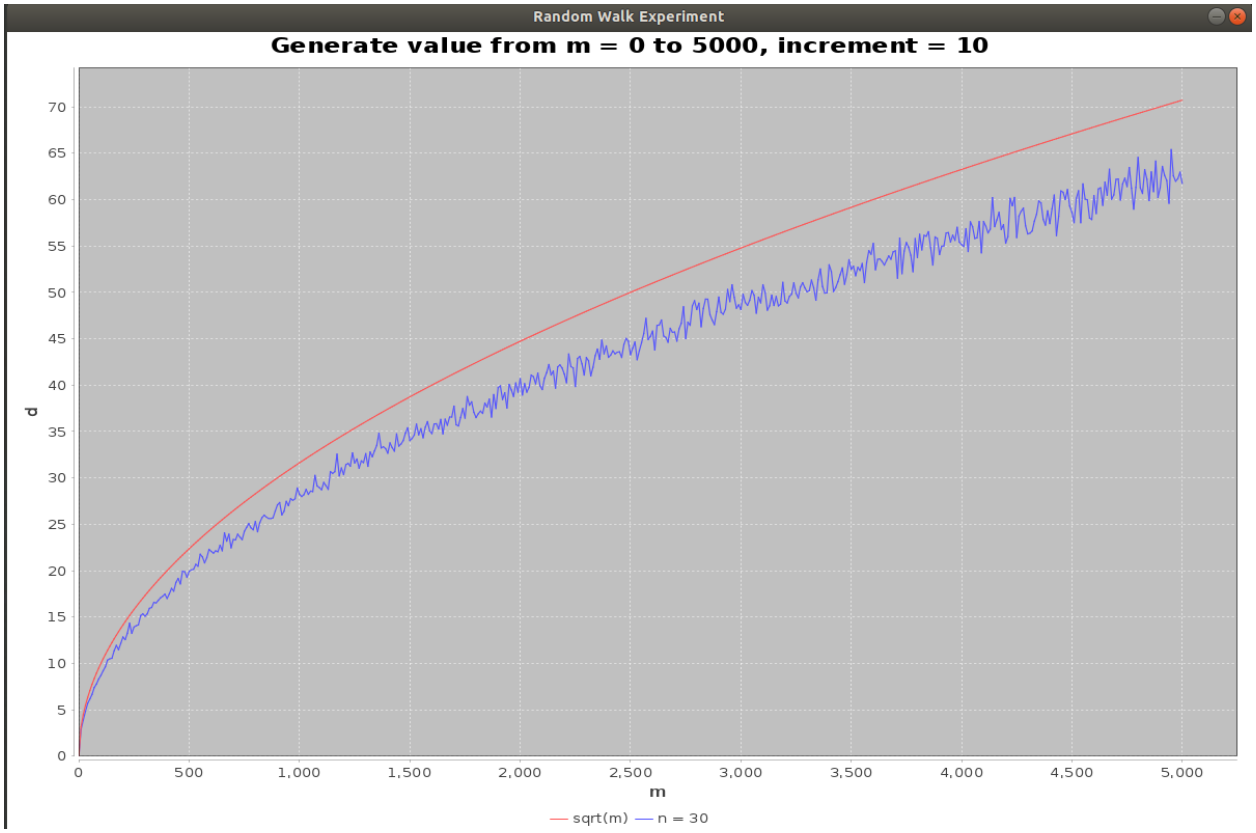
To support the evidence of the conclusion changes were made in main code to plot the data over a large range of values for multiple simulations.

EVIDENCE

Changes were made in the main code of RandomWalk class to run against multiple values for ‘m’, ‘n’ to run for a minimum number of simulations (all of these values can be passed on the command line). After running the experiment for the number of simulations for a given value of ‘n’ and ‘m’ the average value of distance obtained from all the Random Walk experiments was considered as ‘d’.

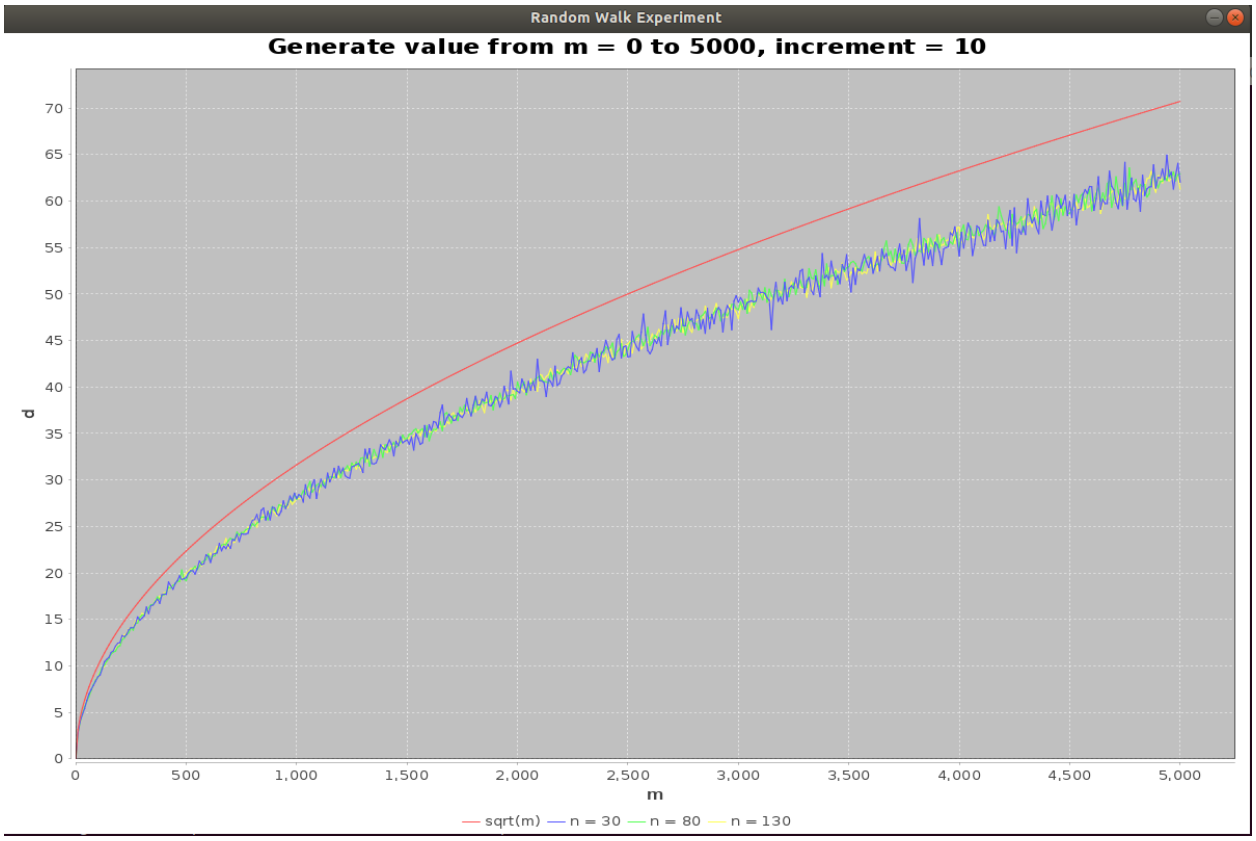
Graph plotted for multiple values of ‘d’ against ‘m’ shows that the shape of the curve is almost like that of $d = c\sqrt{m}$ where ‘c’ is any arbitrary constant. Each experiment was run for 20 times and the mean value of ‘d’ was plotted against ‘n’. Another curve was plotted for $d = \sqrt{m}$ to compare the graph with the random data generated.

Command Line Argument: “0,5000,10 30 20”

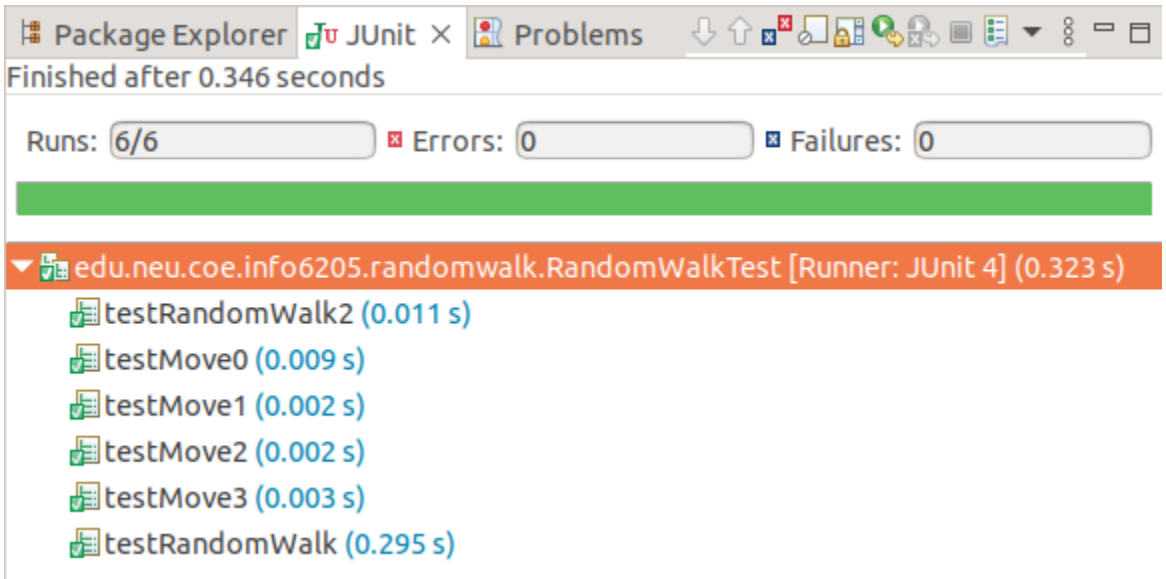


Also multiple values of total experiments run, ‘n’ was also taken to see if that affects the relationship between ‘d’ and ‘m’ and it can be seen that the overall shape of the curve remains same.

Command Line Argument: “0,5000,10 30,130,50 20”



UNIT TESTS



SOURCE CODE CHANGES

Apart from filling in the missing code for the various methods in the experiment. The main code was also changed in RandomWalk class to allow running the experiment for multiple values and then plotting them on a graph without running the program repeatedly.

Another RandomWalkGraphPlotter was created to handle the logic for plotting the data on a chart.

Command Line Arguments can take following forms

1. If Experiment is to be performed for only one value of 'n'
   <start-m>,<end-m>,<increment-m> <n> <simulations>

2. If Experiment is to be performed for multiple values of 'n'
   <start-m>,<end-m>,<increment-m> <start-n>,<end-n>,<increment-n> <simulations>

Explanation of the arguments:

- start-m: Starting value of 'm'
- end-m: Ending value of 'm'
- increment-m: Value to increase 'm' for next iteration

- n: Number of experiments to run (valid only when running for one value of 'n')
- start-n: Starting value of 'n'
- end-n: Ending value of 'n'
- increment-n: Value to increase 'n' for next iteration

- simulations: Number of times to run the experiment for a given value of 'm' and 'n'

NOTE: You can also run the code on the terminal using maven. (**Recommended to run the code on java version 18.0.2.1**). The parameter passed in *exec.args* will act as the command line arguments.

```
mvn clean compile install org.codehaus.mojo:exec-maven-plugin:1.5.0:java
-Dexec.mainClass="edu.neu.coe.info6205.randomwalk.RandomWalk"
-Dexec.args="0,5000,10 30 20" -DskipTests
```

IMPORTANT: If you choose to run the code directly from your IDE it is important that you update maven libraries on your system. For different IDEs the process is different. It can be done directly from the command line by using the following command without depending on how to the IDE updates the maven libraries**:**

```
mvn dependency:resolve
```

LIBRARIES ADDED

Following dependencies were added in pom.xml to use JFreeChart library to plot the graph

```
<dependency>
	<groupId>org.jfree</groupId>
	<artifactId>jfreechart</artifactId>
	<version>1.5.0</version>
</dependency>

<dependency>
	<groupId>org.jfree</groupId>
	<artifactId>jcommon</artifactId>
	<version>1.0.23</version>
</dependency>
```