

Program Structures and Algorithms
Spring 2024

NAME: Neha Devarapalli

NUID: 002883137

GITHUB LINK: <https://github.com/nehadevarapalli/INFO6205>

Task: To implement the code for the Random Walk experiment and deduce the relationship between the mean distance (d) and the number of steps taken (m).

Relationship Conclusion: $d \approx \sqrt{m}$

1. With each step, the man can move only +1 or -1 in the 'x' or 'y' direction.
2. Since there is no bias in the walk, the average value of each individual displacement in ' $\langle x_i \rangle$ ' and ' $\langle y_i \rangle$ ' is equal to 0.
3. Also since there is no correlation between the steps (stochastic), the dot product of 'x' and 'y' is 0 i.e.,

$$\sum x_i x_j = 0 \text{ and } \sum y_i y_j = 0$$

With these assumptions we will calculate the mean displacement of a walk after 'm' steps and then the root mean square displacement.

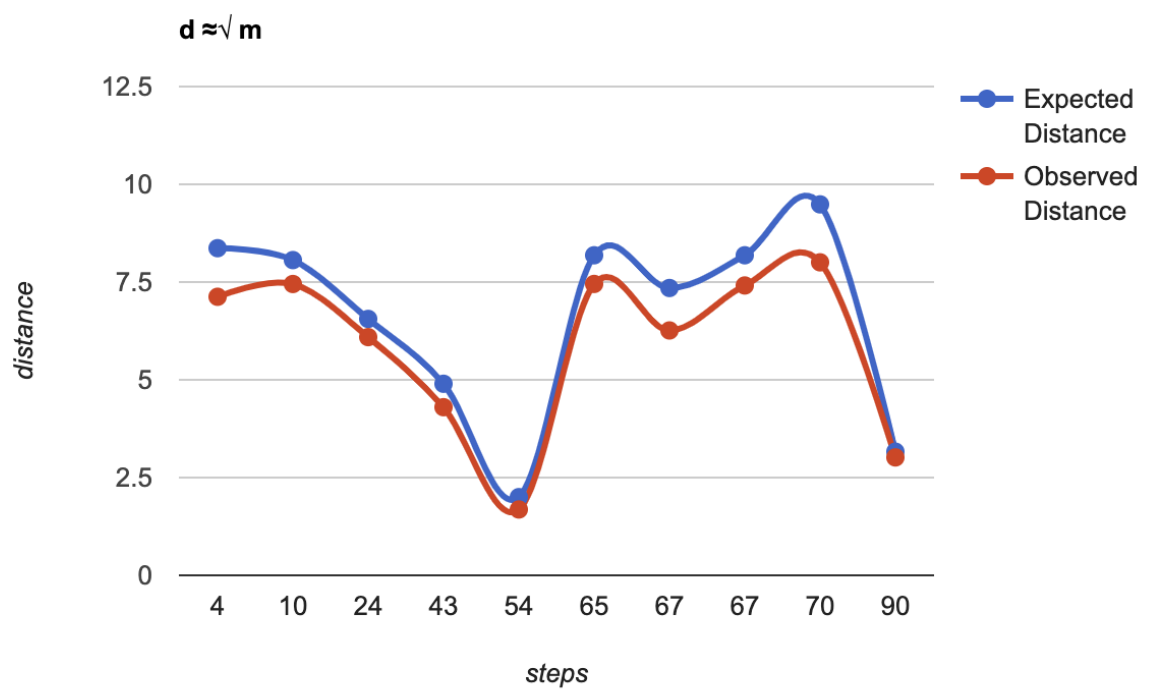
1. The mean displacement can be calculated by calculating the mean / average of the sum of individual displacements ($\sum x$ and $\sum y$ from 1 to m). But we know that $\langle x_i \rangle$ and $\langle y_i \rangle$ are 0 and hence, the mean displacement also is 0.
2. Now, to calculate the root mean square displacement, we have to take the sum of the displacements, square them and calculate the average of each ($\langle (\sum x)^2 \rangle$ and $\langle (\sum y)^2 \rangle$ from 1 to m. When we perform the above calculation there will be terms like:
 $\langle \sum x_i^2 + \sum x_i x_j \rangle$ and $\langle \sum y_i^2 + \sum y_i y_j \rangle$ where $i \neq j$ and $i, j = 1$ to m
Since there are no correlations between the steps, $\sum x_i x_j$ and $\sum y_i y_j$ will be 0.
And the terms $\sum x_i^2$ and $\sum y_i^2$ will have 'm' such terms of the length equal to 1 each and hence the average will be 'm'. SO, $d^2 \approx m$
Hence, the root mean square distance (d) $\approx \sqrt{m}$

Evidence to support that conclusion:

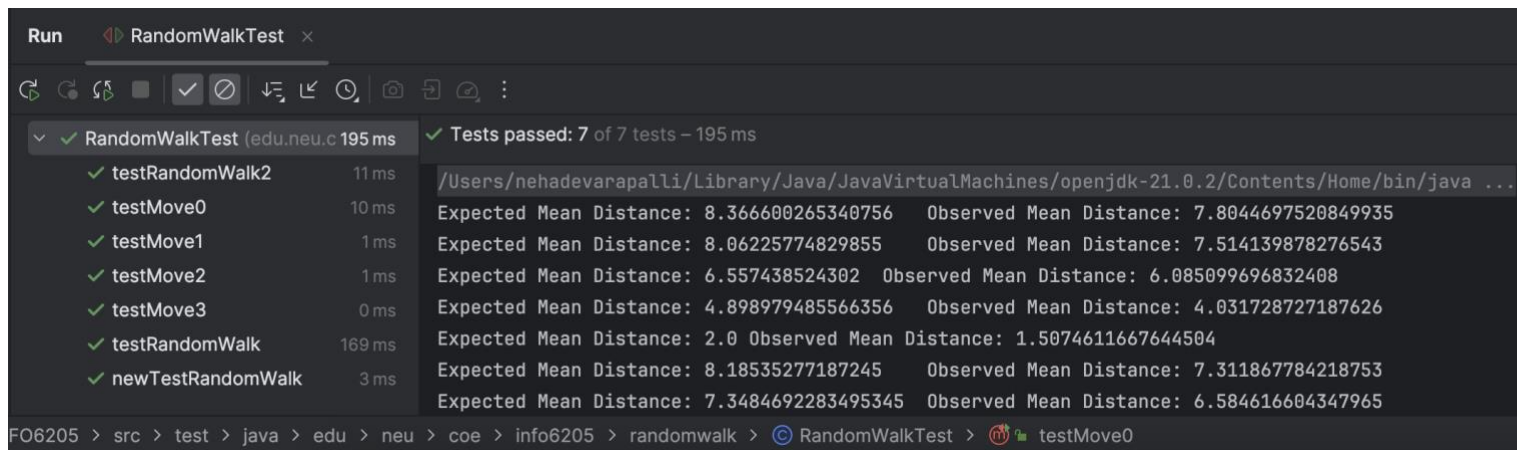
This is the data that I have got after running my own unit test where I calculate the expected distance with the relation proved above and compare it with the observed distances.

	# Expected	# Observed
1	8.3666	7.1237
2	8.0623	7.4504
3	6.5574	6.0919
4	4.8990	4.2999
5	2.0000	1.6841
6	8.1854	7.4518
7	7.3485	6.2625
8	8.1854	7.4129
9	9.4868	8.0047
10	3.1623	3.0155

I have also plotted a graph to observe the relation between the two,



Unit Test Screenshots:



```
Run RandomWalkTest x
✓ RandomWalkTest (edu.neu.c 195 ms) ✓ Tests passed: 7 of 7 tests – 195 ms
  ✓ testRandomWalk2 11 ms
  ✓ testMove0 10 ms
  ✓ testMove1 1 ms
  ✓ testMove2 1 ms
  ✓ testMove3 0 ms
  ✓ testRandomWalk 169 ms
  ✓ newTestRandomWalk 3 ms
/Users/nehadevarapalli/Library/Java/JavaVirtualMachines/openjdk-21.0.2/Contents/Home/bin/java ...
Expected Mean Distance: 8.366600265340756 Observed Mean Distance: 7.8044697520849935
Expected Mean Distance: 8.06225774829855 Observed Mean Distance: 7.514139878276543
Expected Mean Distance: 6.557438524302 Observed Mean Distance: 6.085099696832408
Expected Mean Distance: 4.898979485566356 Observed Mean Distance: 4.031728727187626
Expected Mean Distance: 2.0 Observed Mean Distance: 1.5074611667644504
Expected Mean Distance: 8.18535277187245 Observed Mean Distance: 7.311867784218753
Expected Mean Distance: 7.3484692283495345 Observed Mean Distance: 6.584616604347965
FO6205 > src > test > java > edu > neu > coe > info6205 > randomwalk > RandomWalkTest > testMove0
```