

Santhosh Krishna Maddi (002127881)

Program Structures & Algorithms Fall 2021

Assignment No 1

Task:

- To implement a code for calculating the euclidean distance between the start point(lamppost) and endpoint of a drunken man's random walk path. He takes n random steps in any direction to reach the endpoint.
- Deduce the relationship between the number of steps(n) and the euclidean distance (d) traveled in a random path by a drunk man.

Relationship Conclusion:

After calculating the euclidean distance traveled by the man from the origin (lamp post) for multiple values of n (number of steps) it can be deduced that distance traveled is approximately equal to the square root of the number of steps when considering each step length as 1 unit.

$$d \approx \text{sqrt}(n)$$

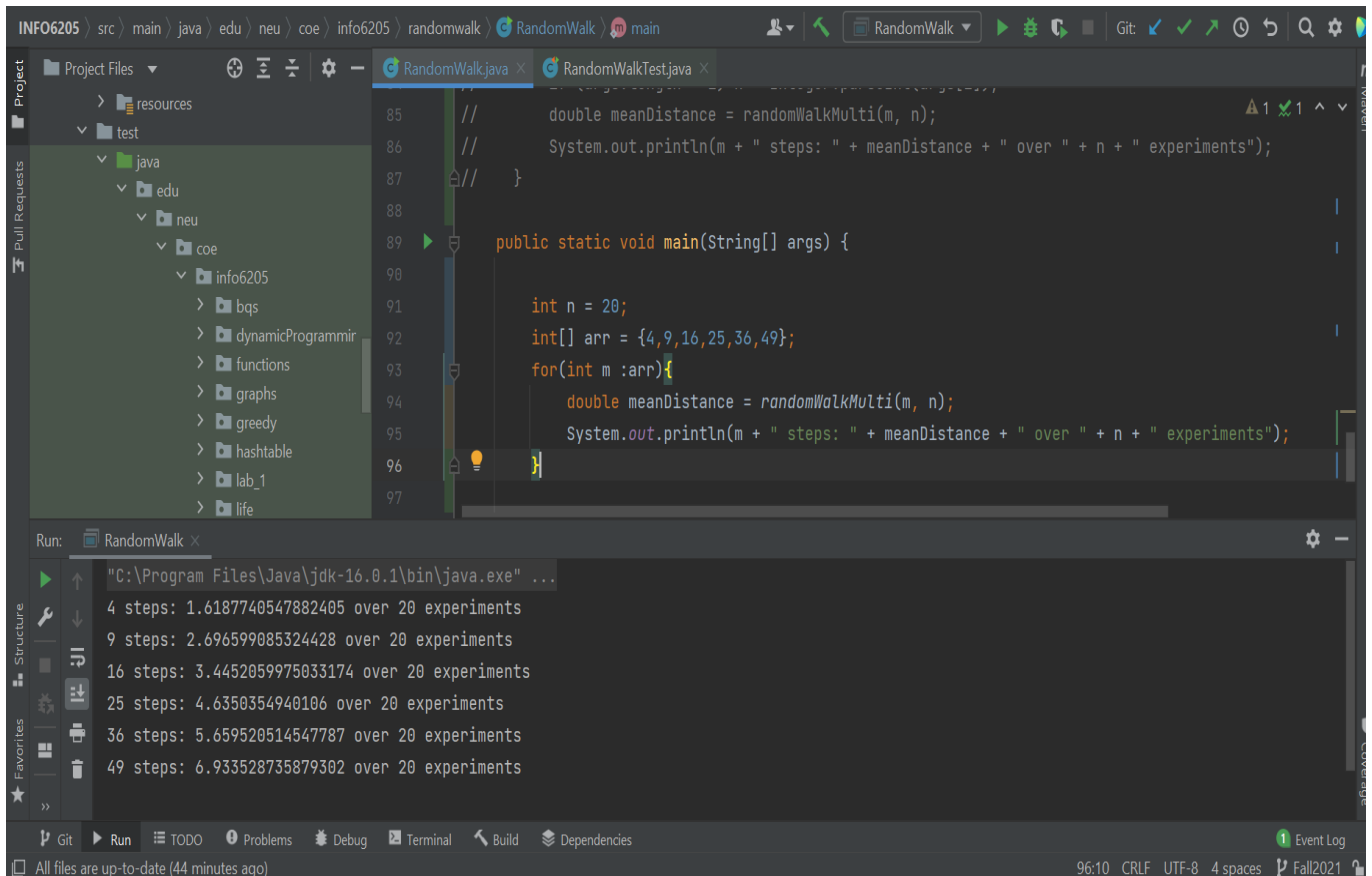
- n = number of steps walked by drunk man
- d = Euclidean distance from the origin (lamp post) after n steps.

Evidence to support the conclusion:

The experiment is conducted for 6 different values of n(number of steps). For each of these values, the experiment is conducted 20 times and the average value of distance for 20 iterations is calculated as shown below:

1. 4 steps: 1.6187740547882405 over 20 experiments
2. 9 steps: 2.696599085324428 over 20 experiments
3. 16 steps: 3.4452059975033174 over 20 experiments
4. 25 steps: 4.6350354940106 over 20 experiments
5. 36 steps: 5.659520514547787 over 20 experiments
6. 49 steps: 6.933528735879302 over 20 experiments

Output (Snapshot of Code output in the terminal)



The screenshot shows an IDE with a project named 'INFO6205'. The code in 'RandomWalk.java' defines a 'main' method that runs 20 experiments for each of the values 4, 9, 16, 25, 36, and 49. The terminal output shows the results of these experiments, with the average distance increasing as the number of steps increases.

```
85 // double meanDistance = randomWalkMulti(m, n);
86 // System.out.println(m + " steps: " + meanDistance + " over " + n + " experiments");
87 // }
88
89 public static void main(String[] args) {
90
91     int n = 20;
92     int[] arr = {4,9,16,25,36,49};
93     for(int m :arr){
94         double meanDistance = randomWalkMulti(m, n);
95         System.out.println(m + " steps: " + meanDistance + " over " + n + " experiments");
96     }
97 }
```

Run: RandomWalk ×

```
"C:\Program Files\Java\jdk-16.0.1\bin\java.exe" ...
4 steps: 1.6187740547882405 over 20 experiments
9 steps: 2.696599085324428 over 20 experiments
16 steps: 3.4452059975033174 over 20 experiments
25 steps: 4.6350354940106 over 20 experiments
36 steps: 5.659520514547787 over 20 experiments
49 steps: 6.933528735879302 over 20 experiments
```

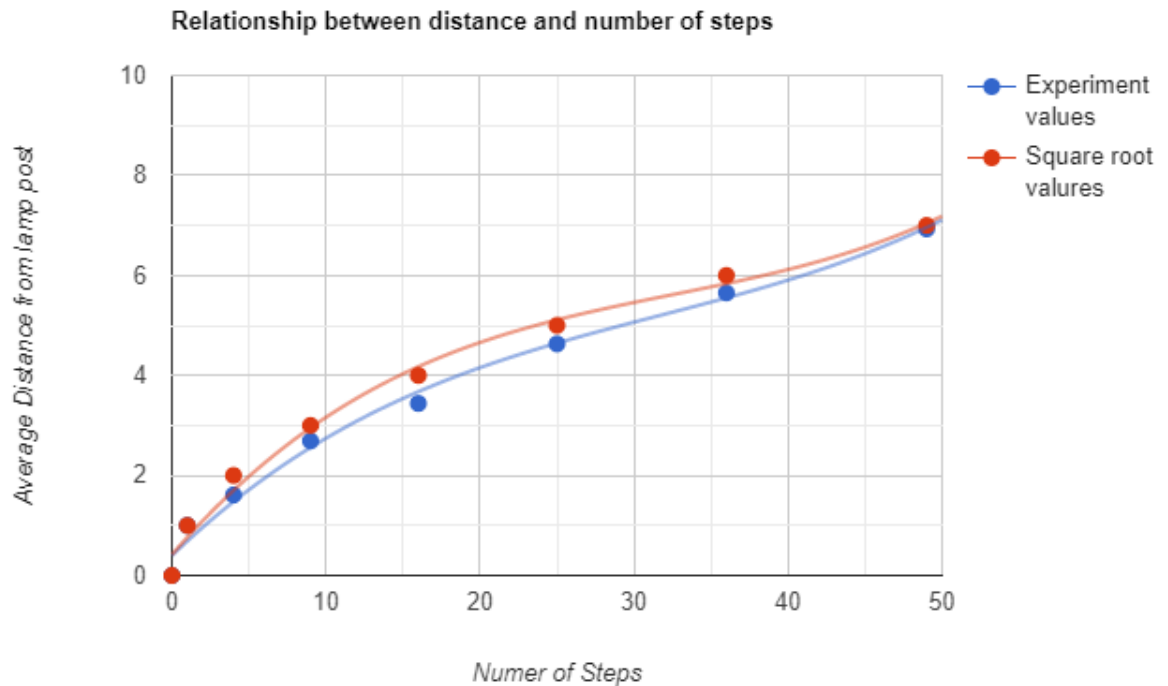
Observations from experiments :

The average distance calculated for different values of n (number of steps) along with the square roots of n are in below table

No of steps	Euclidean distance	The square root of the No of steps
4	1.61	2
9	2.69	3
16	3.44	4
25	4.63	5
36	5.65	6
49	6.93	7

Graphical Representation of the experiment values

The values from the above table are plotted as a graph to determine the relationship between the square root values of n and experimental values of d as shown below



X-axis: Number of steps

Y-axis: Average distance from lamp post for 20 experiments

From the above graph, it can be deduced that distance traveled is approximately equal to the square root of the number of steps

Unit tests result:

The below snapshot shows the successful run of unit test cases.

