

PSA – Assignment 1

Random Walk

Name: Vedantini Gaikwad

NUID: 002998254

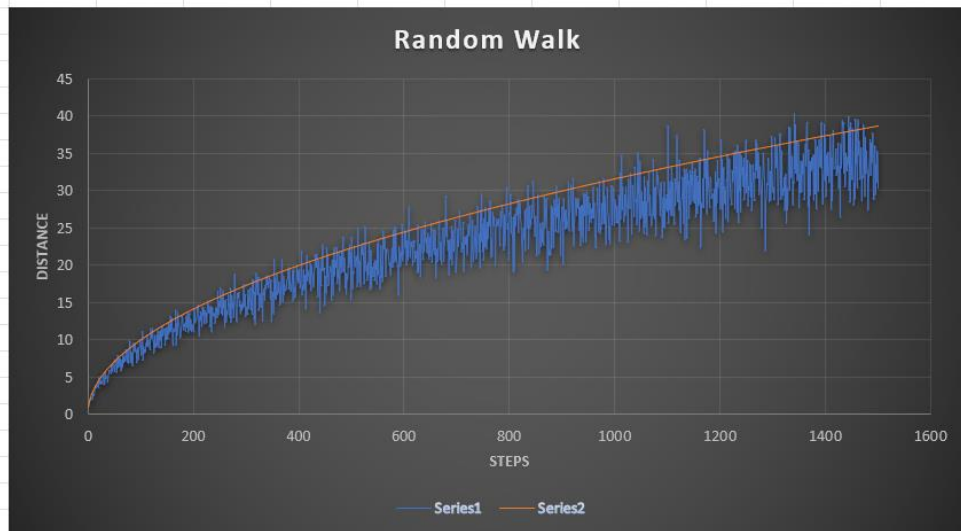
1. Relationship between the Euclidean distance (d) and steps taken (m): -

After performing the experiment 1500 times, I found out that the average distance, d, is approximately equal to the square root of the steps taken, m. Therefore,

$$d \approx \sqrt{m}$$

2. Evidence: -

Steps(m)	Distance(d)	Sqrt(m)
1	1	1
2	1.212825877	1.414213562
3	1.643498389	2
4	1.944610008	2.236067977
5	1.82844229	2.449489743
6	2.101751448	2.645751311
7	2.133959548	2.828427125
8	2.339252473	3
9	2.090661779	3.16227766
10	2.759599852	3.31662479
11	2.564737499	3.464101615
12	2.649177543	3.605551275
13	3.22838047	3.741657387
14	3.816918578	3.872983346
15	3.790887799	4



3. Code: -

```
/**
 * Perform a random walk of m steps
 *
 * @param m the number of steps the drunkard takes
 */
1 usage  xiaohuanlin *
private void randomWalk(int m) {
    for(int i=0; i<m ; i++){
        randomMove();
    }
}

/**
 * Private method to generate a random move according to the rules of the situation.
 * That's to say, moves can be (+-1, 0) or (0, +-1).
 */
1 usage  xiaohuanlin
private void randomMove() {
    boolean ns = random.nextBoolean();
    int step = random.nextBoolean() ? 1 : -1;
    move(ns ? step : 0, ns ? 0 : step);
}

/**
 * Method to compute the distance from the origin (the lamp-post where the drunkard starts) to his current position.
 *
 * @return the (Euclidean) distance from the origin to the current position.
 */
1 xiaohuanlin *
public double distance() { return Math.sqrt(Math.pow(x,2) + Math.pow(y,2)); }
```

```

public static void main(String[] args) {
    if (args.length == 0)
        throw new RuntimeException("Syntax: RandomWalk steps [experiments]");
    int m = Integer.parseInt(args[0]);
    int n = 30;
    if (args.length > 1) n = Integer.parseInt(args[1]);
    for(int i = 0; i <= 1500 ; i++) {
        double meanDistance = randomWalkMulti(m, n);
        //System.out.println(m + " steps: " + meanDistance + " over " + n + " experiments");
        System.out.println(Math.sqrt(meanDistance));
        m = m + 1;
    }
}

```

4. Unit Test: -

The screenshot shows the IntelliJ IDEA IDE with the `RandomWalkTest.java` file open. The code in the file is as follows:

```

double meanDistance = randomWalkMulti(m, n);
//System.out.println(m + " steps: " + meanDistance + " over " + n + " experiments");
System.out.println(Math.sqrt(meanDistance));
m = m + 1;
}

```

Below the code editor, the `Run` tab displays the test results for `RandomWalkTest`. The tests passed are:

Test Name	Duration
testRandomWalk2	19 ms
testMove0	18 ms
testMove1	2 ms
testMove2	2 ms
testMove3	2 ms
testRandomWalk	248 ms

The summary shows: **Tests passed: 6 of 6 tests - 291 ms**. The process finished with exit code 0.