Program Structures and Algorithms
Spring 2024

NAME: Ting Guo
NUID: 002834835
GITHUB LINK:https://github.com/Nangongnuanshan/INFO6205

Task:Assignment 1

Relationship Conclusion:

I think there is a positive correlation between m and d, and d increases with the increase of m, which is about $E(d) = \sqrt{m}$.

Evidence to support that conclusion:

1.1 Proof

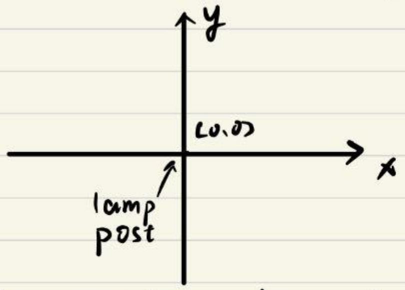
For the convenience of drawing and writing mathematical formulas, I have documented the proof process in a notebook. I am attaching a screenshot here.

Assignment 1 (Random Walk)

Evidence

Ting Guo

Assume the initial position coordinates to be $(0,0)$



When moving along the x-axis:
every 1 step, the possible values for x: -1 & 1

$\because$ random movement

$\therefore P(X=1) = P(X=-1) = \frac{1}{2}$

$\therefore E(X) = \frac{1}{2}\times 1 + \frac{1}{2}\times(-1) = 0$

$\because$ the movement of x is independent

$\therefore Var(X) = E(X^2) - [E(X^2)] = (-1)^2\cdot\frac{1}{2} + (1)^2\cdot\frac{1}{2} = 1.$

When moving along the y-axis:
The simulation of y is the same as that of x

$\therefore$ every 1 step for y:

$E(Y) = 0$

$Var(Y) = 1$

After m steps, we assume that this person moved α steps along the x-axis and β step along the y-axis.

$\alpha + \beta = m$

$V(X_\alpha) = \alpha \quad V(Y_\beta) = \beta$

$V(X_\alpha) + V(Y_\beta) = \alpha + \beta = m$,

$\therefore d = \sqrt{X_\alpha^2 + Y_\beta^2}$

We want to know how far(d), generally speaking.
So we need $E(d)$

$\because E(d^2) = E(X_\alpha^2) + E(Y_\beta^2)$

$\because Var(X_\alpha) = E(X_\alpha^2) - [E(X_\alpha)]^2 = E(X_\alpha^2) - 0 = \alpha$

$Var(Y_\beta) = \beta$

$\therefore E(d^2) = \alpha + \beta = m$

$\therefore E(d) = \sqrt{m}$

1.2 Comparison of code execution results

```
(base) guoting@guotingdeMacBook-Pro randomwalk % java edu.neu.coe.info6205.randomwalk.RandomWalk 16
16 steps: 4.085284402993043 over 30 experiments
(base) guoting@guotingdeMacBook-Pro randomwalk % java edu.neu.coe.info6205.randomwalk.RandomWalk 25
25 steps: 4.240720957658648 over 30 experiments
(base) guoting@guotingdeMacBook-Pro randomwalk % java edu.neu.coe.info6205.randomwalk.RandomWalk 36
36 steps: 4.850150387137902 over 30 experiments
(base) guoting@guotingdeMacBook-Pro randomwalk % java edu.neu.coe.info6205.randomwalk.RandomWalk 49
49 steps: 6.054445731395663 over 30 experiments
(base) guoting@guotingdeMacBook-Pro randomwalk % java edu.neu.coe.info6205.randomwalk.RandomWalk 64
64 steps: 6.7419232049702895 over 30 experiments
(base) guoting@guotingdeMacBook-Pro randomwalk % java edu.neu.coe.info6205.randomwalk.RandomWalk 81
81 steps: 7.136517620814374 over 30 experiments
(base) guoting@guotingdeMacBook-Pro randomwalk % java edu.neu.coe.info6205.randomwalk.RandomWalk 100
100 steps: 8.393128544011255 over 30 experiments
(base) guoting@guotingdeMacBook-Pro randomwalk % java edu.neu.coe.info6205.randomwalk.RandomWalk 10000
10000 steps: 72.73882399856944 over 30 experiments
```

When the number is small, the randomness experienced may be relatively higher. However, when attempting larger numbers, it can be observed that the fluctuation in values significantly decreases, aligning more closely with the patterns summarized above.

```
(base) guoting@guotingdeMacBook-Pro randomwalk % java edu.neu.coe.info6205.randomwalk.RandomWalk 1000000
1000000 steps: 1003.0805175707114 over 30 experiments
(base) guoting@guotingdeMacBook-Pro randomwalk % java edu.neu.coe.info6205.randomwalk.RandomWalk 9000000
9000000 steps: 3221.0499528287987 over 30 experiments
```

Unit Test Screenshots: