

電腦數值模擬導論--實習六

B03702030 會計三 吳懿峰

實習六(a)

1.

程式碼

```
Option Explicit
Sub random_walk_1D()
Dim i&, j&, sum_position&, abs_sum_position As Long
Dim x As Integer
Dim position(1 To 1000) As Long
Dim stTimer As Single
Dim avg_position!, avg_abs_position As Single
```

```
x = 0
```

```
RandomizeX
```

```
Do
```

```
stTimer = Timer
```

```
sum_position = 0
```

```
abs_sum_position = 0
```

```
avg_position = 0
```

```
avg_abs_position = 0
```

```
For i = 1 To 1000
```

```
    position(i) = 0
```

```
    For j = 1 To 10000
```

```
        If RndX > 0.5 Then
```

```
            position(i) = position(i) + 1
```

```
        Else
```

```
            position(i) = position(i) - 1
```

```
        End If
```

```
    Next
```

Next

For i = 1 To 1000

sum_position = sum_position + position(i)

abs_sum_position = abs_sum_position + Abs(position(i))

Next

avg_position = sum_position / 1000

avg_abs_position = abs_sum_position / 1000

ActiveWorkbook.Worksheets("第三頁").Select

ActiveSheet.Cells(3, 2 + 5 * x).Value = "no."

ActiveSheet.Cells(3, 3 + 5 * x).Value = "position"

For i = 4 To 1003

ActiveSheet.Cells(i, 2 + 5 * x).Value = i - 3

ActiveSheet.Cells(i, 3 + 5 * x).Value = position(i - 3)

Next

ActiveSheet.Cells(3, 4 + 5 * x).Value = "Avg.="

ActiveSheet.Cells(4, 4 + 5 * x).Value = avg_position

ActiveSheet.Cells(3, 5 + 5 * x).Value = "Abs.Avg="

ActiveSheet.Cells(4, 5 + 5 * x).Value = avg_abs_position

ActiveSheet.Cells(3, 6 + 5 * x).Value = "Time"

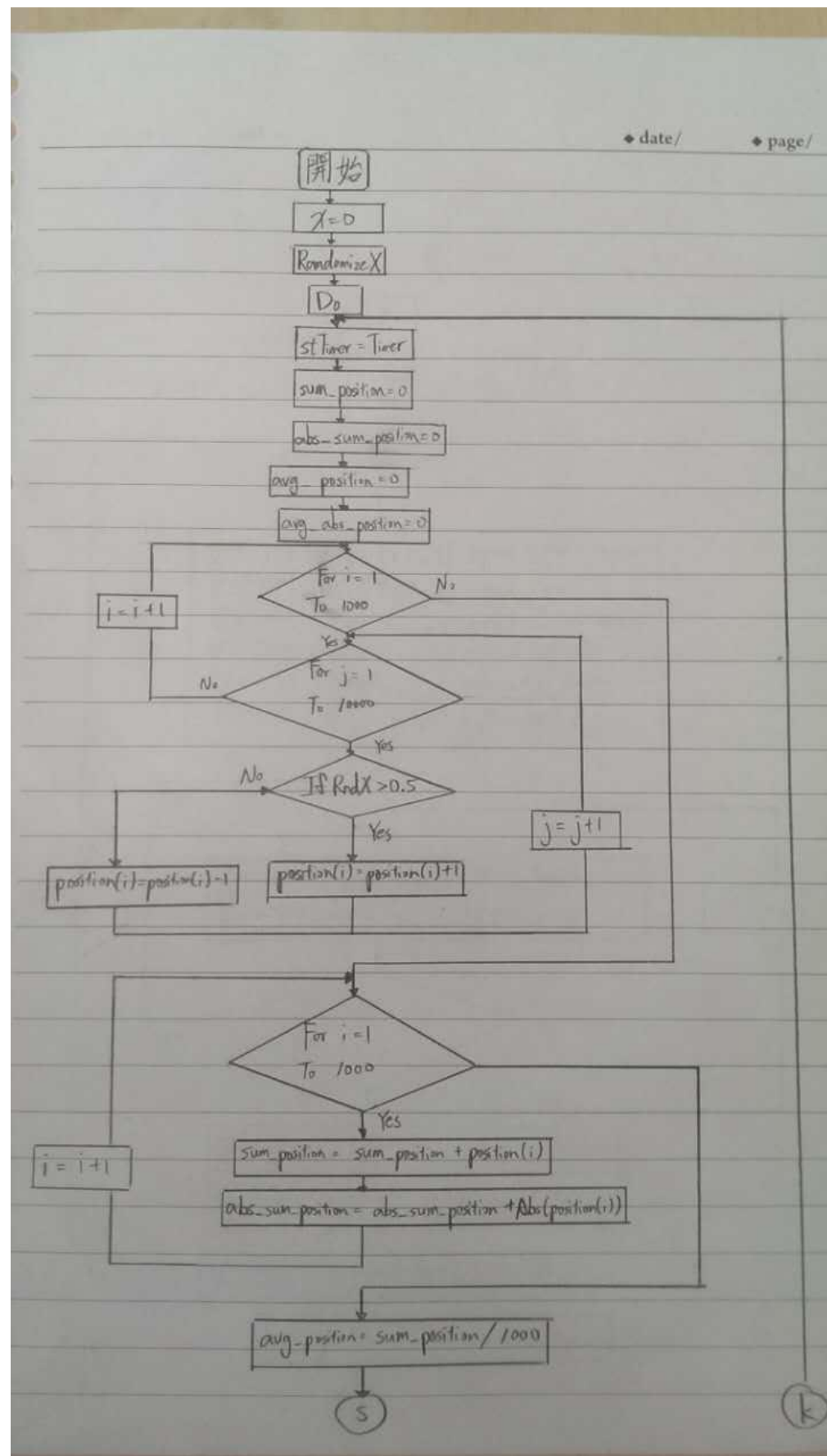
ActiveSheet.Cells(4, 6 + 5 * x).Value = Timer - stTimer

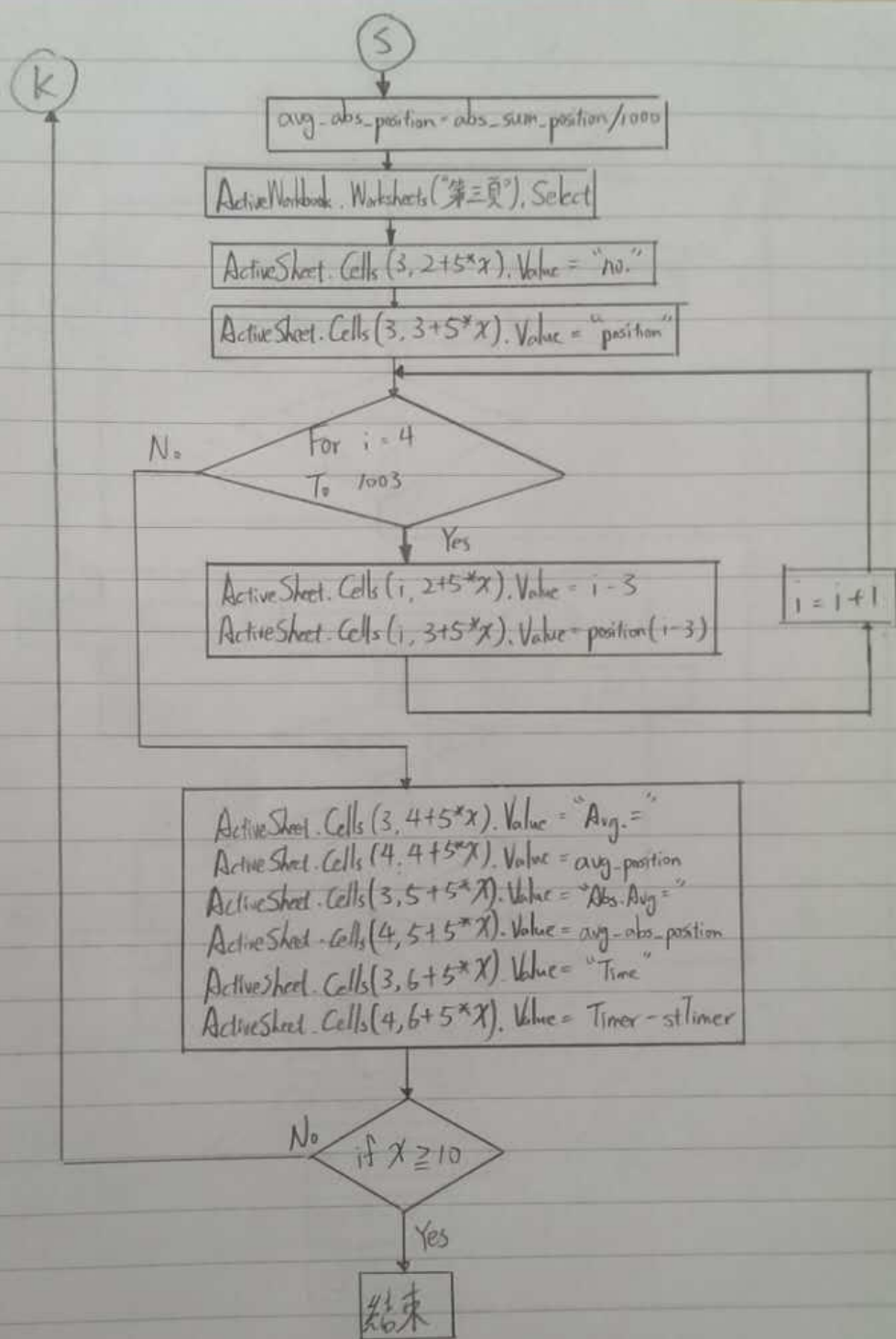
x = x + 1

Loop Until x >= 10

End Sub

流程圖





2. 列出此模擬的結果(詳見 excel 作業六(a) 第三頁)

第1次					第2次				
no.	position	Avg.=	Abs.Avg=	Time	no.	position	Avg.=	Abs.Avg=	Time
1	-156	1.746	82.326	2.324219	1	38	1.014	78.906	2.3125
2	16				2	128			
3	-194				3	74			
4	104				4	-24			
5	-28				5	122			
6	154				6	-110			
7	202				7	36			
8	-46				8	80			
9	84				9	44			
10	48				10	14			
第3次					第4次				
no.	position	Avg.=	Abs.Avg=	Time	no.	position	Avg.=	Abs.Avg=	Time
1	-170	-1.708	77.264	2.3125	1	58	-0.988	81.008	2.300781
2	-32				2	-138			
3	-94				3	0			
4	178				4	-62			
5	16				5	-136			
第5次					第6次				
no.	position	Avg.=	Abs.Avg=	Time	no.	position	Avg.=	Abs.Avg=	Time
1	-80	-0.814	77.234	2.280273	1	120	2.636	81.952	2.296875
2	-26				2	-106			
第7次					第8次				
no.	position	Avg.=	Abs.Avg=	Time	no.	position	Avg.=	Abs.Avg=	Time
1	-54	-3.28	77.824	2.292969	1	12	-2.632	82.172	2.293945
2	-286				2	-172			
3	-16				3	-10			
4	-156				4	158			
5	116				5	50			
第9次					第10次				
no.	position	Avg.=	Abs.Avg=	Time	no.	position	Avg.=	Abs.Avg=	Time
1	18	-2.034	77.574	2.313477	1	14	4.154	79.918	2.283203
2	-62				2	-80			
3	2				3	54			
4	0				4	106			
5	-52				5	-12			
6	-72				6	-80			
7	-2				7	-8			
8	134				8	-62			
9	-174				9	-20			
10	-160				10	-200			
11	40				11	50			

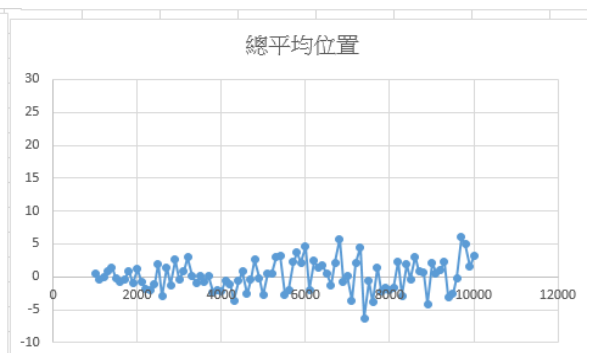
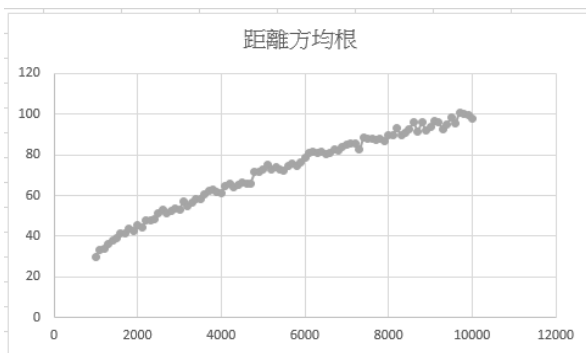
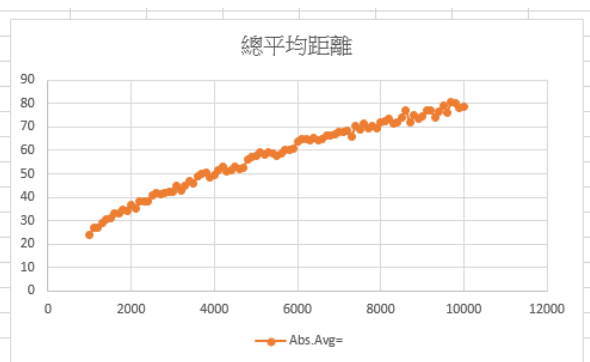
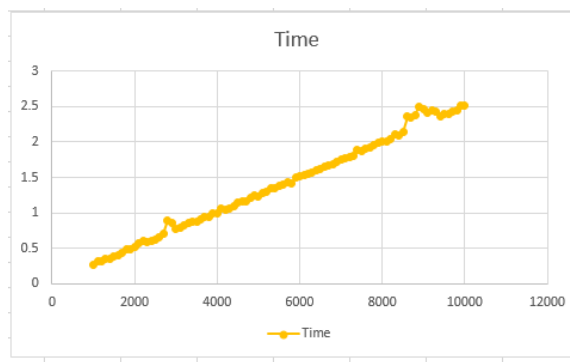
3. 總平均位置、總平均距離、距離方均根，三數值與跳動次數關係

(詳見 excel 作業六(a) 工作表二)

可以看出時間、總平均距離、距離方均根與跳動次數成正相關，而

總平均位置則變動不大。

跳動次數	Avg.=	Abs.Avg=	距離方均根	Time
1000	0.446	23.65800095	29.78005981	0.265625
1100	-0.45	26.68600082	33.33436584	0.3125
1200	-0.046	27.03800011	33.84665298	0.3125
1300	0.794	28.88999939	35.91918564	0.34375
1400	1.474	30.70999908	38.07177353	0.34375
1500	-0.214	31.1779995	39.20127487	0.390625
1600	-0.816	33.27999878	41.56873703	0.40625
1700	-0.45	33.11399841	41.37578964	0.4375
1800	0.93	34.77799988	43.63828659	0.476563
1900	-0.954	33.90200043	42.44992447	0.484375
2000	1.172	36.64799881	45.67055893	0.515625
2100	-0.728	35.36399841	44.13733292	0.5625



4. 跳動有三種可能且機率相等，結果如何？

(詳見 excel 作業六(a) 工作表三)

結果並沒有太大的不同。

5. 心得與想法

其實這次的題目看起來很難，實際操作的邏輯架構卻很簡單，話雖如此，這個作業還是燒掉了不少時間，大概是因為還沒真正熟悉 VBA 的緣故吧。一維度的隨機跳動結果是會落在接近原點的位置，大致與我原先的猜測相同，因為就期望值來說也應該會落在 0 上(一半機率往正，一半機率往負)。而當跳動變成有三種可能的時候也是一樣的結果，顯見一維模擬還是比較直觀，可以用期望值來作預測的。比較讓我思考的項目是跳動次數與總平均距離以及距離的方均根關係，在最後的結果中我們發現跳動次數越多，總平均距離以及距離的方均根的值也越大，這大概是因為，在跳動越多次時，他跳動的範圍變得更廣，雖然平均來說會落在原點附近，但跑出來的 range 卻會比跳動次數小的還大，需要注意的是總平均距離是一正比關係，但距離的方均根則是呈現一上升但逐漸趨緩的曲線。

實習六(b)

1. 程式碼

Option Explicit

Sub random_walk_2D1()

Dim i&, j As Long

Dim x%, y As Integer

Dim stTimer As Single

Dim pi As Double

Dim position_x(1 To 1000) As Single

Dim position_y(1 To 1000) As Single

Dim r As Single

Dim avg_dis_position As Single

Dim angle_p(1 To 1000) As Single

Dim sum_angle!, avg_angle!, sum_position_x!, sum_position_y!, avg_position_x!,

avg_position_y As Single

pi = WorksheetFunction.pi

x = 0

RandomizeX

Do

stTimer = Timer

sum_position_x = 0

sum_position_y = 0

avg_dis_position = 0

sum_angle = 0

ReDim pos(10000)

For i = 1 To 1000

position_x(i) = 0

position_y(i) = 0

For j = 1 To 10000

pos(j) = Rnd

position_x(i) = position_x(i) + Cos(pos(j) * 2 * pi)

position_y(i) = position_y(i) + Sin(pos(j) * 2 * pi)

Next

If position_x(i) > 0 And position_y(i) >= 0 Then

angle_p(i) = Atn(position_y(i) / position_x(i)) * 180 / pi

ElseIf position_x(i) < 0 And position_y(i) >= 0 Then

angle_p(i) = (Atn(position_y(i) / position_x(i)) * 180 / pi) + 180

ElseIf position_x(i) < 0 And position_y(i) < 0 Then


```

    angle_p(i) = (Atn(position_y(i) / position_x(i)) * 180 / pi) + 180
    ElseIf position_x(i) > 0 And position_y(i) < 0 Then
    angle_p(i) = (Atn(position_y(i) / position_x(i)) * 180 / pi) + 360
    ElseIf position_x(i) = 0 And position_y(i) > 0 Then
    angle_p(i) = 90
    ElseIf position_x(i) = 0 And position_y(i) < 0 Then
    angle_p(i) = 270
    End If

Next
For i = 1 To 1000
    sum_position_x = sum_position_x + position_x(i)
    sum_position_y = sum_position_y + position_y(i)
    sum_angle = sum_angle + angle_p(i)
Next

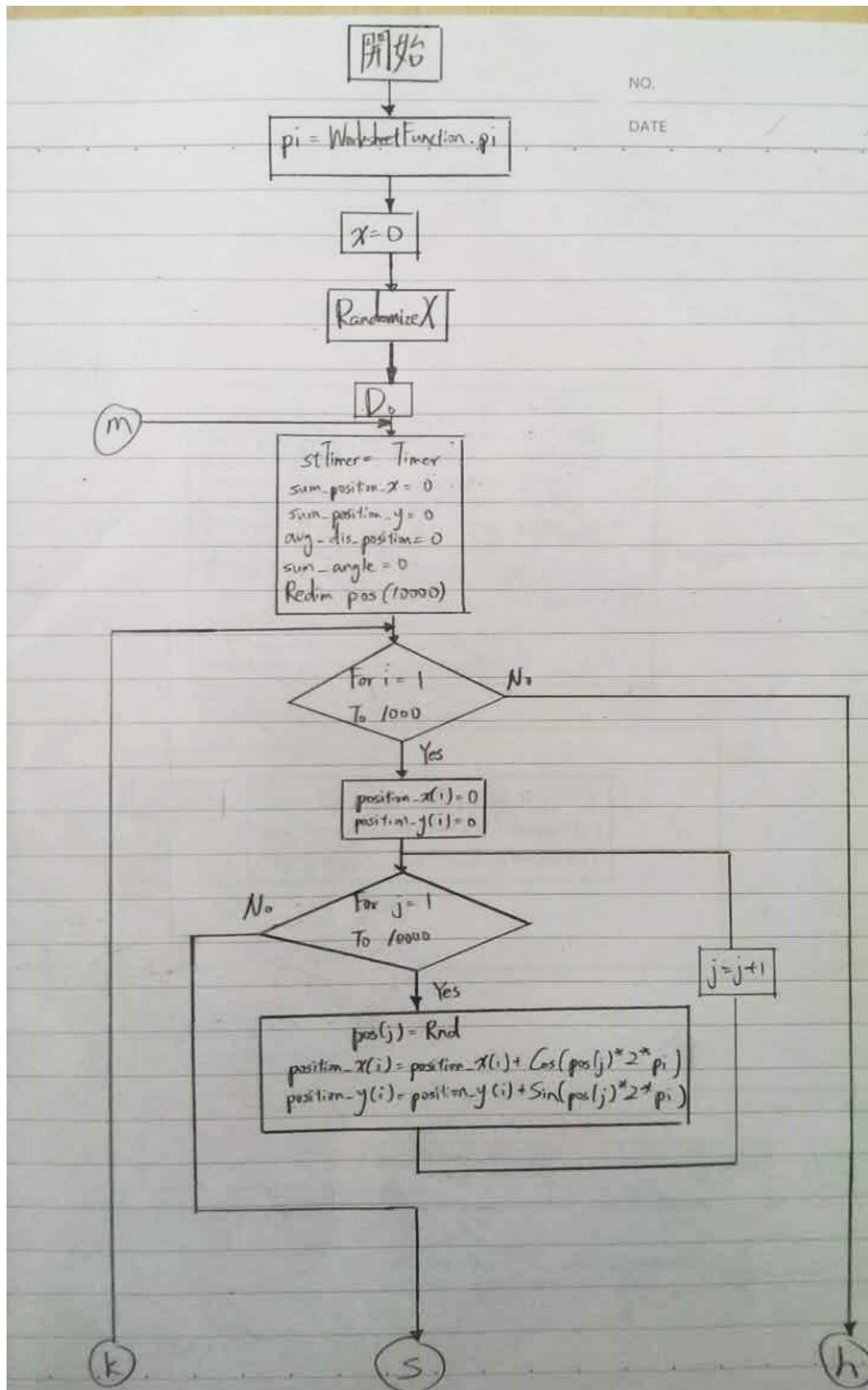
avg_position_x = sum_position_x / 1000
avg_position_y = sum_position_y / 1000
avg_angle = sum_angle / 1000
avg_dis_position = Sqr(avg_position_x ^ 2 + avg_position_y ^ 2)
ActiveWorkbook.Worksheets("第三頁").Select
ActiveSheet.Cells(3, 3 + 7 * x).Value = "no."
ActiveSheet.Cells(3, 4 + 7 * x).Value = "X 座標"
ActiveSheet.Cells(3, 5 + 7 * x).Value = "Y 座標"
ActiveSheet.Cells(3, 6 + 7 * x).Value = "Angle"
ActiveSheet.Cells(3, 7 + 7 * x).Value = "平均距離"
ActiveSheet.Cells(3, 8 + 7 * x).Value = "平均角度"
ActiveSheet.Cells(3, 9 + 7 * x).Value = "時間"

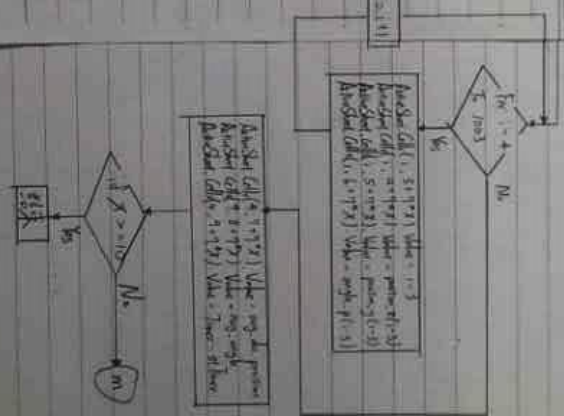
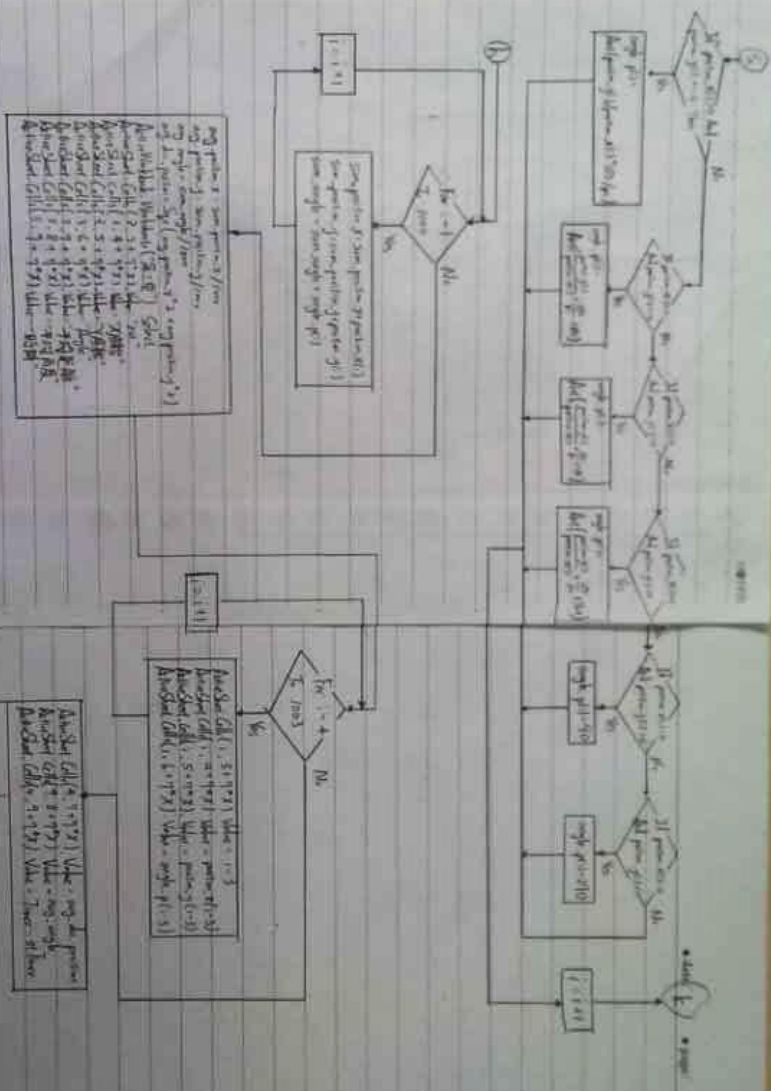
For i = 4 To 1003
    ActiveSheet.Cells(i, 3 + 7 * x).Value = i - 3
    ActiveSheet.Cells(i, 4 + 7 * x).Value = position_x(i - 3)
    ActiveSheet.Cells(i, 5 + 7 * x).Value = position_y(i - 3)
    ActiveSheet.Cells(i, 6 + 7 * x).Value = angle_p(i - 3)
Next

ActiveSheet.Cells(4, 7 + 7 * x).Value = avg_dis_position
ActiveSheet.Cells(4, 8 + 7 * x).Value = avg_angle
ActiveSheet.Cells(4, 9 + 7 * x).Value = Timer - stTimer
x = x + 1
Loop Until x >= 10
End Sub

```

流程圖



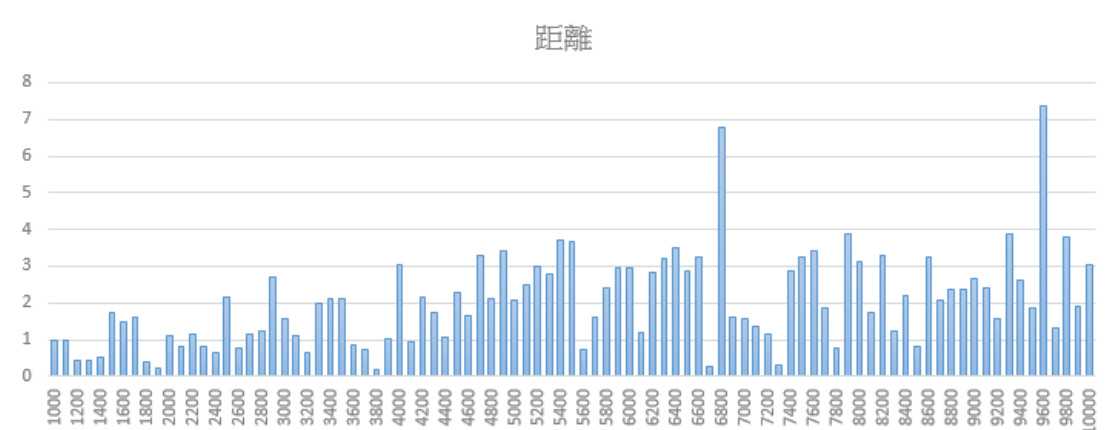


2. 十次模擬之數據(詳見 excel 作業六(b) 第三頁)

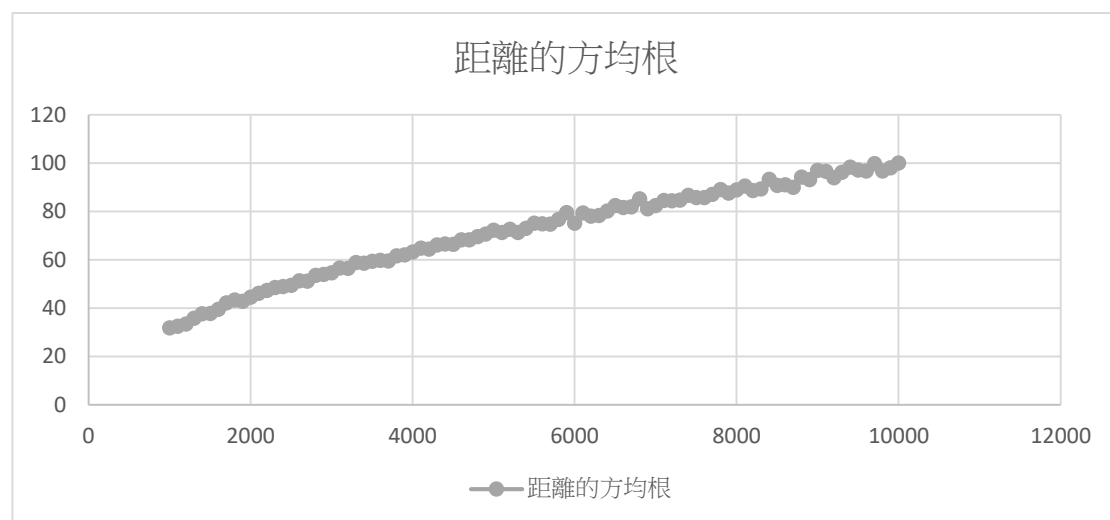
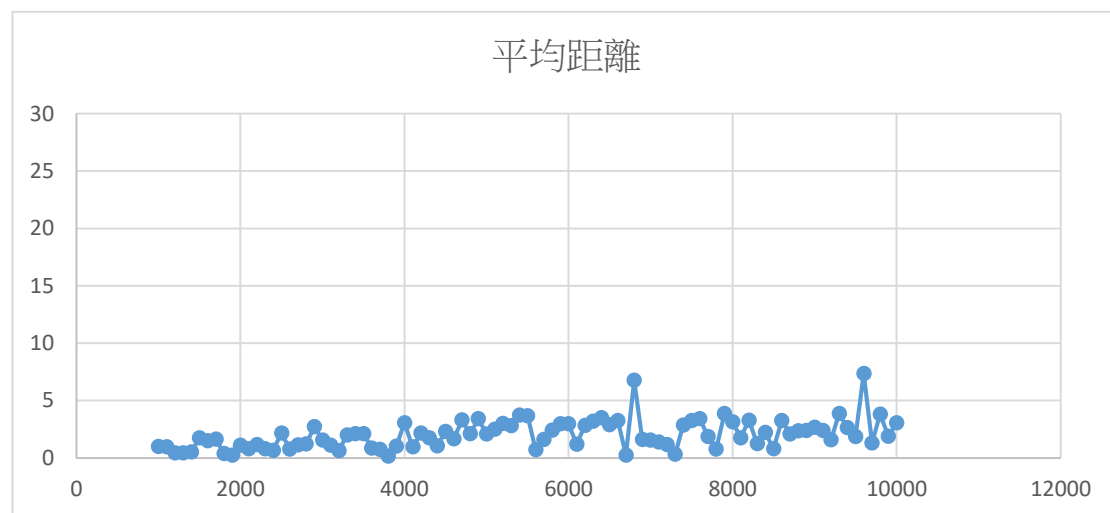
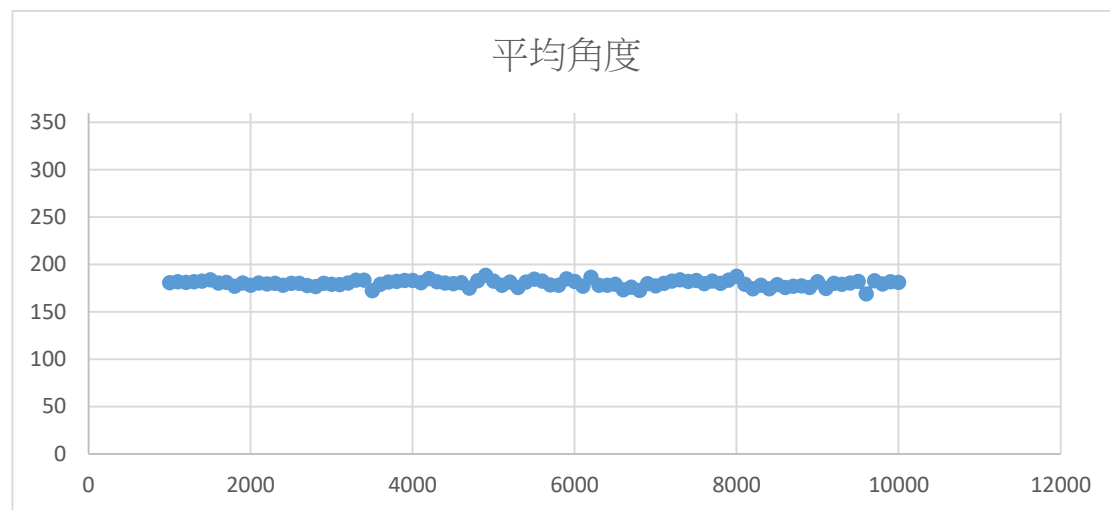
第1次							第2次						
no.	X座標	Y座標	Angle	平均距離	平均角度	時間	no.	X座標	Y座標	Angle	平均距離	平均角度	時間
1	-20.1426	83.40752	103.5768	3.019538	183.4863	4.035156	1	-32.9455	-24.383	216.5052	1.714365	180.1923	3.996094
2	-118.24	61.80555	152.4033				2	3.024367	40.9886	85.78004			
3	-1.60619	87.99889	91.04567				3	-77.761	-77.3089	224.833			
4	62.8196	-26.8477	336.8592				4	82.67497	41.03733	26.3984			
5	51.62888	9.596288	10.52944				5	14.48328	69.19362	78.1778			
6	-11.6818	-55.8451	258.1852				6	-57.1203	-11.0894	190.9868			
7	-61.8743	-26.0647	202.8434				7	-52.8004	-22.0625	202.6774			
8	-41.0786	1.242242	178.2679				8	-85.7371	14.33123	170.5106			
9	-43.8735	-174.882	255.9165				9	-47.4621	-71.836	236.5473			
10	-2.03533	65.4752	65.6663				10	45.1125	51.5000	241.5000			
第3次							第4次						
no.	X座標	Y座標	Angle	平均距離	平均角度	時間	no.	X座標	Y座標	Angle	平均距離	平均角度	時間
1	-35.0593	2.43807	176.022	2.064181	180.0243	3.972656	1	31.61946	55.15783	60.17632	2.703261	184.1653	3.9375
2	-33.5777	24.47873	143.9073				2	29.50332	74.10867	68.29205			
3	21.17467	-77.9556	285.1963				3	113.9413	-89.5382	321.8388			
4	-33.7995	39.82666	130.3201				4	-31.0113	-179.276	260.1861			
5	21.52548	-99.7179	282.1812				5	-42.814	-49.6368	229.2207			
6	-28.729	-24.8027	220.8051				6	54.01487	18.49777	18.90413			
7	-72.5183	-113.769	237.4859				7	30.16846	6.608982	12.35654			
8	-37.9881	-73.2129	206.7005				8	25.72979	-117.789	282.3221			
9	11.40601	50.80000	50.12250				9	62.12001	51.0055	241.5000			
10	-11.40601	50.80000	50.12250				10	62.12001	51.0055	241.5000			
第5次							第6次						
no.	X座標	Y座標	Angle	平均距離	平均角度	時間	no.	X座標	Y座標	Angle	平均距離	平均角度	時間
1	-38.4264	44.28662	130.9474	3.45323	176.3289	3.964844	1	-49.5195	100.4212	116.2487	3.232432	186.2052	3.867188
2	-5.59482	-22.289	255.9092				2	75.10397	56.96723	37.18074			
3	39.89183	103.3466	68.89339				3	-25.0337	64.84869	111.1082			
4	47.69885	1.16605	1.400379				4	-90.3188	-74.9634	219.6922			
5	82.11327	80.87477	44.56464				5	43.23511	54.93102	51.79442			
6	32.65979	-16.4012	333.3349				6	-66.1126	25.58883	158.8411			
7	24.16012	-91.9077	284.7283				7	11.21502	57.28319	78.92263			
8	44.99696	35.8488	38.54412				8	-14.5296	-68.2299	257.9784			
9	44.99696	35.8488	38.54412				9	-14.5296	-68.2299	257.9784			
10	44.99696	35.8488	38.54412				10	-14.5296	-68.2299	257.9784			
第7次							第8次						
no.	X座標	Y座標	Angle	平均距離	平均角度	時間	no.	X座標	Y座標	Angle	平均距離	平均角度	時間
1	21.6762	20.9816	44.06713	1.843286	178.3063	3.902344	1	-32.7745	-144.352	257.208	2.046815	181.1436	3.886719
2	164.0906	-10.8279	356.2247				2	-66.1865	-9.70606	188.3428			
3	-54.6969	22.7551	157.4116				3	-78.0765	-111.188	234.9233			
4	47.27544	44.17115	43.05575				4	40.35764	-113.344	289.5989			
5	-81.3977	-100.026	230.8626				5	83.18457	-53.9306	327.0437			
6	-119.33	-74.8096	212.0842				6	-89.1486	-31.0999	199.2316			
7	69.72981	-40	330.1596				7	-103.823	101.0081	135.7874			
8	-5.429	96.0132	93.23631				8	-114.378	24.49954	167.91			
9	46.12215	-183.302	284.1235				9	-94.6422	0.550455	179.6668			
10	46.12215	-183.302	284.1235				10	-94.6422	0.550455	179.6668			
第9次							第10次						
no.	X座標	Y座標	Angle	平均距離	平均角度	時間	no.	X座標	Y座標	Angle	平均距離	平均角度	時間
1	-25.6292	-21.0725	219.4273	2.828009	182.7958	3.945313	1	71.73011	99.84015	54.30466	1.793948	175.231	3.953125
2	82.46137	106.9564	52.36847				2	82.50002	-142.75	300.025			
3	-69.8095	-117.028	239.183				3	44.64488	-36.4725	320.753			
4	-23.9096	-77.153	252.7819				4	-73.3418	-88.1159	230.2283			
5	28.5916	50.84765	60.65099				5	6.458992	77.67037	85.24628			
6	-15.0377	114.5339	97.47986				6	-51.0123	-11.7189	192.938			

3. 用跳動次數與距離作直方圖(詳見 excel 作業六(b) 工作表 2)

跳動次數	平均距離	平均角度	距離的方均根	時間
1000	0.982034981	180.830307	31.71959877	0.51171875
1100	0.967912257	181.5969543	32.38942337	0.57421875
1200	0.426669329	181.1207886	33.29763794	0.61328125
1300	0.419034302	181.5781708	35.68464661	0.68359375
1400	0.526969552	182.4650574	37.61044312	0.734375
1500	1.736750841	183.7441559	37.67481232	0.76171875
1600	1.500870824	180.219696	39.39634705	0.82421875
1700	1.627257466	181.1120605	42.09076691	0.8984375
1800	0.388673425	177.049881	43.3230896	0.94921875
1900	0.226000577	180.4129486	42.72220612	1.00390625
2000	1.101423502	178.1092377	44.47228241	1.03515625
2100	0.796493113	180.5222321	46.10424805	1.0625
2200	1.153316021	179.2280884	47.21263504	1.12109375
2300	0.795806885	179.8943939	48.44333267	1.171875
2400	0.660227418	177.9855957	48.85640717	1.22265625
2500	2.172334433	180.1797943	49.41025543	1.3046875
2600	0.754099488	180.0284119	51.20289612	1.33984375
2700	1.142207503	177.5753937	51.10327148	1.39453125
2800	1.215757728	176.8323212	53.47836304	1.41015625
2900	2.710708326	180.1358185	53.85308102	1.4021875



4. 平均距離與平均方位角與次數的關係



平均角度與跳動次數、平均距離與跳動次數無關，平均角度大約落在 180 度上下，而平均距離則在 0 到 8 之間。

但是跳動距離的平方根則會與跳動次數呈現一正相關的關係，經觀察，其關係應為一上升但逐漸趨緩的曲線。

5. 是否會做三維、四維模擬？

大致上了解原理及作法，三維的話應該要將隨機擴散的分布想成一個在半徑為一公尺的圓上的隨機跳動，需要設三個陣列並套用合理的數學公式，四維的則是加上四個陣列等等...，以此類推。

6. 三維的模擬結果和一維、二維有何異同？

根據我的猜測，三維的隨機跳動還是會非常靠近原點(出發點)的，是與一維、二維較相似的結果，這是由於變數隨機分布，其總平均距離理應會因為正負抵銷而使其在原點附近。而三維之距離方均根與一維及二維則會一樣，我的猜測是依據一維與二維的結果，一維與二維的跳動次數與距離方均根的關係皆是一上升但逐漸趨緩的曲線，且數值也非常相近，因此根據這樣的推導我猜測三維也會是一樣的結果。

7. 心得

二維的 **Random Walk** 模擬概念與一維的很像，但是因為要考慮 **x** 座標與 **y** 座標，且每次移動距離都是 1，因此在計算時需要用到三角函數的概念。最為麻煩的部分則是將座標轉換成角度，而我想到的方法是利用 **Arctan** 函數先將其轉換成徑度，再 $\ast 180/\pi$ 得出角度，並利用 **If** 迴圈判斷各種不同的象限對角度作不同的調整得出正確的角度。

然而二維的結果與一維一樣，其最後的分布位置大概集中在原點附近，而平均角度則是逼近 180 度，這個結果想來也是非常合理，因為 **Randomize** 後的數值是平均分配，因此其期望值會接近 0.5，也就是轉換後的 180 度角。

很有趣地，觀察了一維與二維的距離方均根後發現其值幾乎是一樣的，這個結果告訴我們，不同的跳動次數可以改變跳動的範圍，而由於兩個數據的關係是固定的，因此可以濃縮成一個 **Function**，在往後幾個象限中，我認為距離方均根與跳動次數仍會依照此一公式進行。