Monte Carlo Method

Exercise:

1. Determine Area of irregular figure

Here, $\frac{F}{A} = \frac{M}{N}$

F = Area of irregular figure
A = Area of square to be considered
M = Points within shape
N = Total random points

Random number between (0-1)

For all cases we will consider 9×9 cm^2 regular shape

From Random number table

$x_1 = 0.96 \times 9 = 8.64$	$y_1 = 0.42 \times 9 = 3.78$
$x_2 = 0.71 \times 9 = 6.39$	$y_2 = 0.10 \times 9 = 0.9$
$x_3 = 0.03 \times 9 = 0.27$	$y_3 = 0.30 \times 9 = 2.7$

According to basic mathematical equation

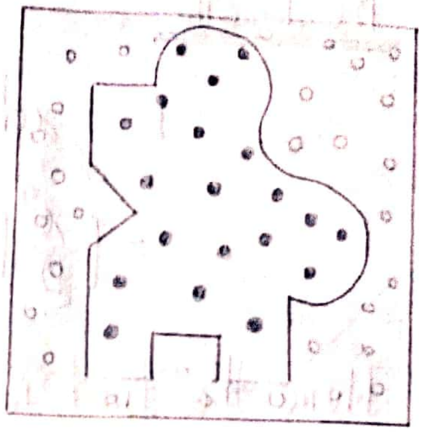
$$\text{Area} = \frac{26}{84} \times (9 \times 9) = 25.07 \text{cm}^2$$

$\because$ Area of square $= 9 \times 9 = 81$ cm^2

Considering the ratio of points falling into range which is 30.95%

Case 1:

$$\text{Area, } A = \frac{20}{50} \times 81 = 32.4\,\text{cm}^2$$



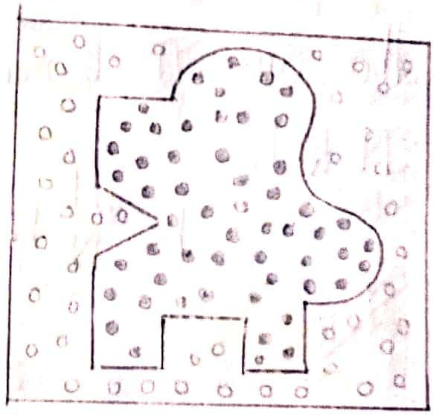
Main area = $25.07\,\text{cm}^2$

$\therefore$ Difference = $(32.4 - 25.07)$

$= 7.33$

Case 2:

$$\text{Area, } A = \frac{36}{100} \times 81 = 29.16\,\text{cm}^2$$



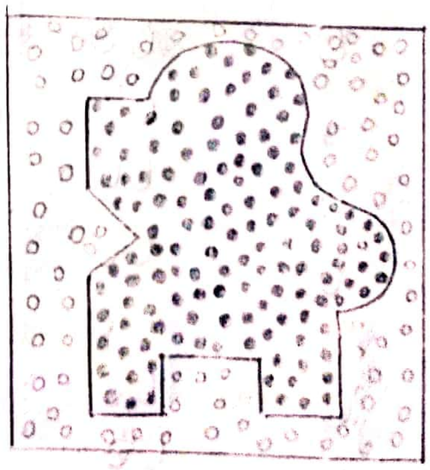
Calculated area = $25.07\,\text{cm}^2$

$\therefore$ Difference = $(29.16 - 25.07)\,\text{cm}^2$

$= 4.09\,\text{cm}^2$

Case 3:

$$\text{Area, } A = \frac{65}{200} \times 81 = 26.325\,\text{cm}^2$$



Difference = $(26.325 - 25.07)\,\text{cm}^2$

$= 1.255\,\text{cm}^2$

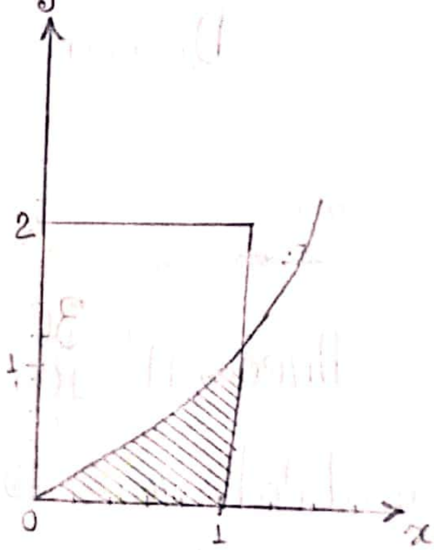
Accuracy grows higher with increasing points

2. Integrals by Monte Carlo

i) Solution:

$$I = \int_0^1 x\,dx$$

$$= \left[\frac{x^2}{2}\right]_0^1 = \frac{1}{2} \quad \text{(The exact value of interval)}$$



Here the limit is between [-0-1)
The ratio of the shaded area I to the area A of rectangele ABCD is M to N

$$I = \frac{M}{N} A$$

Simulation of $I = \int_0^1 x\,dx$

Random Number	x 0.1(rn)	Random Number	y 1(rn)	M	N
9	0.9	0.57	0.57	0	1
7	0.7	0.18	0.18	1	2
6	0.6	0.90	0.90	1	3
6	0.6	0.76	0.76	1	4
5	0.5	0.42	0.42	3	5
3	0.3	0.77	0.77	3	6
5	0.5	0.18	0.18	4	7
2	0.2	0.8	0.8	4	8
4	0.4	0.0	0.0	5	9

x has random raising in between 0-10 and y to 0-1.

$I = \frac{6}{9}(2\times 1) = 1.5$ which is quite the exact value

ii) Solutions:

Given that,

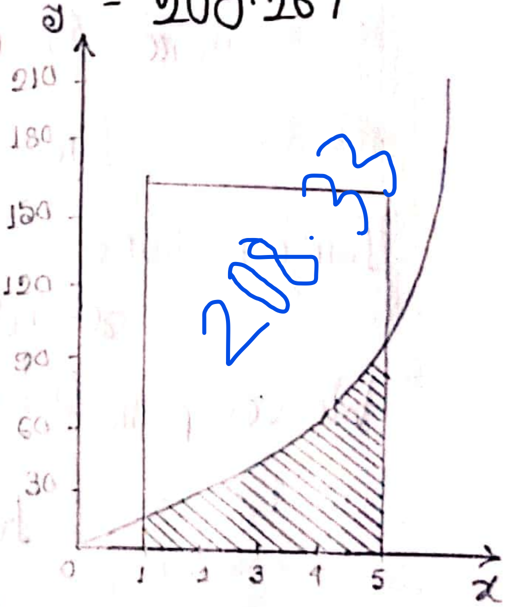
$$I = \int_1^5 \frac{x^4}{3}\,dx = \int_a^b f(x)\,dx$$

The exact value of integral, $I = \left[\frac{x^5}{15}\right]_1^5 = \left(\frac{3125-1}{15}\right)$

$= 208.267$

Simulation of $I = \int_1^5 \frac{x^4}{3}\,dx$

Random Number	x 0.1(r)	Random Number	y 175(r)	$x^4/3$	M	N
12	1.2	.57	99.75	.69	0	1
19	1.9	.18	31.05	4.3	0	2
23	2.3	.00	00	9.3	1	3
35	3.5	.27	47.25	50.02	2	4
41	4.1	.42	73.5	94.2	3	5
11	1.1	.45	78.75	0.49	3	6
13	1.3	.87	152.3	0.95	3	7
20	2.0	.25	43.8	5.33	3	8
29	2.9	.10	17.5	23.57	3	9
37	3.7	.60	105	62.5	3	10
47	4.7	0.77	115.5	162.7	4	11
33	3.3	0.26	45.75	39.5	4	12
25	2.5	0.05	8.75	13.02	4	13
26	2.6	0.40	70	15.23	4	14



We can't draw the rectangle which is less than the f(x) maximum value.

Here the x has random generated 2 10-50 and for y its opted to be 0-1. For $x = 0.1$ (Random)

$y = 175$ (Random)

$$I = \left(\frac{4}{14}\right)(175 \times 4)$$

$= 199.99 \approx 208.267$ which is near to the approximate value.

3. Determine the value of Pi

Here, Given that,

$$\text{Area of circle} = \frac{\pi D^2}{4} = \frac{\pi (2r)^2}{4} = \pi r^2$$

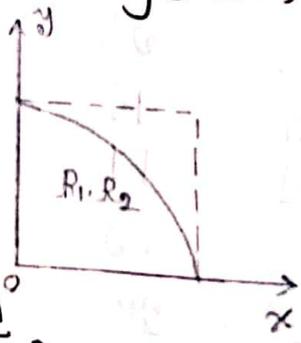
For equation,

$$x^2 + y^2 \leq 1 \quad \because x, y \geq 0$$

We've pair of Random number R_1, R_2 in range $(0,1)$

$$R_1^2 + R_2^2 \leq 1$$

$$\therefore R_2 \leq \sqrt{1 - R_1^2}$$



MN will take approach and get value $\frac{\pi}{4}$, Area under the curve

R_1	R_2	$\sqrt{1-R_1^2}$	I/O	R_1	R_2	$\sqrt{1-R_1^2}$	I/O
.82	.95	.5724	0	.76	.18	.6499	1
.34	.14	.9404	1	.33	.81	.9440	1
.20	.84	.9797	1	.35	.96	.9368	0
.58	.14	.8146	1	.29	.57	.9570	1
.52	.03	.8542	1	.95	.47	.3123	0
.51	.60	.8602	1	.84	.51	.5426	1
.34	.38	.9404	1	.34	.68	.9404	1
.72	.11	.6940	1	.85	.11	.5268	1
.50	.55	.8760	1	.58	.92	.8146	0
.08	.21	.9967	1	.69	.59	.7238	1

For 20 points $= \frac{16}{20} \times 4 = 3.2$

R_1	R_2	$\sqrt{1-R_1^2}$	I/O	R_1	R_2	$\sqrt{1-R_1^2}$	I/O
.82	.95	0.5724	0	.85	.11	.5268	1
.34	.14	.9404	1	.58	.92	.8146	0
.20	.84	.9797	1	.69	.59	.7238	1
.58	.14	.8146	1	.48	.37	.8773	1
.52	.03	.8542	1	.51	.72	.8602	1
.51	.60	.8602	1	.06	.33	.9982	1
.34	.38	.9404	1	.22	.90	.9755	1
.72	.11	.6940	1	.80	.76	.6000	0
.50	.55	.8760	1	.56	.29	.8265	1
.08	.21	.9967	1	.06	.66	.9982	1
.76	.18	.6499	1	.92	.40	.3919	0
.33	.81	.9440	1	.51	.11	.8602	1
.35	.96	.9368	0	.13	.44	.9915	1
.29	.57	.9570	1	.65	.74	.9599	1
.95	.47	.3123	0	.60	.27	.8000	1
.84	.51	.5426	1	.51	.16	.8602	1
.34	.68	.9409	1	.50	.41		1
				.13	.20	.9915	1

R_1	R_2	$\sqrt{1-R_1^2}$	1/0	R_1	R_2	$\sqrt{1-R_1^2}$	1/0
.94	.68	.3412	0	.51	.95	.8216	0
.26	.28	.9656	1	.78	.75	.6258	0
.33	.16	.9440	1				

For 40 points, $\pi = \frac{31}{40} \times 4 = 3.10$

4. Pi value determining program:

```
import math
import random
import matplotlib.pyplot as plt

random.seed(1)

N=1000
hit=0
interval = 1/1000

ax=[]   ay=[]
for i in range (1,1001):
        x = i*interval
        ax.append(x)
        y = math.sqrt (1-x*x)
        ay.append(y)

hx=[]  hy=[]
mx=[]  my=[]
```

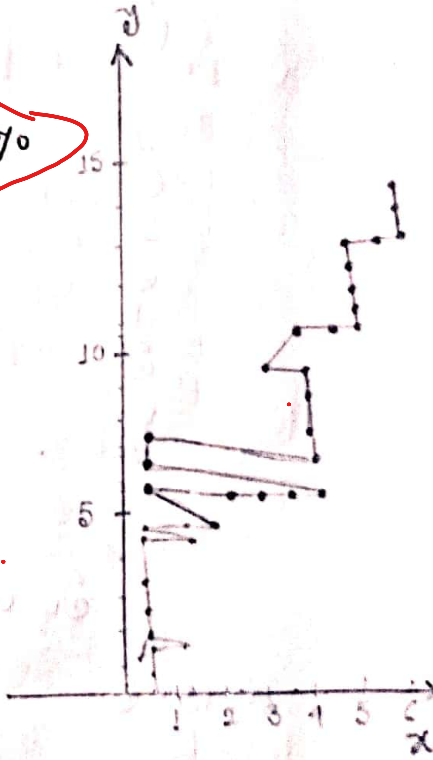
```
plt.figure (figsize = (6,6))
plt.plot ([[0,1,1,0,0],[0,0,1,1,0]], c='blue')

plt.plot (ax,ay, c='blue')
```

```
for i in range (1,N+1):
    x= random.uniform (0,1)
    y= random.uniform (0,1)
    if y <= math.sqrt (1-x*x)
        hit+=1
        hx.append(x).
        hy.append (y)
    else:
        mx.append(x)
        my.append(y)
est_pi = hit*4/N
print (est_pi)
```

5. Random walk Problem

The problem associated with these 40%, 10% 25%, 25% forward, backward, left, right respectively covering 75, 45, 60, 60 cm



Direction	Probability	Random Num
Forward	0.4	0, 1, 2, 3
Backward	0.1	4
Right	0.25	5, 6, 7
Left	0.25	8, 9

Random Num	Direction	Position x	Position y	Random	Direction	Position x	Position y
6	R	.6	0	5	R	3.6	6.3
2	F	.6	.75	7	R	3.2	6.3
0	F	.6	1.5	4	B	4.2	5.55
6	R	1.2	1.5	2	F	4.2	6.3
8	L	.6	1.5	0	F	4.2	7.05
4	B	.6	1.05	1	F	4.2	7.8
5	R	1.2	1.05	2	F	4.2	8.55
9	L	.6	1.05	3	F	4.2	9.3
3	F	.6	1.8	9	L	3.6	9.3
2	F	.6	2.55	5	R	4.2	9.3
2	F	.6	3.3	1	F	4.2	10.05
1	F	.6	4.05	3	F	4.2	10.8
7	R	1.2	4.05	4	B	4.2	10.05
8	L	.6	4.05	6	R	4.8	10.05
7	R	1.2	4.05	7	R	5.4	10.05
6	R	1.8	4.05	1	F	5.4	10.8
1	F	1.8	4.8	2	F	5.4	11.55
9	L	1.2	4.8	3	F	5.4	12.3
0	F	1.2	5.55	5	R	6.0	12.3
2	F	1.2	6.3	6	R	6.6	12.3
3	F	1.2	7.05	9	L	6.0	12.3
4	B	1.2	6.3	1	F	6.0	13.05
5	R	1.8	6.3	1	F	6.0	13.8
6	R	2.4	6.3	4	B	6.0	13.05
7	R	3.0	6.3	5	R	6.6	13.05

So distance travelled 13.05 in y axis & 6.6 in x axis.

6. Bombing Problems Simulations:

$x_1 = 500 Z_x$

$y_1 = 300 Z_y$

RNN	x	RNN	y	Result	RNN	x	RNN	y	Result
·23	115	·24	72	H	·13	65	·15	45	H
-1·16	-580	-0·02	-6	M	·02	10	-·43	-129	H
0·39	195	0·64	192	H	-1·05	-525	-·57	-171	M
-1·90	-950	-1·04	-312	M	-·55	-275	-·52	-156	M
-0·78	-390	0·68	204	H	·9	450	-·13	-39	H
-0·02	-10	-0·47	-141	H	1·16	580	1·13	339	H
-0·40	-200	-0·75	-225	H	·24	120	1·02	306	M
-0·66	-330	-0·44	-132	H	·37	185	-0·2	-60	H
1·41	705	1·21	363	M	-·95	-475	·93	279	M
0·07	35	-0·08	-24	H	-·03	-15	-1·01	-303	H
0·53	265	-1·56	-468	M	-1·17	-585	-·15	-45	M
0·03	15	1·49	447	M	-·68	-340	-·21	-63	M
-1·19	-595	-1·19	-357	M	·25	125	1·53	459	M
0·11	55	-1·86	-558	M	·17	85	·14	42	H
0·16	80	1·21	363	M	1·11	555	·16	48	H
-0·82	-410	0·75	225	H	·20	100	1·17	351	M
0·42	210	-1·50	-450	M	-·35	-175	-1·18	-354	H
-0·89	-445	0·19	57	H	·5	250	·29	87	H
0·49	249	-1·44	-432	M	-1·00	-500	·30	90	M
-0·21	-105	0·65	195	H	-·53	-265	·52	156	M

Number of hit = 20

Number of miss = 20

$\therefore$ % of strike = 50

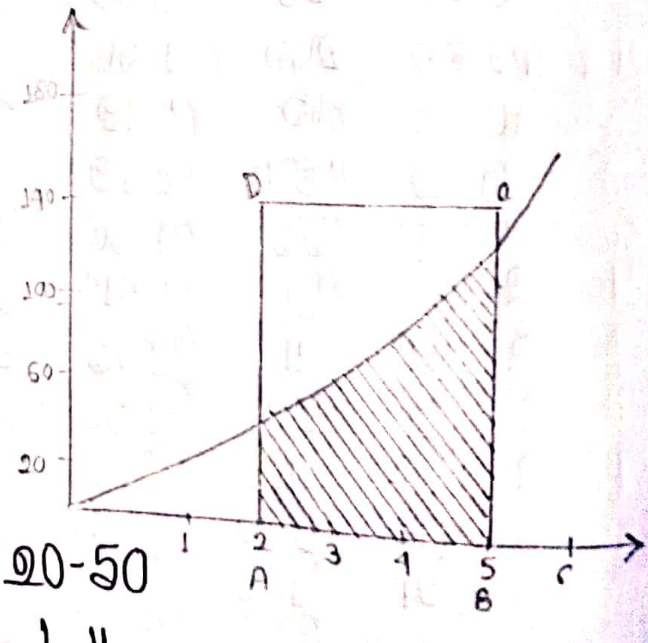
7. Random Variable:

$$x \sim N(\mu = 50,\ \sigma^2 = 65)$$
$$y \sim N(\mu = 100,\ \sigma^2 = 100)$$
$$z \sim N(\mu = 150,\ \sigma^2 = 150)$$

RN1	x	RN2	y	RN3	z	$x+y$	ω
-0.46	20.1	0.09	109	0.59	238.5	129.1	0.59
-1.05	-18.3	0.56	156	-0.67	49.5	137.7	2.78
-0.15	40.3	-0.02	98	0.41	211.5	138	0.65
-0.81	-2.65	3.10	410	0.51	226.5	407.35	1.79
-0.74	1.9	-0.44	56	1.53	379.5	57.9	0.15
-0.39	24.65	-0.33	67	-0.37	94.5	91.65	0.96
-0.45	20.75	0.40	140	-0.27	109.5	160.75	1.46
-2.44	-108.6	0.38	138	-0.15	127.5	29.4	0.23

8. Integral By Monte Carlo

$$I = \int_2^5 x^3 dx = \int_a^b f(x) dx$$
$$= \left[\frac{x^4}{4}\right]_2^5 = 152.25$$



Ratio of ABCD shaded area $I = \frac{M}{N} A$

Here x Ranges random number 20-50 and y (0-1) if $y \le x^3$ then count the number in range M

Random Number	x 0-π	Random Number	y 140(π)	x3	M	N
22	2.2	.57	79.8	10.65	0	1
25	2.5	.18	25.2	15.63	0	2
18	1.8	.00	0.0	5.83	1	3
45	4.5	.90	126.0	91.13	1	4
25	2.5	.05	7.00	15.63	2	5
27	2.7	.77	107.8	19.68	2	6
48	4.8	.66	92.4	110.60	3	7
43	4.3	.10	14.0	79.51	4	8
40	4.0	.76	106.4	64.0	4	9
47	4.7	.42	58.8	103.83	5	10
38	3.8	.78	109.2	54.87	5	11
33	3.3	.88	123.2	35.94	5	12
24	2.4	.03	4.2	13.82	6	13
47	4.7	.09	12.6	103.80	7	14
42	4.2	.77	107.6	74.09	7	15
25	2.5	.61	85.4	15.63	7	16

$$I = \left(\frac{7}{16}\right) \times (140 \times 3) = 183.75$$

1702024 Abid Ahammed

10. Bearing Problem:

Hours	Probability	Cum-Prob	Random digit
1200	·10	·10	01-10
1400	·25	·35	11-35
1600	·30	·65	36-65
1800	·25	·90	66-90
2000	·10	1·0	91-00

Delaytime	Probability	Cum-Prob	Random Digit
4	0·3	0·3	1-3
6	0·6	0·9	4-9
8	0·1	1·0	0

RD	Life	Clock	RD	Delay	RD	Life	Clock	RD	Delay	RD	Life	Clock	RD	Delay
25	1400	1400	9	6	65	1600	1600	5	6	94	2000	2000	7	6
82	1800	3200	8	6	91	2000	3600	0	8	87	1800	3800	0	6
16	1400	4600	0	8	30	1400	5000	5	6	63	1600	5400	6	6
14	1400	6000	5	6	66	1800	6800	6	6	66	1800	7200	2	4
82	1800	7800	2	4	32	1600	8400	5	6	30	1400	8600	2	4
45	1600	9400	3	4	29	1400	9800	3	4	69	1800	10400	7	6
81	1800	11200	7	6	11	1400	11200	2	4	37	1600	12000	2	4
80	1800	13000	3	4	43	1600	12800	8	6	1	1200	13200	8	6
97	2000	15000	7	6	40	1600	14400	0	8	66	1800	15000	8	8
62	1600	16600	4	6	65	1600	16000	8	6	51	1600	16600	3	6
30	1400	18000	9	6	82	1800	17800	2	4	92	2000	18600	8	6
31	1600	19600	5	6	73	1800	19600	4	6	36	1600	20200	3	4
80	1800	21400	6	6	15	1400	21000	4	6	47	1600	21800	7	6
72	1800	23200	7	6	70	1800	22800	6	4	80	1800	23600	0	4
Total Delay				1960					13080					1976

Delay From calculation = 80+80+76
= 236

Cost of bearing = 69x14 = 966

Total downtime = 966+236 = 1202

Cost of downtime = 1202x5 = 6010

Cost of repairment = ~~1380~~ 1202x $\frac{19}{42}$ = 543.76

11. Relaiability Problem Program

```
import math
import random
import matplotlib.pyplot as plt

n=100000
Delay=0
Life=0
for i in range(1,n):
    RD1 = random.randint(0,100)
    RD11 = random.randint(0,9)

if 1<RD1<10:
    Life = Life+1000
elif 11<RD1<24:
    Life = Life+1100
elif 25<RD1<48:
    Life = Life+1200
elif 49<RD1<62:
    Life = Life+1300
elif 63<RD1<74:
    Life = Life+1400
elif 75<RD1<84:
    Life = Life+1500
elif 85<RD1<90:
    Life = Life+1600
elif 91<RD1<95:
    Life = Life+1700
elif 96<RD1<98:
    Life - Life+1800
else:
    Life = Life+1900
if 1<RD11<3:
    delay = delay+4;
```

```
elif 4< RD11<9:
    delay = delay+6
else:
    delay = delay+8
```

```
Printf (delay)
Printf (Life)
```

13. Hinge Assembly Problem

```
import math
import random
import matplotlib.pyplot as plt

n = 1000 , count = 0
for in in range (1, n):
    RDa = random.rand (0,1)
    RDb = random.rand (0,1)
    RDc = random.rand (0,1)
    RDd = random.rand (0,1)

    a = RDa(0.1) + 1.95
    b = RDb(0.1) + 1.95
    c = RDc + 29.5    d = 34 + RDd
    sum = a+b+c
if sum > d
    count++;
Result = count / n x 100
Print (Result)
```