66310837

นายจิรัฐ ฟองดา

```
import matplotlib.pyplot as plt
import numpy as np

# np.random.rand(100) #random 0-1
np.random.rand(10)*100

array([89.49372742, 44.36925006, 81.29913188, 86.08531135,
1.2202966 ,
       19.63796647, 45.78734564, 85.95306341, 99.23903904,
7.43436877])

np.random.randint(1, 100, 50)

array([71, 64, 45, 37, 44, 44, 37, 72, 53, 66, 64, 55,  8, 14, 48, 61,
83,
       57, 97, 31, 32, 73, 55, 52, 48, 76, 90, 71, 10, 48, 99, 43, 10,
82,
       35, 13, 20, 36, 17, 47, 28, 70,  4, 18,  3, 99, 61,  3, 58,
98],
      dtype=int32)

np.random.seed(1)
for x in range(0, 20):
    numbers = np.random.rand(6)
    print(numbers)

[4.17022005e-01 7.20324493e-01 1.14374817e-04 3.02332573e-01
 1.46755891e-01 9.23385948e-02]
[0.18626021 0.34556073 0.39676747 0.53881673 0.41919451 0.6852195 ]
[0.20445225 0.87811744 0.02738759 0.67046751 0.4173048  0.55868983]
[0.14038694 0.19810149 0.80074457 0.96826158 0.31342418 0.69232262]
[0.87638915 0.89460666 0.08504421 0.03905478 0.16983042 0.8781425 ]
[0.09834683 0.42110763 0.95788953 0.53316528 0.69187711 0.31551563]
[0.68650093 0.83462567 0.01828828 0.75014431 0.98886109 0.74816565]
[0.28044399 0.78927933 0.10322601 0.44789353 0.9085955  0.29361415]
[0.28777534 0.13002857 0.01936696 0.67883553 0.21162812 0.26554666]
[0.49157316 0.05336255 0.57411761 0.14672857 0.58930554 0.69975836]
[0.10233443 0.41405599 0.69440016 0.41417927 0.04995346 0.53589641]
[0.66379465 0.51488911 0.94459476 0.58655504 0.90340192 0.1374747 ]
[0.13927635 0.80739129 0.39767684 0.1653542  0.92750858 0.34776586]
[0.7508121  0.72599799 0.88330609 0.62367221 0.75094243 0.34889834]
[0.26992789 0.89588622 0.42809119 0.96484005 0.6634415  0.62169572]
[0.11474597 0.94948926 0.44991213 0.57838961 0.4081368  0.23702698]
```

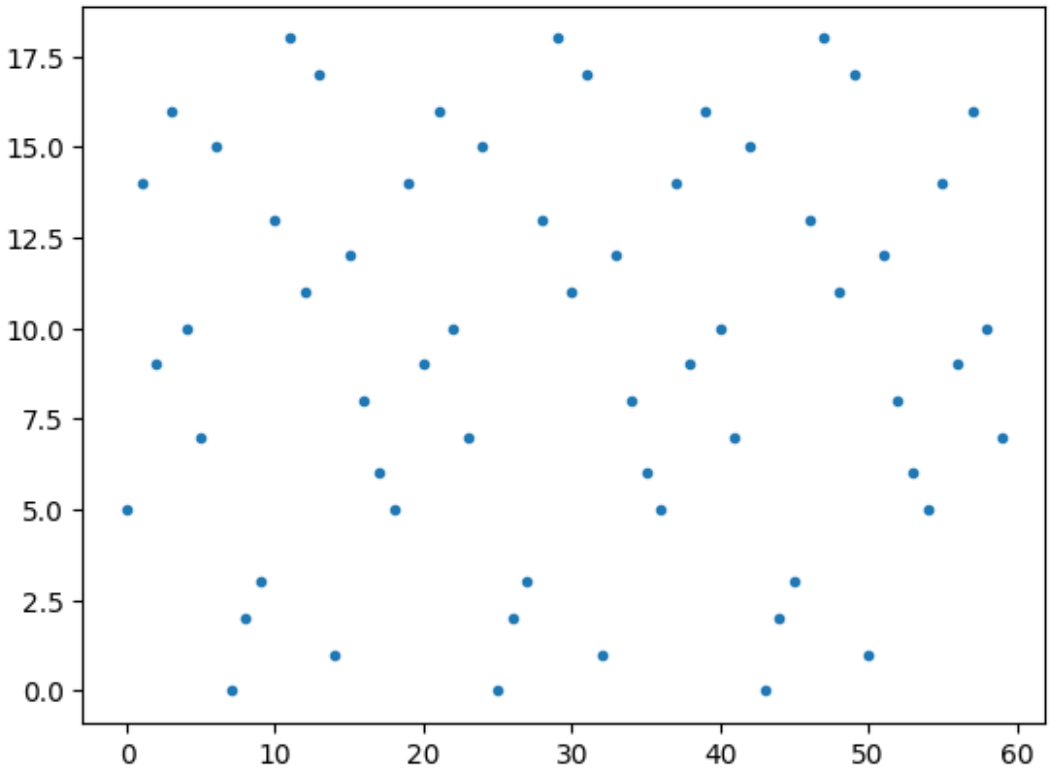
```
[0.90337952 0.57367949 0.00287033 0.61714491 0.3266449  0.5270581 ]
[0.8859421  0.35726976 0.90853515 0.62336012 0.01582124 0.92943723]
[0.69089692 0.99732285 0.17234051 0.13713575 0.93259546 0.69681816]
[0.06600017 0.75546305 0.75387619 0.92302454 0.71152476 0.12427096]
```

```
s = 5 #seed
a = 10 #multipliter
c = 2 #increment
m = 19 #modulus

n = 60
x = np.zeros(n)
x[0] = 5
for i in range(1, n):
    x[i] = (a * x[i-1] + c) % m

plt.plot(x, ".")
```

```
[<matplotlib.lines.Line2D at 0x271bd99ed20>]
```

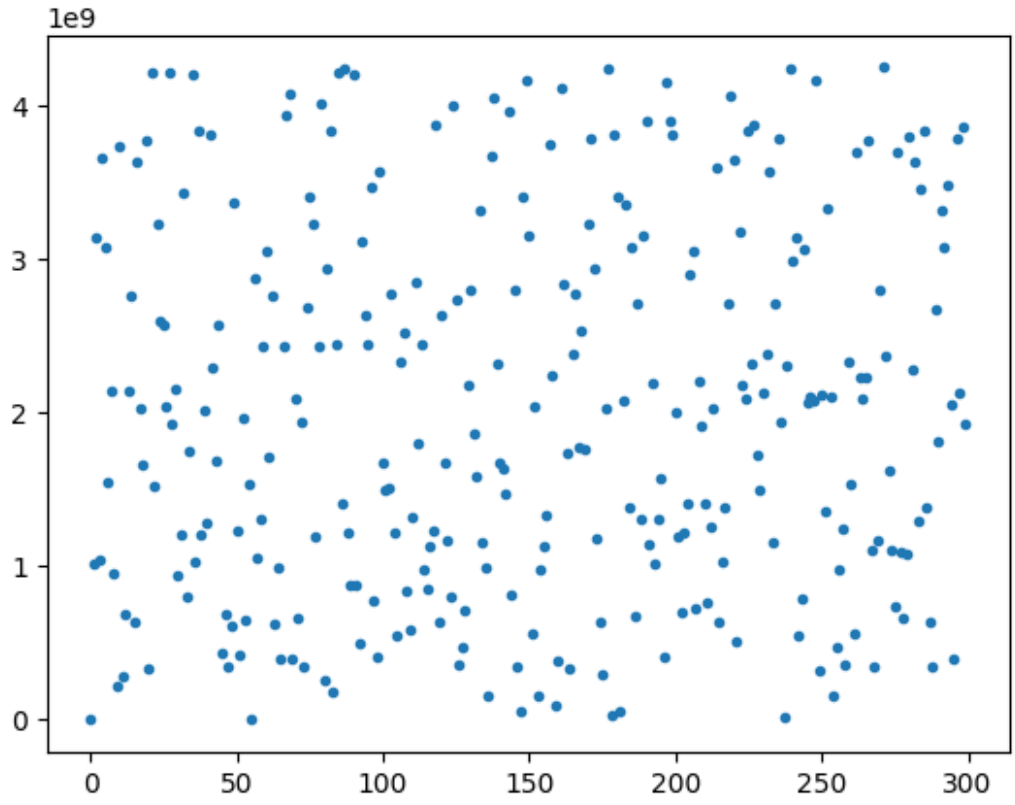



```
s = 30
a = 1664525
c = 1013904223
m = 2**32
```

```
n = 300
x = np.zeros(n)
x[0] = 5
for i in range(1, n):
    x[i] = (a * x[i-1] + c) % m

plt.plot(x, ".")
```

```
[<matplotlib.lines.Line2D at 0x271bda103b0>]
```




```
toss = np.random.choice([0, 1, 10])
print(toss)
```

```
10
```

```
N = 30 # number of samples (tosses)
M = 30 # number of experiments

expec_list = []
for j in range(M):
    toss_list = []
    for i in range(N):
        toss = int(np.random.choice([0, 1]))
        toss_list.append(toss)
    expec = np.array(toss_list).sum()/N
```

```
    expec_list.append(expec)
    print(toss_list, expec)

expec_list = np.array(expec_list)
```

```
[0, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0,
0, 0, 1, 0, 1, 1, 1] 0.4666666666666667
[1, 0, 0, 1, 1, 1, 0, 0, 0, 1, 1, 0, 1, 1, 1, 1, 1, 0, 0, 1, 1, 0, 1,
0, 1, 0, 1, 1, 0, 0] 0.5666666666666667
[1, 1, 1, 1, 0, 0, 1, 1, 0, 0, 1, 1, 1, 0, 0, 0, 0, 1, 1, 0, 0, 1, 0,
1, 0, 0, 0, 1, 1, 0] 0.5
[0, 0, 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 0,
1, 1, 0, 1, 0, 1, 1] 0.6333333333333333
[0, 1, 1, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0, 1, 1, 0, 1, 0, 0, 0, 0,
0, 0, 0, 0, 0, 1, 0] 0.4
[1, 1, 0, 0, 1, 1, 1, 1, 1, 0, 1, 1, 0, 1, 1, 1, 1, 0, 0, 1, 0, 1, 0,
0, 0, 1, 0, 1, 0, 1] 0.6
[0, 0, 1, 1, 0, 1, 0, 0, 1, 0, 1, 1, 0, 1, 1, 0, 1, 1, 0, 1, 0, 1, 0,
1, 1, 1, 1, 1, 0, 0] 0.5666666666666667
[1, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 1, 0, 0, 1, 1, 0, 0, 1, 0, 1, 1, 1,
1, 0, 0, 1, 1, 0, 1] 0.5
[0, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 1,
1, 1, 1, 1, 1, 0, 1] 0.5
[1, 0, 0, 1, 0, 1, 1, 1, 0, 1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 0, 1, 1, 0,
0, 0, 0, 0, 1, 1, 0] 0.43333333333333335
[1, 0, 0, 0, 1, 1, 0, 1, 0, 1, 1, 1, 0, 1, 1, 1, 1, 0, 1, 1, 0, 0, 0,
0, 0, 0, 1, 1, 0, 0] 0.5
[1, 0, 1, 0, 0, 0, 0, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 0, 0, 1,
1, 0, 0, 1, 0, 1, 1] 0.3333333333333333
[1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1, 1, 1, 1, 1, 0, 0, 0, 1, 0, 1, 1, 0,
0, 0, 1, 0, 1, 1, 1] 0.5333333333333333
[0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1, 0, 0, 0, 0, 1, 1, 1, 0, 0,
1, 0, 0, 1, 1, 0, 0] 0.43333333333333335
[1, 0, 1, 1, 1, 1, 1, 1, 1, 0, 1, 0, 1, 0, 1, 1, 0, 0, 0, 1, 1, 1, 1,
0, 0, 1, 1, 1, 0, 1] 0.6666666666666666
[0, 0, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0,
1, 0, 1, 1, 1, 0, 0] 0.5
[0, 0, 1, 0, 1, 1, 1, 1, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0,
1, 1, 1, 1, 0, 1, 0] 0.5
[0, 0, 1, 0, 0, 0, 1, 1, 0, 1, 1, 1, 1, 0, 1, 0, 0, 0, 1, 0, 0, 0, 0,
1, 0, 1, 1, 0, 0, 0] 0.4
[1, 0, 1, 0, 1, 1, 0, 1, 1, 0, 0, 0, 1, 0, 0, 1, 0, 0, 1, 0, 1, 0, 1,
1, 0, 1, 0, 0, 0, 0] 0.43333333333333335
[0, 0, 1, 1, 1, 1, 0, 1, 1, 0, 1, 0, 0, 0, 0, 1, 0, 0, 0, 0, 1, 1, 1,
1, 1, 0, 0, 1, 1, 1] 0.5333333333333333
[1, 1, 0, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1, 0, 0, 1, 1, 0, 0, 1, 0, 0, 0,
1, 0, 0, 0, 1, 1, 1] 0.5333333333333333
[1, 0, 1, 0, 0, 0, 1, 1, 0, 0, 0, 1, 0, 1, 1, 0, 0, 1, 1, 1, 0, 1, 0,
1, 0, 1, 1, 1, 0, 0] 0.5
[1, 1, 1, 0, 0, 0, 0, 0, 1, 0, 1, 1, 0, 0, 0, 1, 0, 0, 1, 1, 1, 1, 0,
```

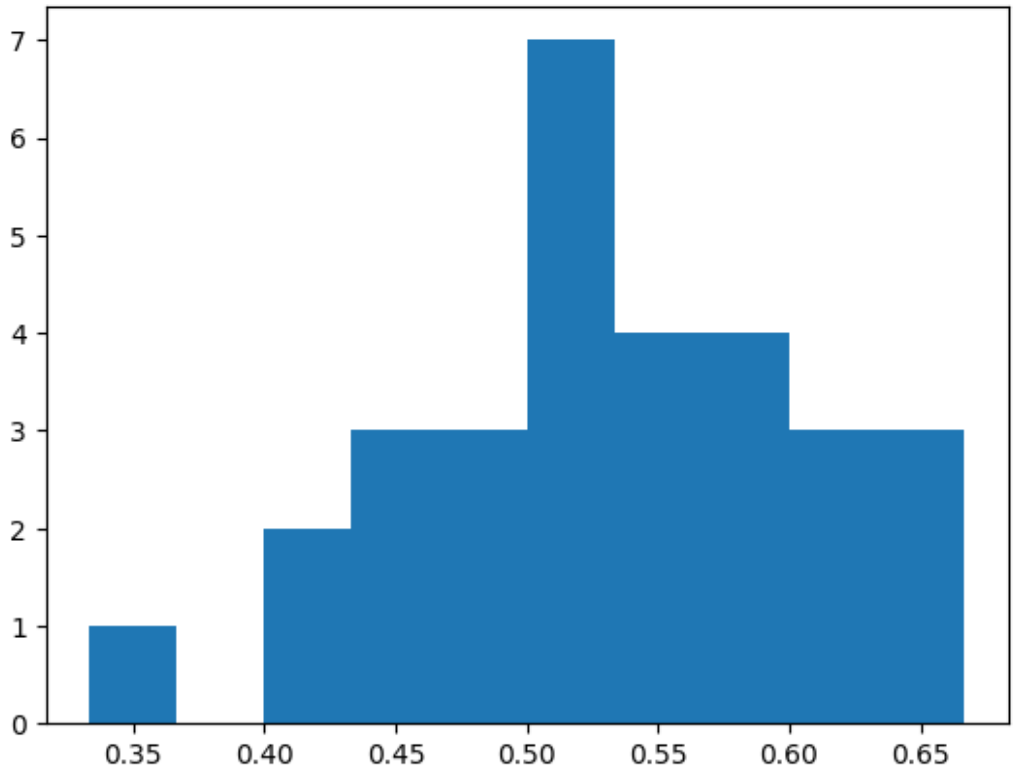
```
1, 0, 0, 0, 1, 0, 1] 0.4666666666666667
[0, 1, 1, 1, 0, 1, 1, 0, 0, 1, 0, 1, 0, 1, 0, 0, 1, 1, 0, 0, 1, 1, 0,
1, 0, 0, 1, 0, 0, 0] 0.4666666666666667
[0, 1, 1, 0, 1, 0, 0, 0, 1, 1, 1, 1, 0, 0, 0, 1, 1, 1, 0, 0, 0, 1, 0,
1, 1, 0, 0, 1, 1, 1] 0.5333333333333333
[1, 1, 0, 1, 1, 0, 1, 1, 1, 0, 0, 0, 0, 1, 1, 1, 1, 0, 1, 1, 0, 1, 0,
1, 0, 1, 0, 1, 1, 0] 0.6
[1, 1, 1, 1, 0, 0, 0, 1, 0, 1, 1, 0, 1, 0, 1, 1, 1, 1, 0, 0, 0, 1, 0,
0, 1, 1, 1, 0, 1, 0] 0.5666666666666667
[1, 0, 0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 1, 1, 0, 0, 1, 0, 0, 1, 1, 1, 0,
1, 1, 1, 1, 1, 1, 1] 0.6333333333333333
[0, 0, 1, 1, 1, 1, 1, 0, 0, 0, 1, 1, 1, 0, 0, 0, 1, 0, 0, 0, 1, 1, 1,
1, 0, 1, 1, 1, 1, 0] 0.5666666666666667
[1, 0, 0, 1, 1, 1, 0, 1, 1, 1, 0, 0, 1, 0, 1, 1, 1, 0, 0, 1, 1, 1, 1,
0, 1, 0, 0, 0, 1, 1] 0.6
```

```
expec_list
```

```
array([0.46666667, 0.56666667, 0.5       , 0.63333333, 0.4       ,
       0.6       , 0.56666667, 0.5       , 0.5       , 0.43333333,
       0.5       , 0.33333333, 0.53333333, 0.43333333, 0.66666667,
       0.5       , 0.5       , 0.4       , 0.43333333, 0.53333333,
       0.53333333, 0.5       , 0.46666667, 0.46666667, 0.53333333,
       0.6       , 0.56666667, 0.63333333, 0.56666667, 0.6       ])
```

```
plt.hist(expec_list)
```

```
(array([1., 0., 2., 3., 3., 7., 4., 4., 3., 3.]),
 array([0.33333333, 0.36666667, 0.4       , 0.43333333, 0.46666667,
        0.5       , 0.53333333, 0.56666667, 0.6       , 0.63333333,
        0.66666667]),
 <BarContainer object of 10 artists>)
```



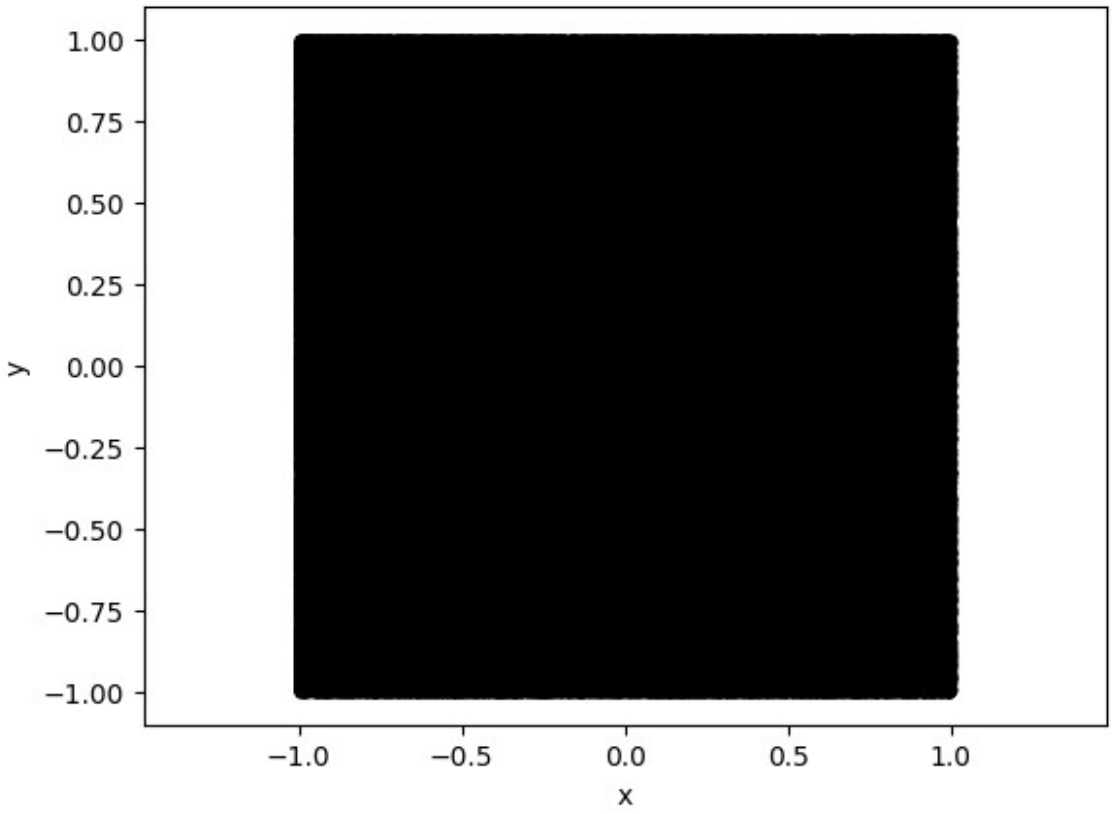

```
print(expec_list.mean(), expec_list.std())
```

```
N = 10**6
samples = np.random.rand(2, N)
xy = samples * 2 - 1.0 # scale sample points
r = np.sqrt(xy[0, :]**2 + xy[1, :]**2) # calculate radius
xy.shape
```

```
(2, 1000000)
```

```
plt.plot(xy[0, :], xy[1, :], "k.")
plt.axis("equal")
plt.xlabel("x")
plt.ylabel("y")
```

```
Text(0, 0.5, 'y')
```




```
incircle = (r <= 1)
incircle
```

```
incircle = (r <= 1)
count_incircle = incircle.sum()
print(count_incircle)
```

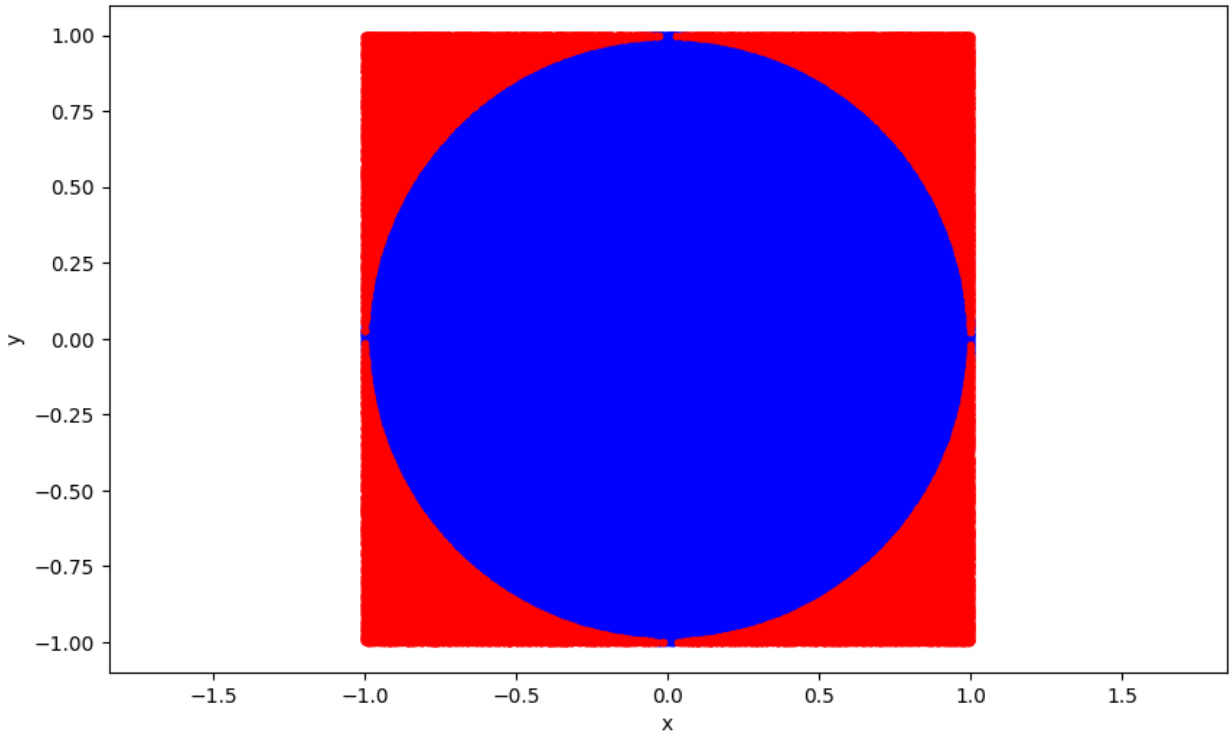
```
784954
```

```
A_approx = (2*2) * (count_incircle)/N
A_approx
```

```
np.float64(3.139816)
```

```
plt.figure(figsize=(10, 6))
plt.plot(xy[0, np.where(incircle)[0]], xy[1, np.where(incircle)[0]],
"b.")
plt.plot(xy[0, np.where(incircle==False)[0]], xy[1,
np.where(incircle==False)[0]], "r.")
plt.axis("equal")
plt.xlabel("x")
plt.ylabel("y")
```

```
Text(0, 0.5, 'y')
```


1.00
0.75
0.50
0.25
0.00
−0.25
−0.50
−0.75
−1.00
y
−1.5
−1.0
−0.5
0.0
0.5
1.0
1.5
x