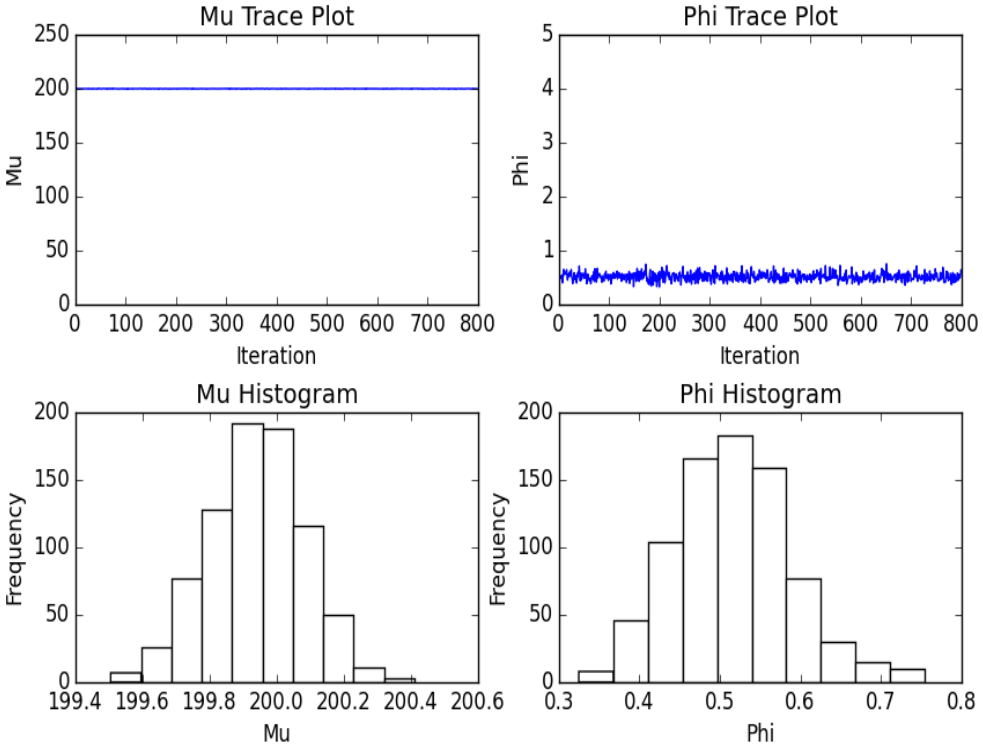
Prob6. Part2.

Code:

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

n=100
sample=np.random.normal(200,np.sqrt(2),n)
sample_average=np.mean(sample)

mu=[0]
phi=[5]

for t in range(1,800):
    mu.append(np.random.normal(sample_average,1.0/np.sqrt(n*phi[t-1])))
    beta=(sum((sample-mu[t])**2.0)/2)
    phi.append(np.random.gamma(n/2.0,1.0/beta))

print mu[:4]# [0, 199.89811778105192, 200.06969766049366, 200.10020634334668],burn-in time=2
print phi[:4]# [5, 0.4508917762751678, 0.5399732239135707, 0.5318182110342291],burn-in time=2

fig = pl.figure()
pl.subplot(2,2,1)
pl.plot(mu)
pl.title('Mu Trace Plot')
pl.xlabel('Iteration')
pl.ylabel('Mu')
pl.subplot(2,2,2)
pl.plot(phi)
pl.title('Phi Trace Plot')
pl.xlabel('Iteration')
pl.ylabel('Phi')
pl.subplot(2,2,3)
pl.hist(mu[2:],bins=10,color='white')
pl.title('Mu Histogram')
pl.xlabel('Mu')
pl.ylabel('Frequency')
pl.subplot(2,2,4)
pl.hist(phi[2:],bins=10,color='white')
pl.title('Phi Histogram')
pl.xlabel('Phi')
pl.ylabel('Frequency')
fig.subplots_adjust(hspace=.4)
pl.show()
```



The first four simulated Mu values include: 0, 199.89811778105192, 200.06969766049366, 200.10020634334668.
The first four simulated Phi values include: 5, 0.4508917762751678, 0.5399732239135707, 0.5318182110342291.
The burn-in time for Mu is 2, and the burn-in time for Phi is also 2. It's hard to point them out in the trace plots since the burn-in time is too short in a scale of 800 iteration.