AA274A Section 1

In [1]:
```
import numpy as np
import math

from scipy.integrate import quad
from scipy.optimize import minimize
import matplotlib.pyplot as plt
```

Define a sin from numpy

In [2]:
```
# Define a sin function using NumPy
def sin_f(x):
    return np.sin(x)
```

Find the minimum of the sin using SciPy

In [3]:
```
# Find the minimum of the function using SciPy
x0 = 0.5 #np.array([1.3, 0.7, 0.8, 1.9, 1.2])
res = minimize(sin_f, x0, method='nelder-mead', options={'xtol': 1e-8, 'disp':
True})
print("\n The minimum of sin(x) is: {}".format(res))
```

```
Optimization terminated successfully.
         Current function value: -1.000000
         Iterations: 34
         Function evaluations: 69

 The minimum of sin(x) is:  final_simplex: (array([[-1.57079632],
       [-1.57079633]]), array([-1., -1.]))
           fun: -1.0
       message: 'Optimization terminated successfully.'
          nfev: 69
           nit: 34
        status: 0
       success: True
             x: array([-1.57079632])
```

Integrate the function from [0, 1]

In [4]:
```
# Integrate the function from [0; 1] using SciPy
I = quad(sin_f, 0, 1)
print("\n Integral of sin(x) from x=0..1: {}".format(I[0]))
```

```
 Integral of sin(x) from x=0..1: 0.459697694132
```

Plot it

In [5]:
```
# Plot the function from 0-2pi
x = np.arange(100)*math.pi / 50
y = sin_f(x)
plt.plot(x,y)
plt.title("Line plot")
plt.xlabel("x")
plt.ylabel("sin(x)")
plt.show()
```