[1]:
```
import numpy as np
from scipy.constants import *
import IPython.display
import PIL.Image
from sympy import Matrix, MatrixSymbol, Eq, Symbol,init_printing
from IPython.display import display, Math, Markdown, Latex
init_printing()
```

[2]:
```
aps = [{'ssid': 'D', 'quality': '70/70', 'signal': -24, 'x' : 0.30, 'y' : 1.90} ,
{'ssid': 'C', 'quality': '70/70', 'signal': -25, 'x' : 8.40, 'y' : 1.50} ,
{'ssid': 'B', 'quality': '70/70', 'signal': -22, 'x' : 5.65, 'y' : 2.35} ,
{'ssid': 'A', 'quality': '70/70', 'signal': -23, 'x' : 4.30, 'y' : 5.30} ]
```

[3]:
```
tx_power = 19
```

[4]:
```
f = 2415000000
def wave(freq):
    return speed_of_light / freq;
```

[5]:
```
gamma = 3.5
d0 = 10
```

[6]:
```
K = -20*np.log10(4*np.pi*d0 / wave(f))
K
```

[6]:
$$-60.106125923633996$$

[7]:
```
distances = {}
for rssi in aps:
    rssi['distance'] = np.exp(np.log(10) * (tx_power + K - rssi['signal'] )/(10␣
 ↪* gamma))*d0
    print (rssi)
```

```
{'ssid': 'D', 'quality': '70/70', 'signal': -24, 'x': 0.3, 'y': 1.9, 'distance':
3.245290250294361}
{'ssid': 'C', 'quality': '70/70', 'signal': -25, 'x': 8.4, 'y': 1.5, 'distance':
3.465971390949713}
{'ssid': 'B', 'quality': '70/70', 'signal': -22, 'x': 5.65, 'y': 2.35,
'distance': 2.8451861628068205}
{'ssid': 'A', 'quality': '70/70', 'signal': -23, 'x': 4.3, 'y': 5.3, 'distance':
3.03866005246217}
```

[8]:
```
matA = np.array([np.array([2*(aps[-1]['x'] - rssi['x']), 2*(aps[-1]['y'] -␣
 ↪rssi['y'])]) for rssi in aps[:-1:]])
matb = np.array([np.array([np.power(rssi['distance'],2) -
                           np.power(aps[-1]['distance'],2) -
                           np.power(rssi['x'],2) -
```

```
                                np.power(rssi['y'],2) +
                                np.power(aps[-1]['x'],2) +
                                np.power(aps[-1]['y'],2)]  ) for rssi in aps[:-1:]])
```

$$\mathbf{x} = \left(A^T A\right)^{\dagger} A^T b \tag{1}$$

[9]:
```
(x,y)=np.matmul(np.matmul(np.linalg.pinv(np.matmul(matA.T,matA)),matA.T),matb)
```

[10]:
```
display(Markdown(rf'$x={x[0]}$'))
display(Markdown(rf'$y={y[0]}$'))
```

$x = 4.1153215374866585$

$y = 1.9272403974566836$

[11]:
```
Latex(r'$x={0}$'.format(x))
```

[11]:

$x = [4.11532154]$

$x = x$

[ ]:
```

```