Question 2. On 'linfit-head.py', write your own linear fitter. Use it for fitting data in the same file.

In [1]:

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

In [2]:

```
x=np.linspace(0,9,num=10)
y=np.array([1.1,2.2,3.1,2.9,4.1,6.7,9.8,11.1,13.2,15.3])
```

Least square method:

Let us consider a general linear functional L:

$$L(\mathbf{a}, \mathbf{x}) = x_0 + a_1 x_1 + a_2 x_2 + \cdots + a_p x_p$$

For given $(\mathbf{a_i}, y_i), i = 1, \cdots, n$, find $\mathbf{x}$ to minimize the following

$$\sum_{i=1}^{n} ||L(\mathbf{a_i}, \mathbf{x_i}) - y_i||_2^2$$

- $L(\mathbf{a_i}, \mathbf{x_i}) \approx y_i \Rightarrow AX \approx Y \Rightarrow \hat{X} = (A^T A)^{-1} A^T Y$

In [4]:

```
# Write your own linear fitter here

# assemble matrix A
A = np.vstack([x, np.ones(len(x))]).T

# turn y into a column vector
y = y[:, np.newaxis]
```

In [5]:

```
# Direct least square regression
alpha = np.dot((np.dot(np.linalg.inv(np.dot(A.T,A)),A.T)),y)
print(alpha)
```

```
[[ 1.62484848]
 [-0.36181818]]
```

In [7]:

```
# Make a plot of your fitting line here
plt.plot(x,y,'o')
plt.plot(x, alpha[0]*x + alpha[1], 'r')
plt.xlabel('x')
plt.ylabel('y')
plt.show()
```