Bitwise Operations

Implement bitwise operations: AND, OR, XOR.

In [1]:

```
import cv2
import matplotlib.pyplot as plt
import numpy as np
```

In [2]:

```
def bitwise_or  (a,b):
    height,width = a.shape[:2]
    c = np.zeros((height,width), np.uint8)
    for i in range(height):
        for j in range(width):
            c[i][j] = a[i][j] or b[i][j]
    return c

def bitwise_xor(a,b):
    height,width = a.shape[:2]
    c = np.zeros((height,width), np.uint8)
    for i in range(height):
        for j in range(width):
            c[i][j] = a[i][j] ^ b[i][j]
    return c

def bitwise_and(a,b):
    height,width = a.shape[:2]
    c = np.zeros((height,width), np.uint8)
    for i in range(height):
        for j in range(width):
            c[i][j] = a[i][j] and b[i][j]
    return c
```

In [3]:

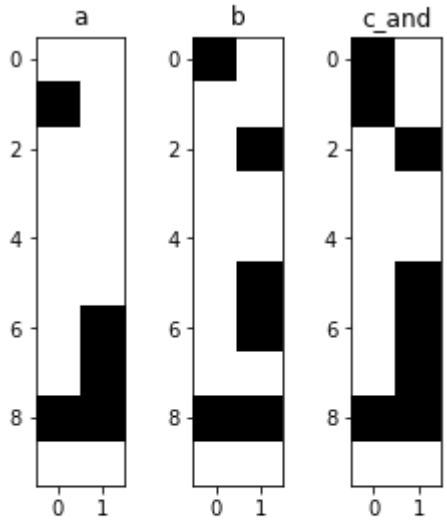
```
height,width  = 10,2
a = 1* (np.random.random((height,width)) > 0.5)
b = 1* (np.random.random((height,width)) > 0.2)
c_xor = bitwise_xor(a,b)
c_and = bitwise_and(a,b)
c_or = bitwise_or(a,b)
```

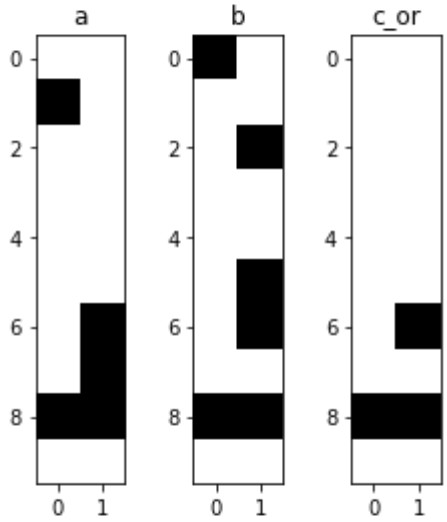
In [4]:

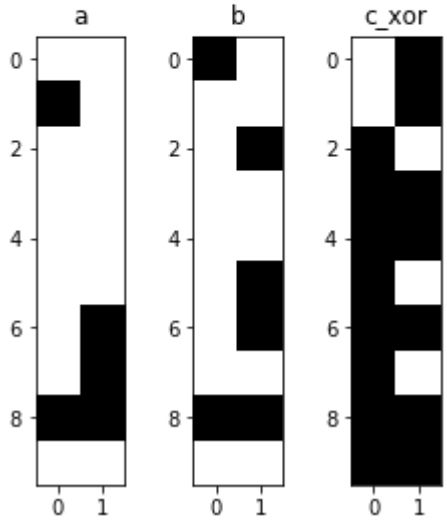
```
plt.figure(figsize=(4,14))

plt.subplot(331)
plt.title("a")
plt.imshow(a, cmap="gray")

plt.subplot(332)
plt.title("b")
plt.imshow(b, cmap="gray")

plt.subplot(333)
plt.title("c_and")
plt.imshow(c_and, cmap="gray")


plt.subplot(334)
plt.title("a")
plt.imshow(a, cmap="gray")

plt.subplot(335)
plt.title("b")
plt.imshow(b, cmap="gray")

plt.subplot(336)
plt.title("c_or")
plt.imshow(c_or, cmap="gray")

plt.subplot(337)
plt.title("a")
plt.imshow(a, cmap="gray")

plt.subplot(338)
plt.title("b")
plt.imshow(b, cmap="gray")

plt.subplot(339)
plt.title("c_xor")
plt.imshow(c_xor, cmap="gray")
```

Out[4]:

```
<matplotlib.image.AxesImage at 0x6960d30>
```


a
b
c_and
0
2
4
6
8
0
1


a
b
c_or
0
2
4
6
8
0
1


a
b
c_xor
0
2
4
6
8
0
1