

```
import numpy as np
```

```
def AND(x1, x2):
    x = np.array([x1, x2])
    w = np.array([0.5, 0.5])
    b = -0.7
    tmp = w[0]*x[0]+w[1]*x[1]+b
    print('tmp=',tmp)
    if tmp <= 0:
        return 0
    else:
        return 1
```

$(3*0.5 + 1*0.5 - 0.7)$

```
↳ 1.3
```

AND(0.71,0.71)

```
↳ tmp= 0.010000000000000009
1
```

AND(0,1)

```
↳ tmp= -0.19999999999999996
0
```

```
def OR(x1, x2):
    x = np.array([x1, x2])
    w = np.array([0.5, 0.5])
    b = -0.2
    tmp = w[0]*x[0]+w[1]*x[1]+b
    print('tmp=',tmp)
    if tmp <= 0:
        return 0
    else:
        return 1
```

OR(0,1)

```
↳ tmp= 0.3
1
```

```
def NAND(x1, x2):
    x = np.array([x1, x2])
    w = np.array([-0.5, -0.5])
    b = 0.7
    tmp = w[0]*x[0]+w[1]*x[1]+b
    print('tmp=',tmp)
    if tmp <= 0:
```

```

    tmp = 0
    return 0
else:
    return 1

```

NAND(1,1)

```

↳ tmp= -0.30000000000000004
0

```

```

def XOR(x1, x2):
    s1 = NAND(x1, x2)
    s2 = OR(x2, x2)
    y = AND(s1, s2)
    return y

```

XOR(0,0)

```

↳ tmp= 0.7
tmp= -0.2
tmp= -0.19999999999999996
0

```

XOR(1,1)

```

↳ tmp= -0.30000000000000004
tmp= 0.8
tmp= -0.19999999999999996
0

```

XOR(0,1)

```

↳ tmp= 0.19999999999999996
tmp= 0.8
tmp= 0.30000000000000004
1

```

