

## 461. Hamming Distance

```
class Solution(object):  
    def hammingDistance(self, x, y):  
        """  
        :type x: int  
        :type y: int  
        :rtype: int  
        """  
        x = x ^ y  
        y = 0  
        while x:  
            y += 1  
            x = x & (x - 1)  
        return y
```

$x \& x-1$  is to remove the last bit

$x \wedge y$  is XOR so it basically gives to you as a result the different bits.

<https://leetcode.com/problems/hamming-distance/discuss/94789/Beats-100-Python>