

```
/*  
两个整数之间的汉明距离指的是这两个数字对应二进制位不同的位置的数目。
```

给出两个整数 x 和 y ，计算它们之间的汉明距离。

注意：

$0 \leq x, y < 2^{31}$.

示例：

输入： $x = 1, y = 4$

输出：2

解释：

```
1   (0 0 0 1)  
4   (0 1 0 0)  
      ↑  ↑
```

上面的箭头指出了对应二进制位不同的位置。

来源：力扣（LeetCode）

链接：<https://leetcode-cn.com/problems/hamming-distance>

著作权归领扣网络所有。商业转载请联系官方授权，非商业转载请注明出处。

```
*/
```

分析：

- 显然将这两个整数取异或以后再统计结果数据中二进制位为1的个数即可，与 191题 https://github.com/Alimy/Break/LeetCodeStudy/tree/master/%E7%AE%97%E6%B3%95/191_%E4%BD%8D1%E7%9A%84%E4%B8%AA%E6%95%B0 思路基本一致。

方法一：C_Solution

```
int hammingDistance(int x, int y)  
{  
    unsigned int z = 0;  
  
    int count = 0;  
  
    z = (unsigned int)(x)^(unsigned int)(y);  
    while(z!=0xffffffff)  
    {  
        count++;  
        z = z|(z+1);  
    }  
    return (32-count);  
}
```

/*

执行结果：

通过

[显示详情](#)

执行用时 :4 ms, 在所有 C 提交中击败了82.65% 的用户

内存消耗 :6.8 MB, 在所有 C 提交中击败了65.13%的用户

*/

AlimyBreak

2019.07.23