Given an array, given a number, assume there must be two number in the array that their sum equal to the number, return their index in the given array.

e.g array = [2,7,11,15], target = 9

Answer = [0, 1] (2+7 = 9, return their index)

思路

暴力解決

從第一個數字開始, 查看除第一個數字以外的所有數字，如果加起來等於target，返回。否則loop完後，第二個數字開始，再和從第三個開始相加。如此類推。

過程會兩個for loop都會訪問array中所有的element,所以

Two pass hash table

建立一個hash table, 首先先把array中所有element以{element: index}的方式放進hash table裡面。然後loop array, 找有沒有hash table裡面有沒有target - i的差，有就返回。

這個方法是兩個O(n)的loop, complexity = O(n). 另外需要一個O(n)的hash table作為儲存。

One pass hash table

比起two pass,這個只訪問hash table一次。先loop through array，target - element, 看hash table 裡面有沒有這個差，沒有再把element以{element: index}的方式放進hash table。這裡可以訪問一次的原因是，由於target必然由nums裡面任意兩個element組成，假設x,y相加為target,那麼當x放進hash table時，y還沒放，那就繼續loop，但輪到y時，由於x已經放進hash table裡面了，所以能立刻返回，y不需要放進hash table裡面。

#enumerate(array) -> i = index, num = element

這個方法是兩個O(n)的loop, complexity = O(n). 另外需要一個O(n)的hash table作為儲存。