我分享一下寫Leetcode 撞牆期的心得，

/\*練習 Leetcode 還是建議先看題目的解答，\*/

/\*等待 Leetcode 撞牆期渡過了，再自已想解題的部份 \*/

網路上有些人認同用Leetcode 去判斷軟體工程師的程度; 有些人則持否定的看法。

無論是那一方都有很充足的理由，

而我是把Leetcode 當成工程師出題目給工程師的解題的遊戲。

雖然看起來像遊戲，但是要解Leetcode 題目其實還滿難的。

因為如果你沒有練習過類似的Leetcode 題目，你就會寫不出來。

所以，有時候考試運氣也是很重要的，考試的考題剛好是你有寫過的題目，因為你有練習過它所以解題的成功率會比較高。

----關於寫Leetcode 的撞牆期--------

有些人是三個月;

有些人是六個月;

有些人是兩年。

例如我的Leetcode 撞牆期是六個月，

我的前六個月基本上都是一直在看考題跟解答，

自已靈機一閃寫出題目的題目數量是基本上是0題。

因為題目都看不懂了，怎麼解題。

每個人剛開始寫 Leetcode 大多會遇到

1. 看不懂英文題目

2. 看懂題目，但是寫不出來

3. 看懂題目，寫出解案

就這三個階段。

而練習 Leetcode 產生的撞牆期是在 1.看不懂英文題目，主要還是因為「英文專有名詞」出現在題目中太多了，導致不理解題意。

這也就是很多人放棄寫Leetcode 的原因。

但是，不要放棄，只要渡過 Leetcode 撞牆期，解題的速度與靈感會快很多。

-----------------------------

最近我寫到一題Leetcode 是一個二維陣列，如果移動到陣列的邊緣(edge)，就要轉向。

轉向也有相對應的方法: 直行->down->left->up->right

這個題目的設計就跟 2D video game 的地圖資訊設定是一樣的，我覺得滿有趣的。

以上是我的心得，

撞牆期是心理學的一種;

你是會寫程式的;

但是你內心的魔鬼控制了你;

讓你害怕了。

要相信自已。

希望大家都能順利渡過撞牆期。