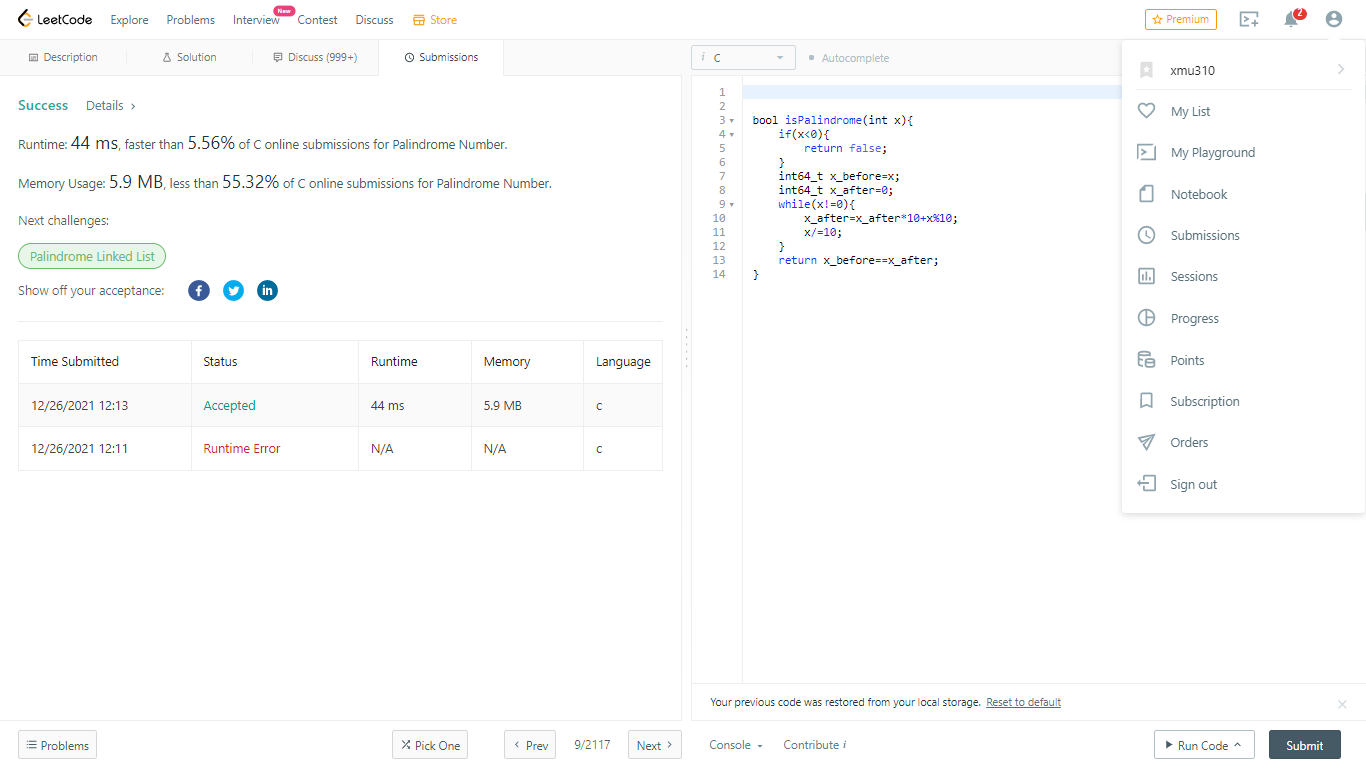
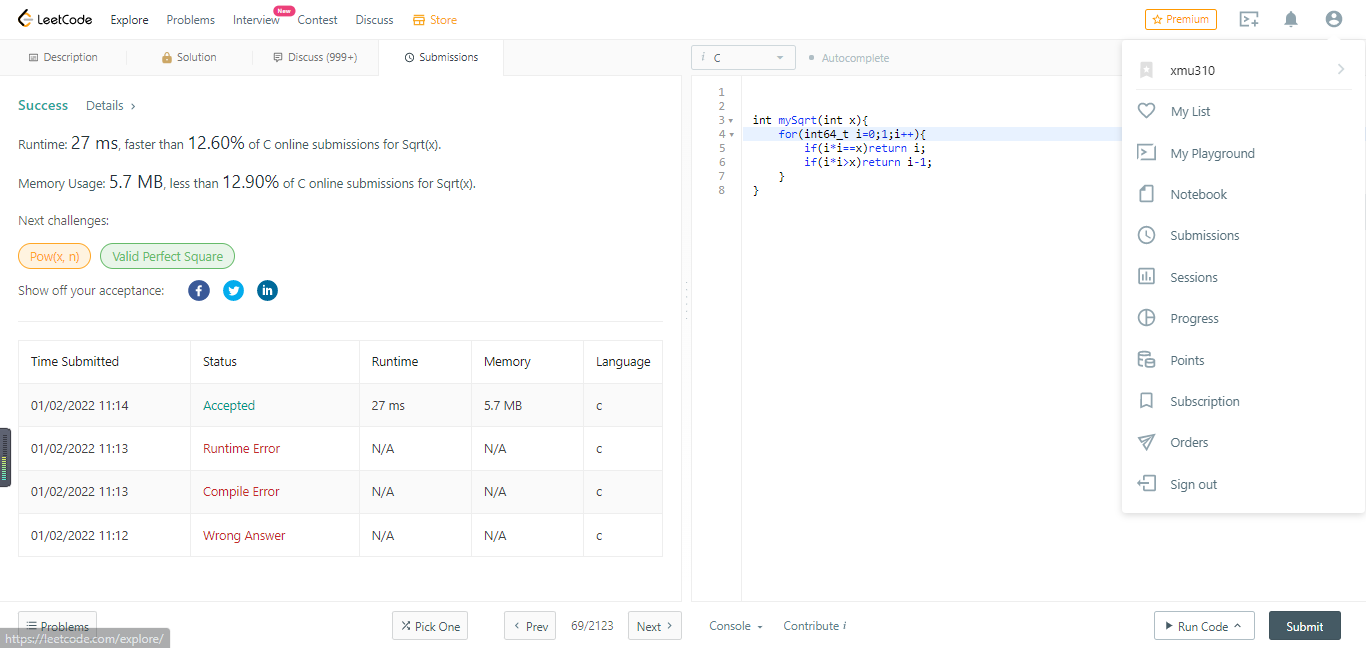
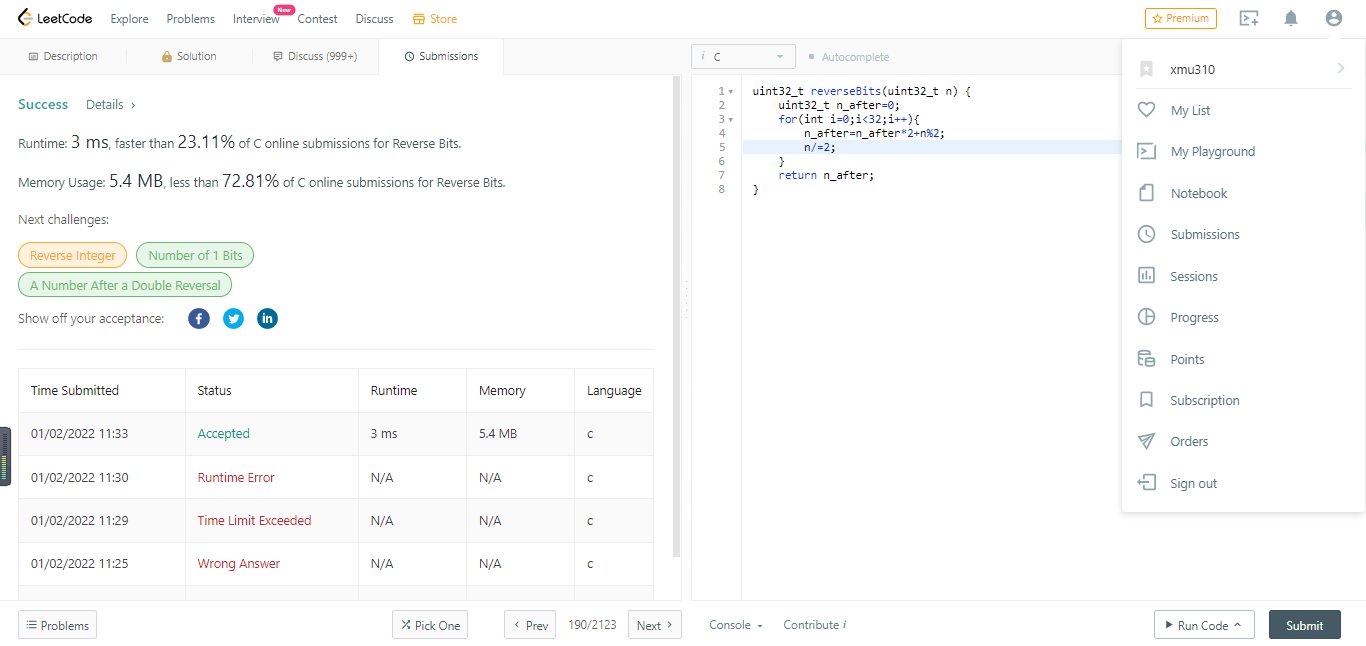
Bonus(41047049S李邦安)

1.Palindrome Number

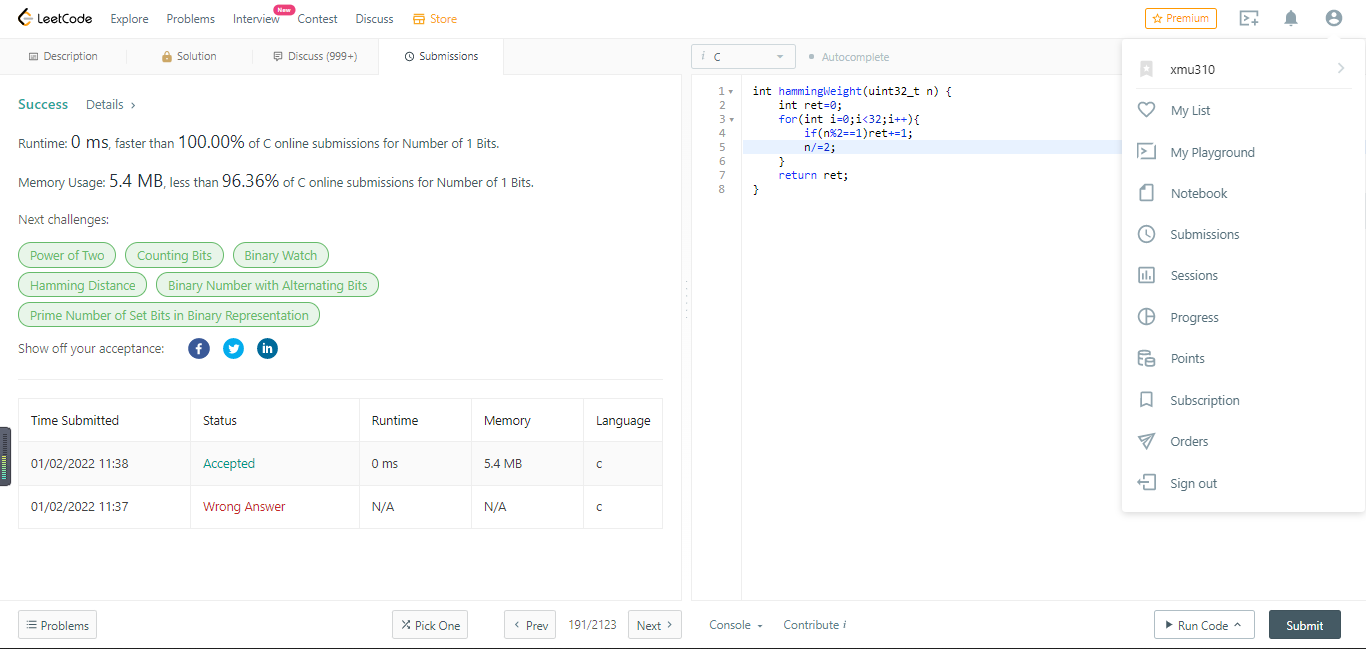
說明:這題要判斷傳入參數是否為回文數。因為負數無法形成回文數，因此先排除負數，接著創建兩個變數，x\_before代表傳入的原始參數數字，x\_after代表原始參數數字逆序後的結果，接著利用while迴圈將參數的數字逆序存入x\_after，最後比較x\_before等不等於x\_after，就知道這個數字是不是回文數了。

2.Sqrt(x)

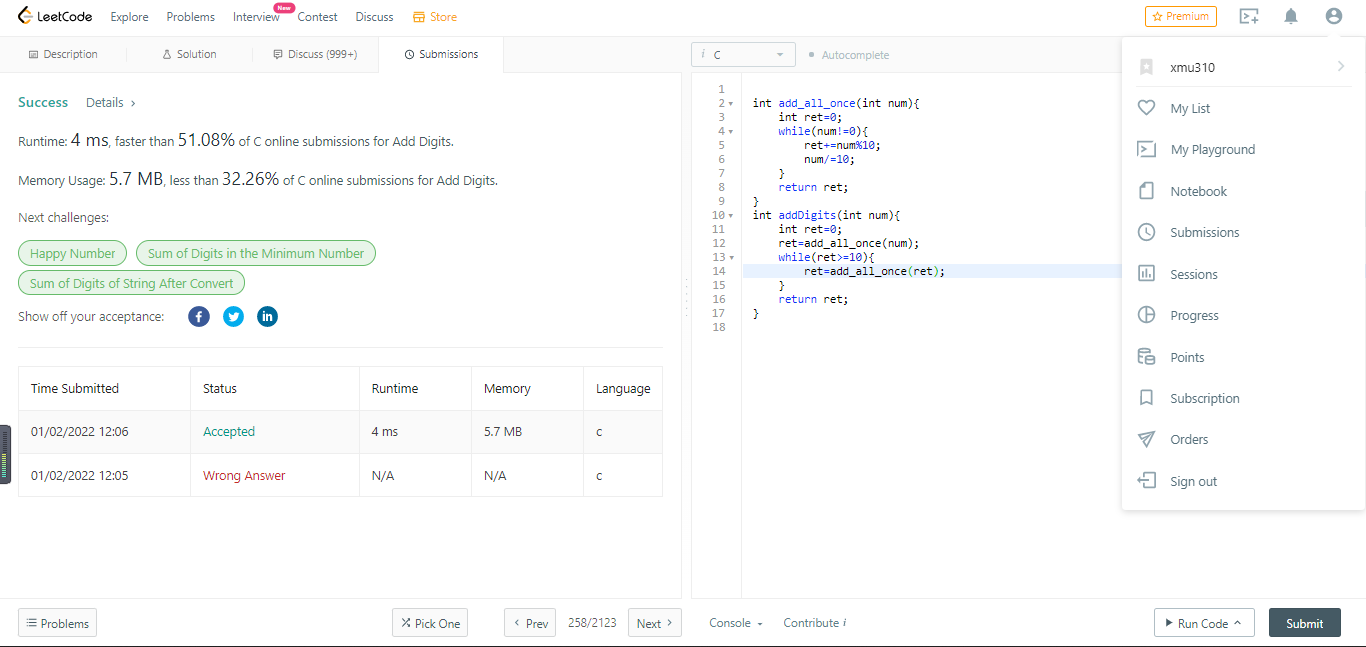
說明:這題是給定非負整數，並找出傳入參數的平方根整數部分。所以我寫了一個for迴圈，迴圈每執行一次，i就會+1，每跑一次迴圈，就看有沒有符合i\*i等於或大於x，如果兩者等於就回傳i(i恰好是參數的平方根)，如果i\*i>x就回傳i-1(因為此時的i大於參數實際平方根，而我們只需要實際平方根的整數部分)，如果條件皆不符合，迴圈就繼續跑，直到條件成立為止。

3.Reverse Bits

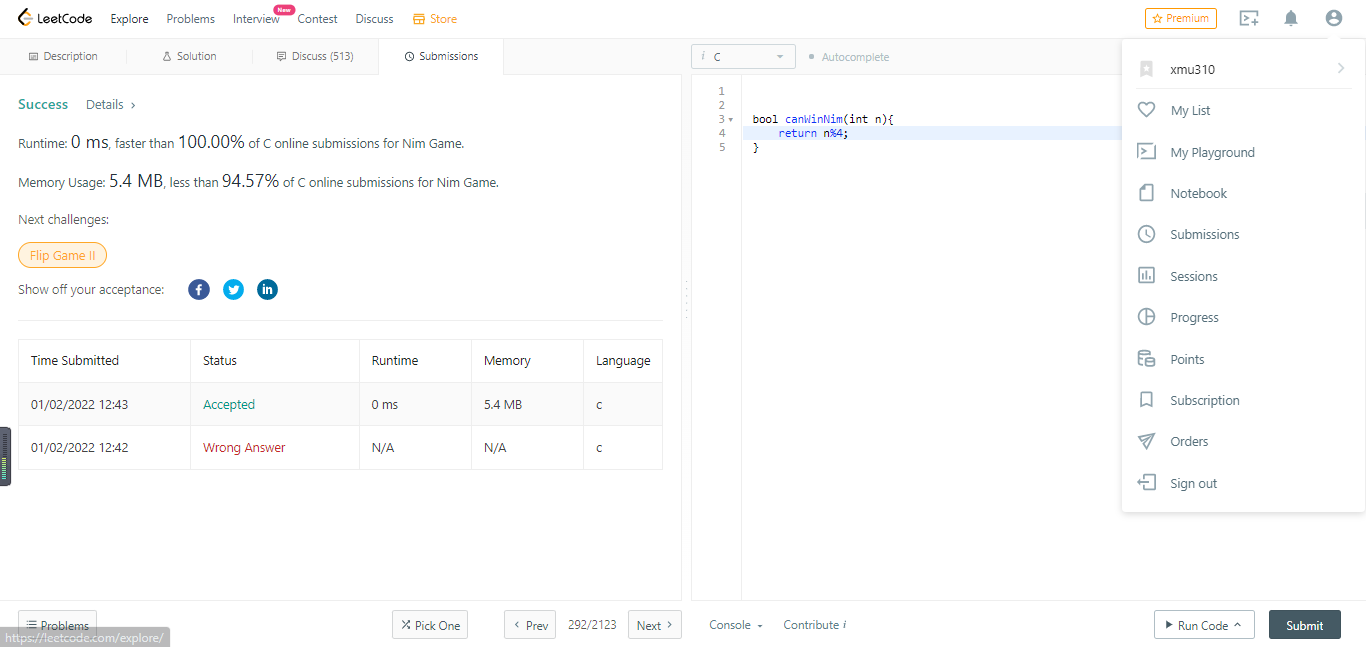
說明:題目要求反轉32bit的數字。因此我先定義了n\_after，代表反轉後的結果，然後我寫了一個for迴圈，迴圈次數32次，把參數數字的位數一個一個逆序分配給n\_after，這樣最終結果就會是反轉後的結果。

4.Number of 1 bits

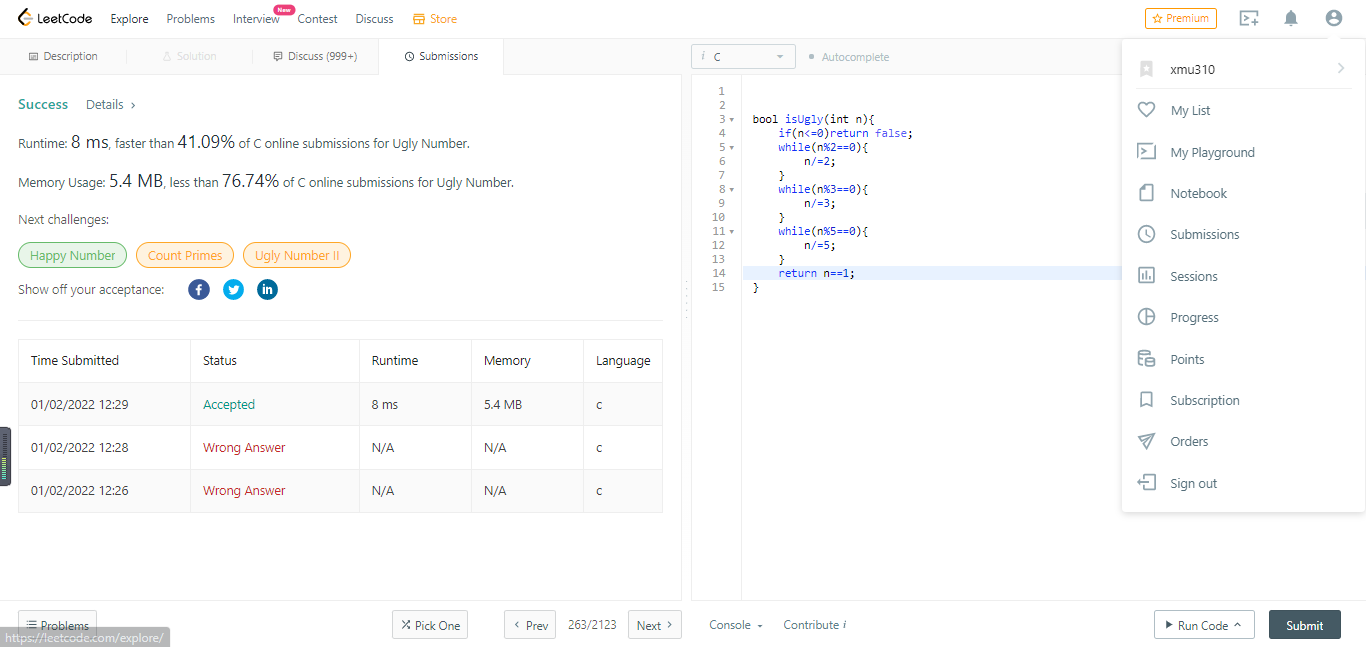
說明:這題給定一個32bit無符號數，要求我們找出32bits中有幾個1。所以這題也很簡單，我先定義一個變數叫ret，代表回傳的結果，然後寫一個for迴圈，總共跑32次，每一次判斷某一個位元是不是1，是的話ret就+1，否則不變，最後ret得到的結果就是32bits中1的數量了。

5.Add Digits

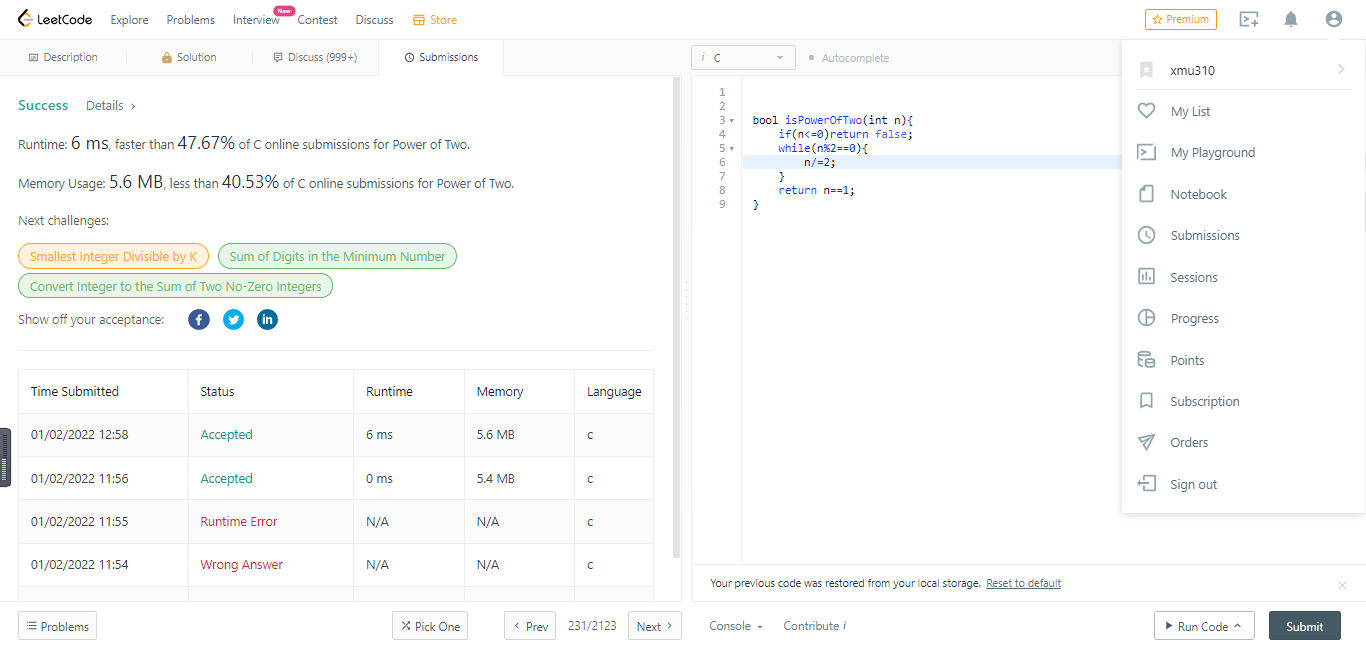
說明:這題給定一個數字，要求我們算出數字的位數和，如果數字和>=10，就再進行一次位數和，直至位數<10，(ex.99的位數和=9+9=18=>1+8=9)。所以要解決這題，我寫了兩個函式，一個是addDigits(主要的函式)，另一個是add\_all\_once(僅執行一次位數和)。add\_all\_once的實現方式很簡單，把位數用while迴圈一個一個位數加起來就好，最後回傳位數和。而addDigits這個函式中第一次先執行一次add\_all\_once(因為位數和至少要運行一次)，然後用while迴圈判斷位數和是否>=10，有的話就再進行一次add\_all\_once，直到條件不成立，最後得到的結果(ret)就是題目要求的位數和。

6.Nim Game

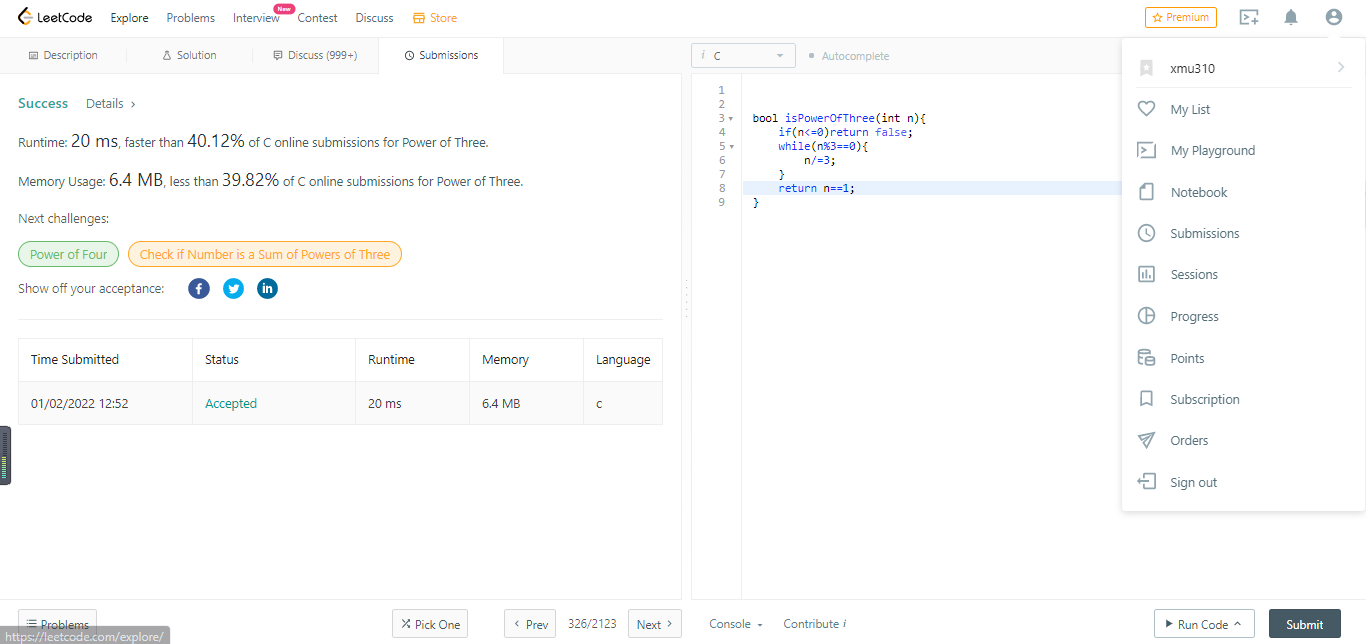
說明:Nim是一種遊戲，雙方輪流從石堆中拿1-3顆石頭，看誰拿到最後一顆石頭就贏了。這題給定Nim這個遊戲中的石頭初始數目，要我們判斷先手是否有贏的可能性。因為以前數學課有提過這個遊戲，因此其實判斷輸贏對我來說很簡單，只要把石頭數目對4取模，如果整除就必輸，沒整除就必贏，(假設雙方皆足夠理性)。原理是因為如果沒整除，那先手可先拿走石頭除以4的餘數顆，讓剩下的石頭成為4的倍數，這樣不管對方怎麼拿，自己都有應對的方法讓石頭繼續保持在4的倍數(ex.1->3,2->2,3->1)，這樣最後一顆石頭必定能被先手拿到。但如果一開始石頭個數被4整除，那先手和後手的立場就對調了，最後會變成後手獲勝，原理就是這樣。

7.Ugly Number

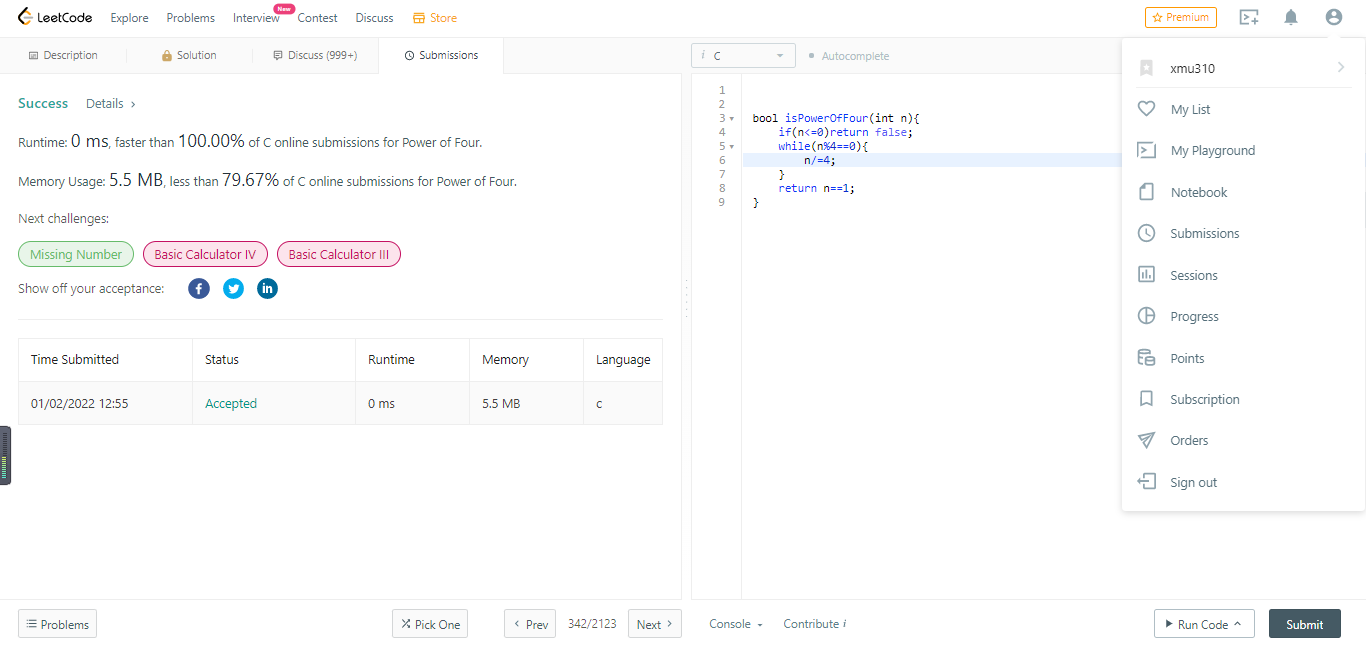
說明:這題要判定給定的數字是不是ugly number(正整數且質因數僅有2 3 5)。要判定其實很簡單，先判斷是不是正數，不是就可以直接return false了，然後接著把數字用while迴圈輪流除以2 3 5，其中迴圈的條件是數字能被2 3 5整除，這樣當三個迴圈都執行完畢後，如果是ugly number，數字應該會被除到只剩1，所以判斷最後的數字是不是1就能判定了。

8.Power Of Two

說明:這題要判斷給定的整數是不是2的冪(2^n,其中n屬於整數)。要判斷是不是2的冪很簡單，因為2的冪恆大於0，所以先判斷數字是不是正數，不是的話就可以直接return false了，然後接著把數字用while迴圈除以2，其中迴圈的條件是數字能被2整除，這樣當迴圈執行完畢後，如果是2的冪，數字應該會被除到只剩1，所以判斷最後的數字是不是1就能判定了。

9.Power Of Three

說明:這題要判斷給定的整數是不是3的冪(3^n,其中n屬於整數)。要判斷是不是3的冪很簡單，因為3的冪恆大於0，所以先判斷數字是不是正數，不是的話就可以直接return false了，然後接著把數字用while迴圈除以3，其中迴圈的條件是數字能被3整除，這樣當迴圈執行完畢後，如果是3的冪，數字應該會被除到只剩1，所以判斷最後的數字是不是1就能判定了。

10.Power Of Four

說明:這題要判斷給定的整數是不是4的冪(4^n,其中n屬於整數)。要判斷是不是4的冪很簡單，因為4的冪恆大於0，所以先判斷數字是不是正數，不是的話就可以直接return false了，然後接著把數字用while迴圈除以4，其中迴圈的條件是數字能被4整除，這樣當迴圈執行完畢後，如果是4的冪，數字應該會被除到只剩1，所以判斷最後的數字是不是1就能判定了。