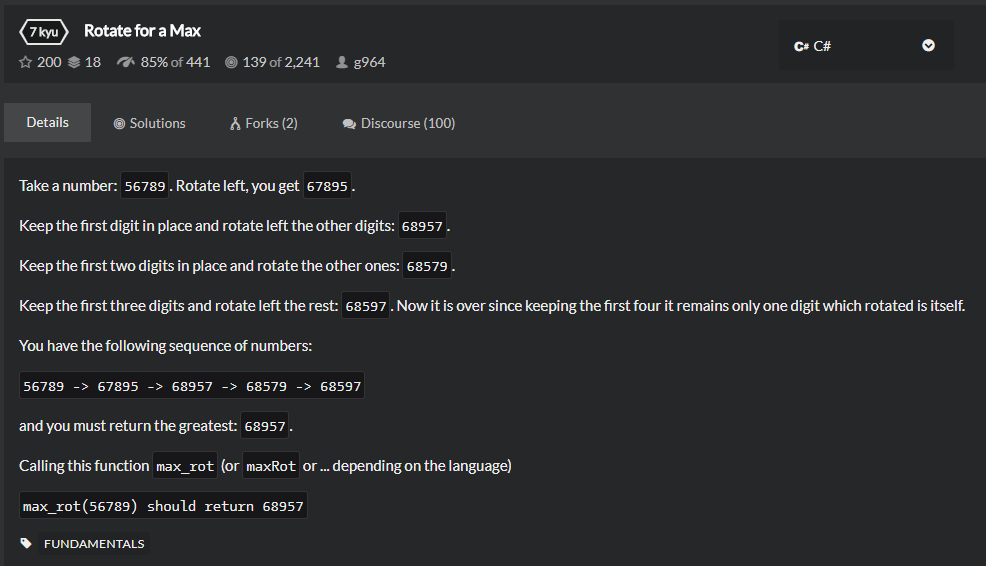
Day9. 轉吧轉吧七彩霓虹燈?－Codewars\_RotateForAMax

今天就用這題開啟吧!!

這題看起來需求就比較多了，終於可以不要只拆那一點點的東西了(?

貼圖實在是太累了，所以從今天開始我決定程式碼的部分用文字的方式試試看



這題題目的需求是，要轉轉轉轉轉，然後輸入的數字字串長度有多長，他就會轉幾次，什麼意思呢?

就是今天輸入了56789，我的數字往左移，5要移到最後面這時候就等於轉1次。

轉第2次的時候就要注意了，第1個數字就不動了，所以67895，往左轉但第1個數字不動，就變68957。

第3次也是，第1、2個數字不動，所以68957會變成68579。

題目看清楚了就來拆一下題目吧!

1. 定義現在已經轉了幾次，這樣才知道我要拿左邊第幾個位元的字原來座移動
2. 擷取目前最左邊的字元，往後挪到最後一個
3. 每進行挪動需要儲存這個數字到陣列
4. 取陣列中最大的數字，然後輸出

呼 這一次似乎好像複雜了一點點，耶，來開始寫Code吧!!

先寫測試囉

經過前幾天的練習，覺得要var那個又要var這個實在是很麻煩，其實就可以直接寫成這個樣子了。

[TestClass]

public class UnitTest1

{

[TestMethod]

public void Input\_1\_Should\_Be\_1()

{

Assert.AreEqual(1,Kata.MaxRot(1));

}

}

而Production Code 也就是老樣子會長成這個樣子

public class Kata

{

public static long MaxRot(long n)

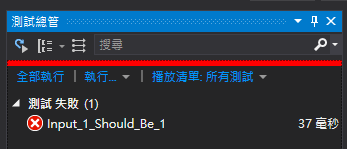
{

throw new System.**NotImplementedException**();

}

}

老樣子，跑個測試，沒過很正常，commit一下



接下來把Production Code改一下，用最小步伐讓Test Case Pass吧!

這一步改出來的Code就相對的醜很多，不過沒關係，忍住，先不Refactor他，先過現在的測試

public static long MaxRot(long n)

{

if (n.ToString().Length < 2)

{

return n;

}

var nString = n.ToString();

var nChar0 = nString[1];

var roatedN = int.Parse(nChar0 + nString.Substring(0, 1));

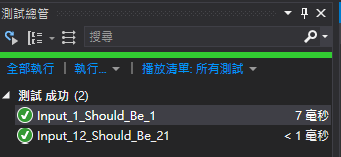
if (roatedN > n)

return roatedN;

return n;

}

接下來跑個測試，PASS!



來寫個第3個測試案例唄

[TestMethod]

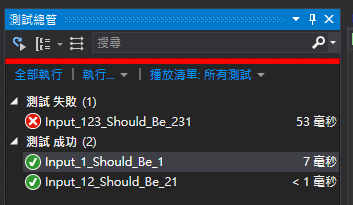
public void Input\_123\_Should\_Be\_231()

{

Assert.AreEqual(231,Kata.MaxRot(123));

}

接下來跑一下測試吧，紅燈! Commit!



接下來因為要考慮到較長數字的輸入，所以就用List add來做，順便判斷List內最大值了。

然後剛才再改的時候發生了很多次的錯誤，原來是因為string方法把remove誤用成substring，難怪第3個測試會一直失敗，於是就改成了這個樣子。

public static long MaxRot(long n)

{

var nString = n.ToString();

var resultList = new List<long>();

resultList.Add(n);

for (var i = 0; i < nString.Length - 1; i++)

{

var charIndex = nString[i];

resultList.Add(int.Parse(nString.Remove(i, 1) + charIndex));

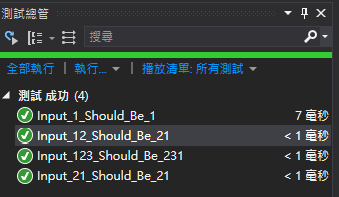
nString = nString.Remove(i, 1) + charIndex;

}

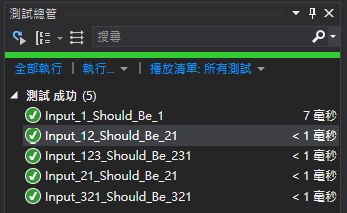
return resultList.Max();

}

現在我們完成他了，就來跑個測試，PASS，Commit一下唄!



Refactor之前來多寫幾個測項來驗證一下會不會錯唄。一樣pass，commit!



接下來refacotr一下唄!

其實仔細看就發現可以把很多去掉跟修飾，就會變成下面這個樣子

public static long MaxRot(long n)

{

var resultList = new List<long> { n };

var nString = n.ToString();

for (var i = 0; i < nString.Length; i++)

{

nString += nString[i];

nString = nString.Remove(i, 1);

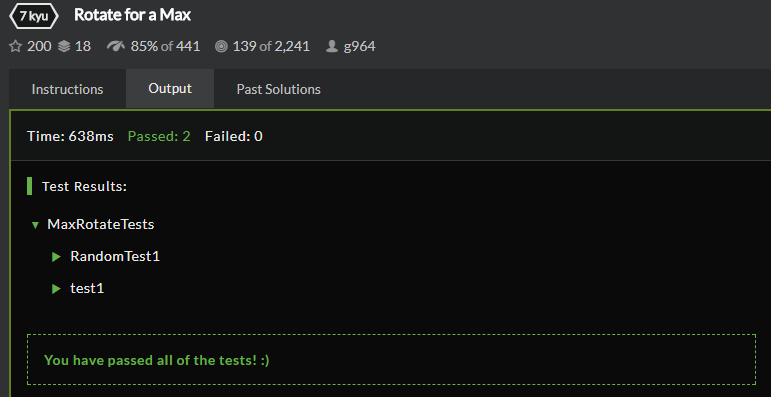
resultList.Add(long.Parse(nString));

}

return resultList.Max();

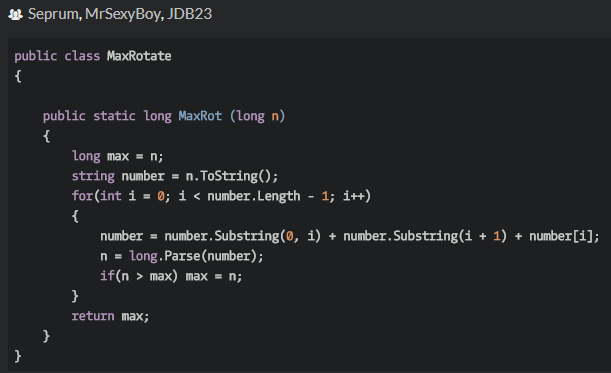
}

然後再Codewars上提交! Pass!!



通過!

老樣子我最喜歡的部分就是去看看別人寫這題寫得如何XD



哦這個方法也可以，我當初好像直覺思考就是要用到max，沒有思考到其實根本可以不用add，然後再max出來，中間就可以做掉了呢

看來還可以再加強XD

Git url : https://github.com/SQZ777/Codewars\_RotateForAMax

下一題，明天見!