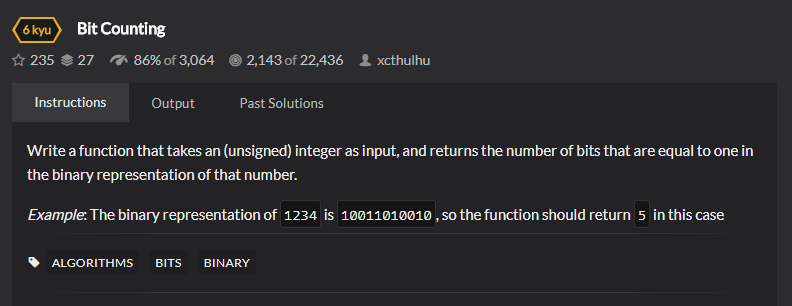
Day25. 史上最簡單6kyu(?) Codewars\_Bit Counting

今天題目超級簡單….

隨機挑到的XD

廢話不多so

題目長這樣



今天的題目需要將輸入的變數2進制化，然後運算其中含有幾個1。

現在就來拆一下題目吧。

1. 2進制化，Convert.toString方法。
2. 運算含有多少個1，Count方法。

一開始我們先來寫二進制化的測試吧!!

```

[TestMethod]

public void BinaryIt\_Input\_1\_Should\_Be\_1()

{

Assert.AreEqual("1", Kata.BinaryIt(1));

}

```

而Production Code 也就是老樣子會長成這個樣子

```

public static string BinaryIt (string s)

{

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

}  
```

老樣子，跑個測試，沒過很正常，紅燈，commit一下

接下來把Production Code改成這個樣子

```

public static string BinaryIt (string s)

{

return “1”;

}  
```

接下來跑個測試，PASS! Commit~

再來加上輸入2的測試。

```

[TestMethod]

public void BinaryIt\_Input\_2\_Should\_Be\_1()

{

Assert.AreEqual("10", Kata.BinaryIt(2));

}

```

而把return加上Conver.ToString方法之後的Production Code就會長這個樣子

```

public static string BinaryIt (string s)

{

return Convert.ToString(n, 2);

}

```

現在可以重構一下Production Code，變成一行的概念。

```

public static string BinaryIt(int n) => Convert.ToString(n, 2);

```

接下來要將Production Code運用剛採寫好的2進制化方法了，所以我們需要寫一個輸入2的測試。

```

[TestMethod]

public void Input\_2\_Should\_Be\_1()

{

Assert.AreEqual(1, Kata.CountBits(2));

}

```

這時候只要加上Linq語法Count就可以直接完成了

```

public static int CountBits(int n)

{

return BinaryIt(n).Count(x => x == '1');

}

```

重構一下變成這個樣子

```

public static int CountBits(int n) => BinaryIt(n).Count(x => x == '1');

```

現在已經完成所有需求啦!

所以現在就可以直接提交了。

所有的Production Code 會長這個樣子。

PS:其實是可以縮成一行的。

```

public static string BinaryIt(int n) => Convert.ToString(n, 2);

public static int CountBits(int n) => BinaryIt(n).Count(x => x == '1');

```

以下是今天所有的測試案例

```

[TestClass]

public class UnitTest1

{

[TestMethod]

public void BinaryIt\_Input\_1\_Should\_Be\_1()

{

Assert.AreEqual("1", Kata.BinaryIt(1));

}

[TestMethod]

public void BinaryIt\_Input\_2\_Should\_Be\_1()

{

Assert.AreEqual("10", Kata.BinaryIt(2));

}

[TestMethod]

public void Input\_2\_Should\_Be\_1()

{

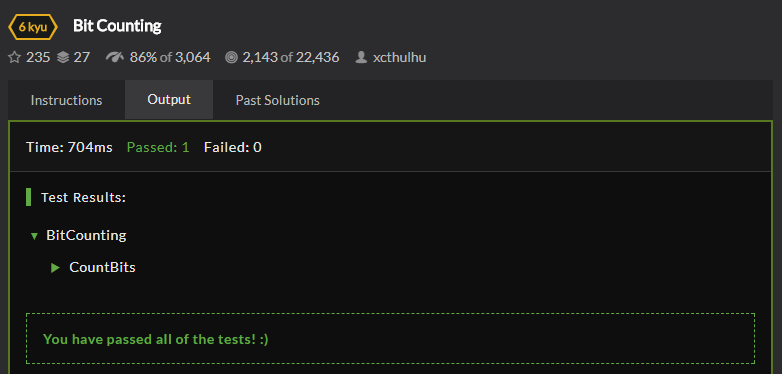
Assert.AreEqual(1, Kata.CountBits(2));

}

}

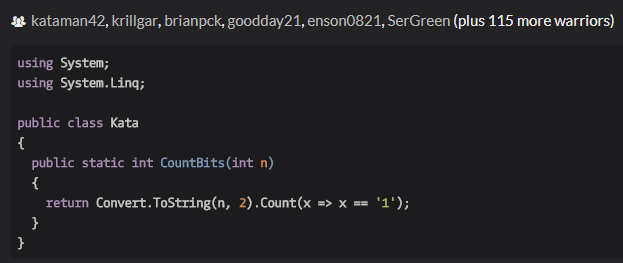
```

最後Submit到Codewars，Pass!



在Codewars上成功提交了~

來看一下其他人怎麼寫吧!



基本上跟我寫得一模一樣

今天這個題目其實就是讓我們知道

了解C#已經幫你寫好的東西有多重要

你不需要重造輪子

所以很快的你就可以運用已經有的方法去實作這樣子簡單的需求

快速又方便

Git url :

<https://github.com/SQZ777/Codewars_BitCounting>

Codewars Link:

<https://www.codewars.com/kata/bit-counting/train/csharp>

下一題，明天見!