

歐幾里得演算法被人們認為是史上第一個演算法。第一次編寫程式是愛達·勒芙蕾絲(Ada Byron)於1842年為巴貝奇分析機編寫求解伯努利微分方程的程式，因此愛達·勒芙蕾絲被大多數人認為是世界上第一位程式設計師。

所謂的歐幾里得算法(Euclidean Algorithm)，其實就是我們在國/高中學過的「輾轉相除法」。

你在國/高中學的應該是用直式的方式計算，計算的過程像下面這樣：

1	1034	893	6
	893	846	
3	141	47	
	141		
	0		

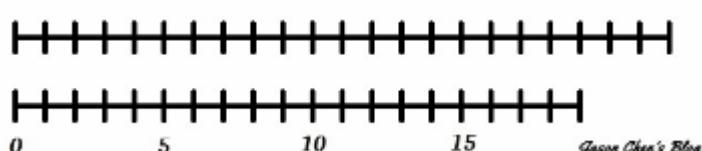
用這個方法演算求解的過程，我們在國小應該就學過一個概念，乘法可以視作累加；除法可以視作累減。

回到歐幾里得演算法，那為什麼像這樣利用輾轉相除的原理就可以找到最大公因數呢？

那我們利用上面的例子， $A=1034$, $B=893$ 來說明好了。

既然我們已求出 $\text{GCD} = 47$, $1034 / 47 = 22$, $893 / 47 = 19$,

那我們就用先 22 格跟 19 格分別來代表 A 跟 B，這樣應該比較好理解，參考下方 gif 圖：



可以看到動圖中兩數(A、B)不停的在相互做相除(累減)的動作，

1.剛開始 A 先被 B 減掉 1 次；

2.再來 B 被 A 的餘數 A' 減掉 6 次；

3.最後 A 的餘數 A' 再被 B 的餘數 B' 減掉 3 次 後餘 0 結束計算。

透過這樣就可以找到兩數之間的最小單位，一格($=47$)。

其實就是不停的用兩個數字相除取餘數，再用餘數輾轉下去相除，直至餘數為 0 時結束。最後附上，「歐幾里得算法」的實現代碼：

Euclidean Algorithm

```
int GCD(int A, int B){  
    int n = 1;  
    while( n != 0){  
        n = A % B;  
        A = B;  
        B = n;  
    }  
    return A;  
}
```