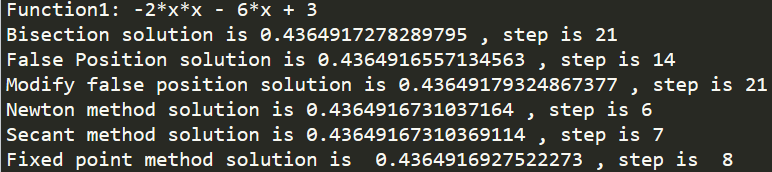
數值方法 HW1解方程式 資工三甲407261128 康智絜

1. 時間分析：迴圈次數

Bisection和Modify false position solution在函式1、2、3中是六個方法中迴圈需要跑最多次的，以Function1: -2\*x\*x - 6\*x + 3為例，若a取-1，b取2，Bisection就需要跑21次，相較之下Newton method solution僅需要跑6次，所需次數就差了三倍多，所以花的時間會比較多。



1. 空間分析：變數個數

Bisection：三個 (a, b, m)

False Position：四個 (a, b, m\_old, m\_new)

Modify False Position：六個 (a, b, m, Fa, Fb, Fm)

Newton Method：兩個 (a, delta)

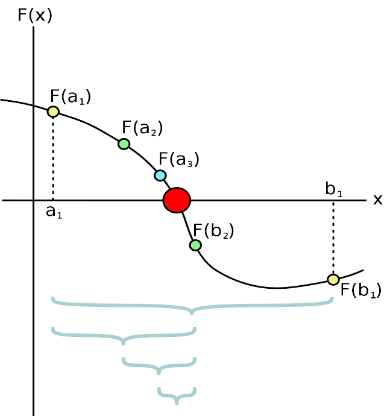
Secant Method：三個 (x1, x2, x3)

Fixed Point Method：兩個 (x0, x1)

結論：

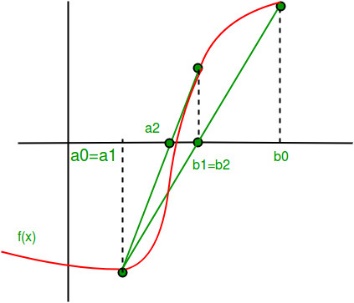
Modify False Position所需要的變數最多，需要六個，而Newton Method及Fixed Point Method所需要的變數最少，僅需要兩個。

1. 過程中的體會
2. Bisection



這個方法跟以前學的二分搜尋法雷同，將數列不斷的分為兩個部份，每次從分割的部份中取中間數比對，實際寫程式時覺得很熟悉。這個方法相對其他五個方法我覺得是比較容易懂並想的到的求根方法，但需要不斷的切割再切割才能找到它的根，並且誤差要小於10^-6，所以需要切非常多次，雖然code很短且好懂，但付出的時間代價蠻大的。

1. False Position



這個方法是利用兩端點不斷拉直線來找到其根，雖然迴圈次數比二分法少，但是卻不比二分法精確。這個方法和Secant Method很相似，迭代式一樣，差別只在於換值的部分。

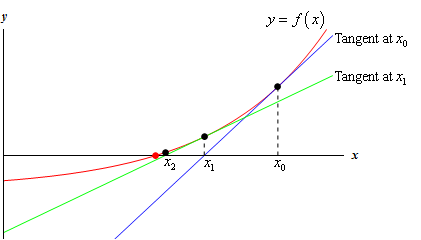
False Position的迭代式：

Secant Method的迭代式：

1. Modify False Position

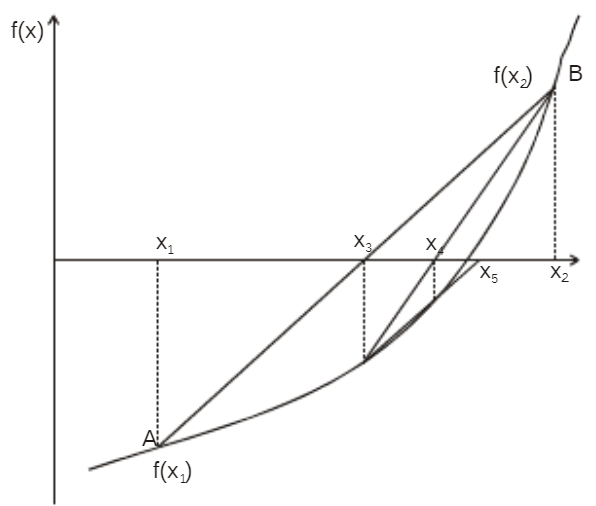
和False Position的差別是False Position的b的位子都沒有移動，而Modify False Position把b的位子往下降。而程式中False Position的code相較Modify False Position比較淺顯明瞭，而且迴圈次數的表現也比Modify False Position好。

1. Newton Method



這個方法不斷利用取切線斜率來找根，這個方法需要微分函式才能完成，相較其他方法比較麻煩，但是他的迴圈次數是最少的，而且程式碼非常短且易懂，但是缺點是函數必須可微分且沒有重根。

1. Secant Method



和Newton Method相似，不斷取兩點連線，但是它不需要取微分。迴圈次數表現僅次於Newton Method，程式碼也跟Newton Method差不多短，優點是不需要微分，且迴圈次數少。

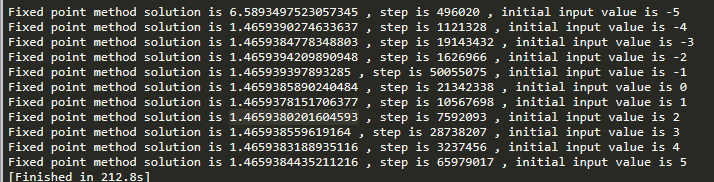
1. Fixed Point Method

這個方法要將函數化成g(x)，有三種方法可以將它畫成g(x)，分別是移項X^2、X、常數項，分別將三種寫成程式碼代入，以Function1: -2\*x\*x - 6\*x + 3為例，移常數項讓它變成-3/(-2\*x-6)的收斂速度是最快的，而移X項的速度比較慢，迴圈次數比較多，而移X^2的會產生ValueError: math domain error。而這個方法在六種方法中，迴圈次數是屬於居中的，缺點是需要額外化成g(x)。

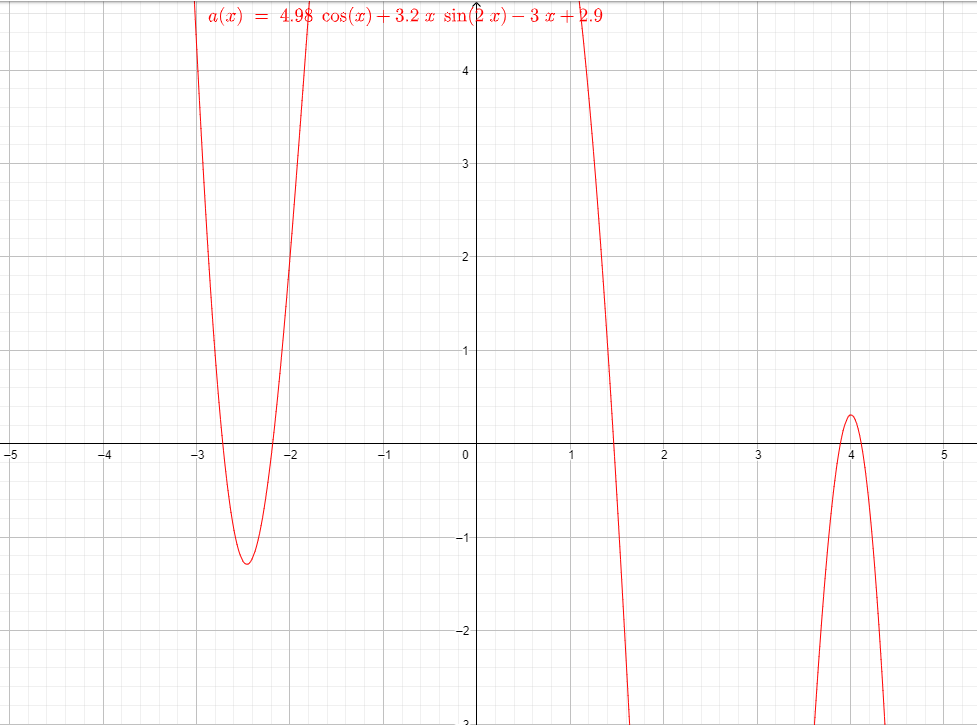
1. 共同題

實驗這六個方法後，我覺得三個最佳的方法是Secant method和Newton method以及False position method。我覺得Secant method最佳是因為他所需要跑的迴圈數已經是第二少的，而且不需要給特別的初值，相較之下雖然Newton method跑的迴圈是最少的，但是前提是要給他接近解的input value才能得到接近的根。False Position Method會成為第三是因為我覺得剩下的其他三個都不是很好。Bisection solution是很暴力的用迴圈跑很多次才得到的，所以所花的迴圈樹是最多的，而且是Newton method的好幾倍。Fixed point method所能得到的根有限，就這題來說他所能求得的解只有1.4659380201604593，效果有點差，而且迴圈必須跑超級多次，至少要10^5才能跑完，所以我覺得是最差的方法。而Modify False Position Method和False Position Method的迴圈數相較之下False Position Method比較少，所以又比Modify False Position Method表現好。

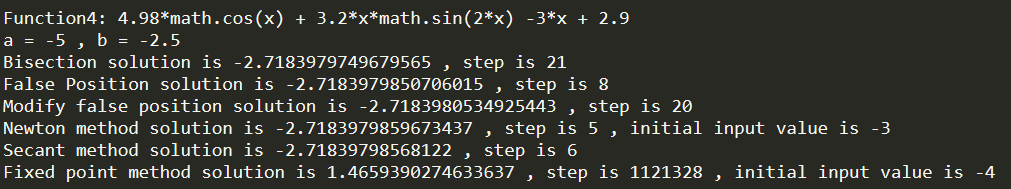
* 下圖為Fixed point method輸入不同初值所得到的解



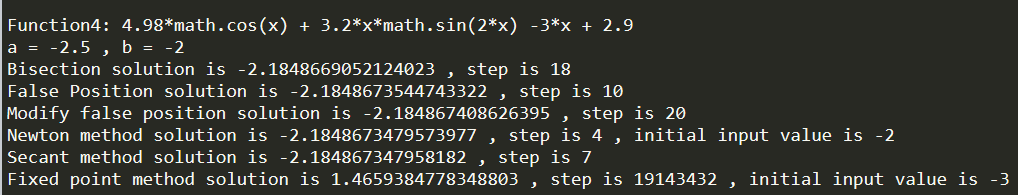
得到所有解的方法：藉由畫圖來知道解大約的位置，進而猜測a及b



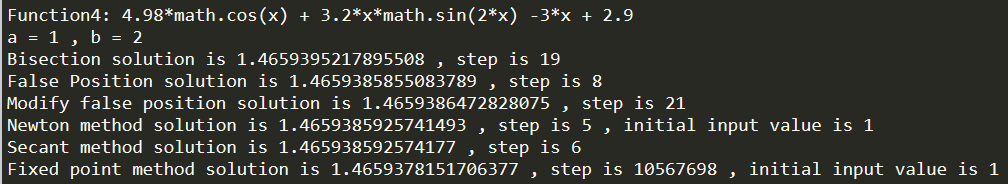
* 令a為-5，b為-2.5



* 令a為-2.5，b為-2



* 令a為1，b為2



* 令a為3，b為4

