

```
In [3]: %runfile C:/Users/jacks/OneDrive/Desktop/hw3/test.py --wdir
真實值: cos(0.75) = 0.7316888688738209
Degree 1: Approximated cos(0.75) = 0.7315914285714286, Error bound = 0.00015300000000000028
Degree 2: Approximated cos(0.75) = 0.7317163265306124, Error bound = 2.652000000000007e-06
Degree 3: Approximated cos(0.75) = 0.7317039556851311, Error bound = 3.513900000000013e-08
In [4]:
```

第一題結果如上，題目要求用 lagrange 跑 $\cos(0.75)$ 值，並給四個 x 跟 $f(x)$ ，將 4 個值分別帶入 lagrange 方程 x 最高次方分別 1,2,3，並用最靠近的點去做

$$E_n(x) = \frac{f^{(n+1)}(\xi)}{(n+1)!} \prod_{i=0}^n (x - x_i)$$

lagrange，error bound 是

```
In [45]: %runfile C:/Users/jacks/OneDrive/Desktop/hw3/test2.py --wdir
when  $x - e^{-x} = 0$  時,  $x \approx 0.567143$ 
real  $e^{-x}=0.5671436686275126$ 
```

第二題結果如上，題目要求找 $f(x)=0$ 的解，即用 $x-e^{-x}$ 當作 $f(x)$ 反向帶入 lagrange，每一項是 $(f_target - fx_values[j]) / (fx_values[i] - fx_values[j])$ ，target 是目標值=0，fx_values 是題目給的 $x-e^{-x}$ 。

```
In [46]: %runfile C:/Users/jacks/OneDrive/Desktop/hw3/test3.py --wdir
用python內建跑Hermite 插值法結果
當  $t = 10$  秒時，預測位置  $D \approx 768.96$  英尺，速度  $V \approx 74.64$  英尺/秒
車輛首次超過 55 mi/h 的時間為  $t \approx 3.15$  秒
車輛的預測最大速度為  $V\_max \approx 92.04$  英尺/秒

newton建Hermite 插值法結果
當  $t = 10$  秒時，預測位置  $D \approx 777.06$  英尺，速度  $V \approx 87.18$  英尺/秒
車輛首次超過 55 mi/h 的時間為  $t \approx 3.94$  秒
車輛的預測最大速度為  $V\_max \approx 87.35$  英尺/秒
```

第三題結果如上，題目要求跑那三個問題，主要問題在 Hermite polynomial，python 內建的結果與自己建 newton 法的 Hermite 插值法答案有出入，大概是他內部為了使 3 次方程式更加絲滑有做調整，newton 建法主要流程如下，將 t,d,v 先建兩次一樣的值到矩陣內，再填充插分商

$$f[t_i, t_{i+1}, \dots, t_j] = \frac{f[t_{i+1}, \dots, t_j] - f[t_i, \dots, t_{j-1}]}{t_j - t_i}$$

，即可建

程 hermite polynomial，再用建好的方程式跑三個問題即可