```
In [44]: %runfile C:/Users/jacks/OneDrive/Desktop/hw3/test.py --wdir
[0.76612909 0.74317044 0.73168887 0.7193015  0.69455151]
0.7661290921161424
Degree 1: Approximated cos(0.698) = 0.7661, Error bound = 2.909211614243734e-05
Degree 2: Approximated cos(0.698) = 0.7661, Error bound = 2.909211614243734e-05
Degree 3: Approximated cos(0.698) = 0.7661, Error bound = 2.909211614243734e-05
Degree 4: Approximated cos(0.698) = 0.7661, Error bound = 2.909211614243734e-05


0.7431704431584757
Degree 1: Approximated cos(0.733) = 0.7432, Error bound = 2.955684152428084e-05
Degree 2: Approximated cos(0.733) = 0.7432, Error bound = 2.955684152428084e-05
Degree 3: Approximated cos(0.733) = 0.7432, Error bound = 2.955684152428084e-05
Degree 4: Approximated cos(0.733) = 0.7432, Error bound = 2.955684152428084e-05


0.7316888688738209
Degree 1: Approximated cos(0.75) = 0.7320771428571429, Error bound = 0.00038827398332197394
Degree 2: Approximated cos(0.75) = 0.7317, Error bound = 1.1131126179120265e-05
Degree 3: Approximated cos(0.75) = 0.7317, Error bound = 1.1131126179120265e-05
Degree 4: Approximated cos(0.75) = 0.7317, Error bound = 1.1131126179120265e-05


0.7193015033391573
Degree 1: Approximated cos(0.768) = 0.7202999999999999, Error bound = 0.0009984966608426182
Degree 2: Approximated cos(0.768) = 0.719254751131222, Error bound = 4.675220793537438e-05
Degree 3: Approximated cos(0.768) = 0.7193, Error bound = 1.5033391572716326e-06
Degree 4: Approximated cos(0.768) = 0.7193, Error bound = 1.5033391572716326e-06


0.6945515091247271
Degree 1: Approximated cos(0.803) = 0.6974, Error bound = 0.0028484908752729465
Degree 2: Approximated cos(0.803) = 0.6942642533936647, Error bound = 0.0002872557310623636
Degree 3: Approximated cos(0.803) = 0.694663951734543, Error bound = 0.00011244260981591037
Degree 4: Approximated cos(0.803) = 0.6946, Error bound = 4.849087527292184e-05
```

第一題結果如上，題目要求用 lagrange 跑 cos 值，並給五個 x 跟 f(x)，將 5 個值分別帶入 lagrange 方程即有解答，error = 實際-估計值。

```
In [45]: %runfile C:/Users/jacks/OneDrive/Desktop/hw3/test2.py --wdir
when x - e^(-x) = 0 時，x ≈ 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 ≈ 768.96 英尺，速度 V ≈ 74.64 英尺/秒
車輛首次超過 55 mi/h 的時間為 t ≈ 3.15 秒
車輛的預測最大速度為 V_max ≈ 92.04 英尺/秒

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

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

，即可建程 hermite polynomial，再用建好的方程式跑三個問題即可