**一、**

只有做Bonus 2

**二、**

**(1)CRR Binomial Tree: Option\_Lookback\_CRR.py**

def calibratedPrice(stockPrice, St, layers):...

class Tree\_Node:...

def lookback\_CRR(StMax, St, T, r, q, sigma, layers, type):...

# main

St = 50

T = 0.25

r = 0.1

q = 0

sigma = 0.4

直接調整 **#main** 下面的變數，呼叫函式存入變數，再印出即可(所有參數以及函式都已預先輸入好，直接執行即可以看到精美的結果)。

輸出看起來會是這樣 :

============================================================

Lookback Option

[ Smax,t = 50 ]

------------------------------------------------------------

n = 100

(CRR Binomial Tree) Price of European Lookback Put : 7.2369

(CRR Binomial Tree) Price of American Lookback Put : 7.4396

------------------------------------------------------------

n = 300

...

**(2)Monte-Carlo: Option\_Lookback\_MonteCarlo.py**

def lookback\_MC(StMax, St, T, r, q, sigma, n, sims, reps):...

# main

StMax = 70

St = 50

T = 0.25

r = 0.1

q = 0

sigma = 0.4

...

直接調整 **#main** 下面的變數，呼叫函式存入變數，再印出即可(所有參數以及函式都已預先輸入好，直接執行即可以看到精美的結果)。

輸出看起來會是這樣 :

==================================================

Lookback Option : European Put

[ Smax,t = 50 ]

--------------------

平均 : 7.111868

標準誤 : 0.054249

九十五趴信賴區間 : [7.003369, 7.220366]

...

**(2)Bonus2: Option\_Lookback\_Cheuk&Vorst.py**

*def* lookback\_CRR\_CheukAndVorst(*St*, *T*, *r*, *q*, *sigma*, *layers*, *type*):...

# main

St = 50

T = 0.25

r = 0.1

q = 0

sigma = 0.4

直接調整 **#main** 下面的變數，呼叫函式存入變數，再印出即可(所有參數以及函式都已預先輸入好，直接執行即可以看到精美的結果)。

輸出看起來會是這樣 :

======================================================================

European Lookback Option

[ Smax,t = 50 ]

----------------------------------------------------------------------

n = 1000

(CRR Binomial Tree) Price of European Lookback Put : 7.61 (Cheuk & Vorst)

======================================================================

American Lookback Option

[ Smax,t = 50 ]

----------------------------------------------------------------------

n = 1000

(CRR Binomial Tree) Price of American Lookback Put : 7.8086 (Cheuk & Vorst)