MA-374 Lab Assignment 3

By-Arush Gupta
210123008

Question 1

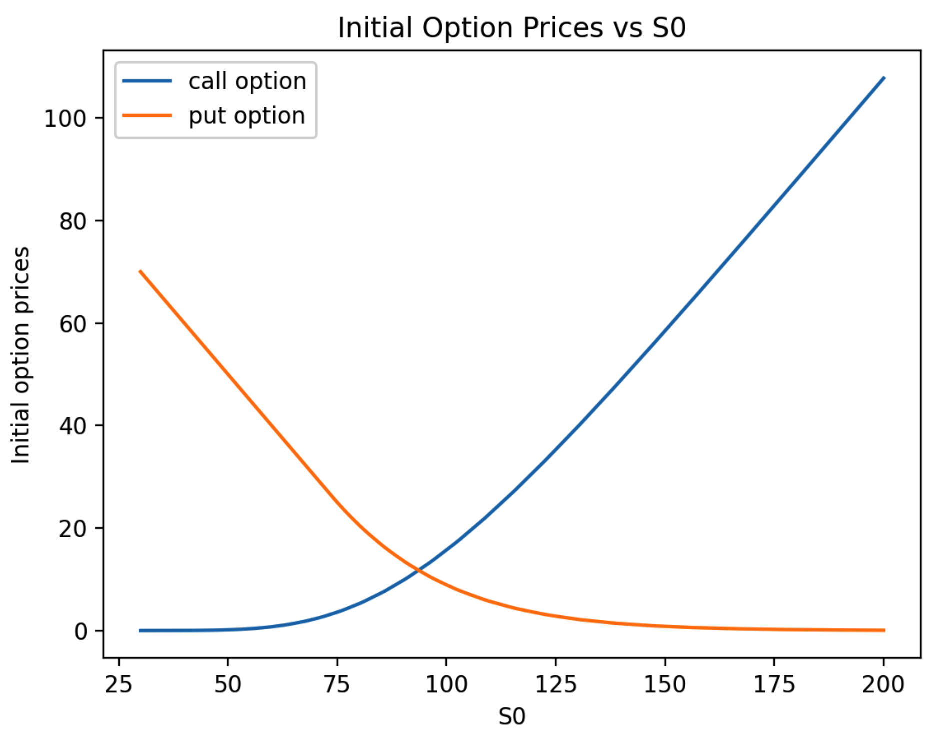
```
*****************************************
Initial Call option price is 15.736778626185815
Initial Put option price  is 8.92311328767774
*****************************************
```

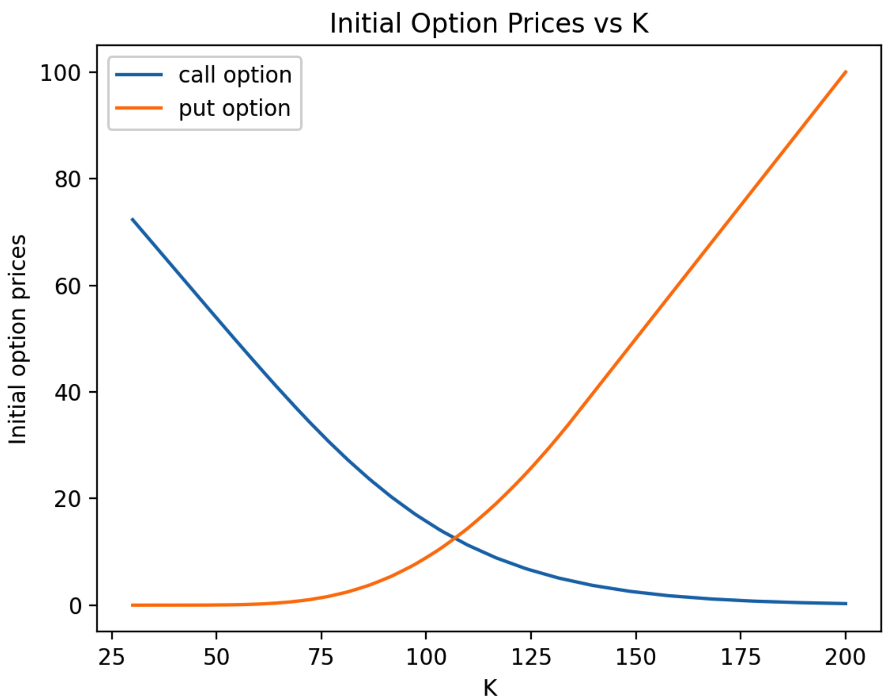
Given American Option with parameters:

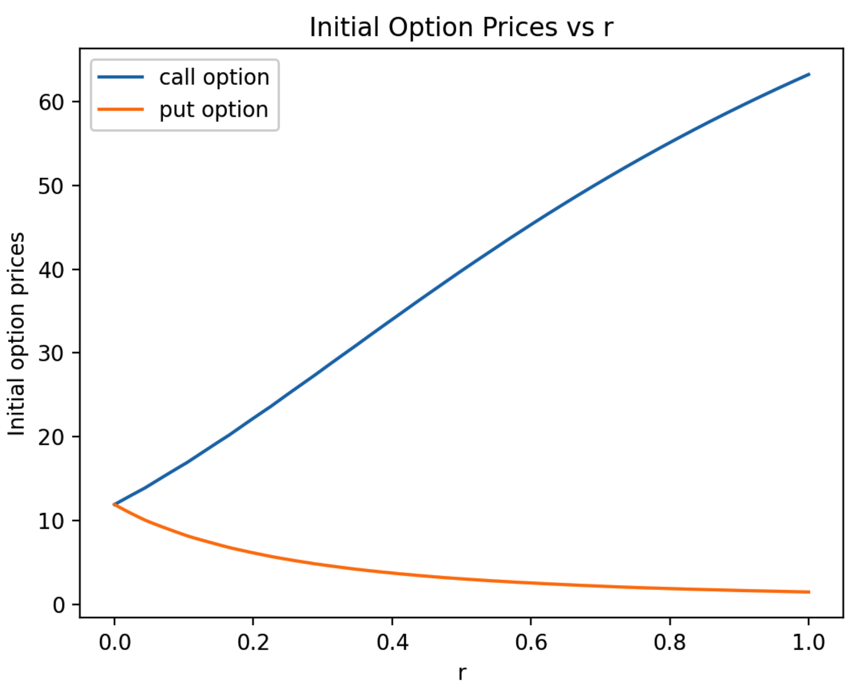
$S(0)= 100$, $K=100$, $T = 1$,$M = 100$, $r = 8\%$, $\sigma = 30\%$

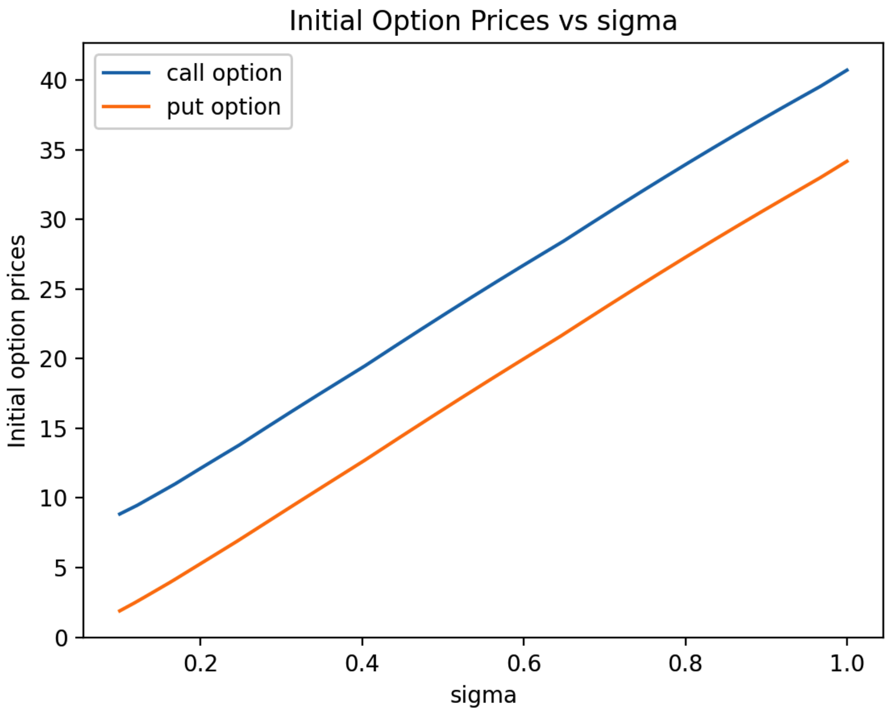
Up_factor = e^(sigma*root(t) + (R-½sigma*sigma)Δt)
Down_factor=e^(-sigma*root(t) + (R-½sigma*sigma)Δt)

2D Graphs

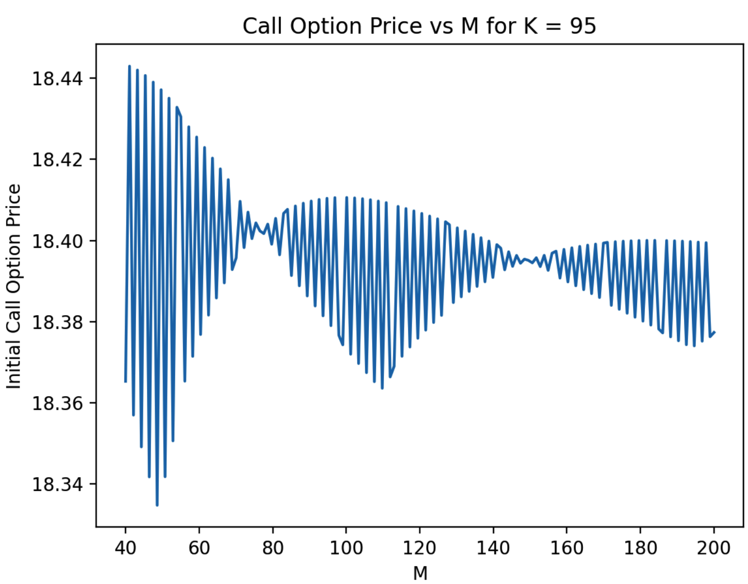



Initial Option Prices vs K
call option
put option
Initial option prices
K
0
20
40
60
80
100
25
50
75
100
125
150
175
200

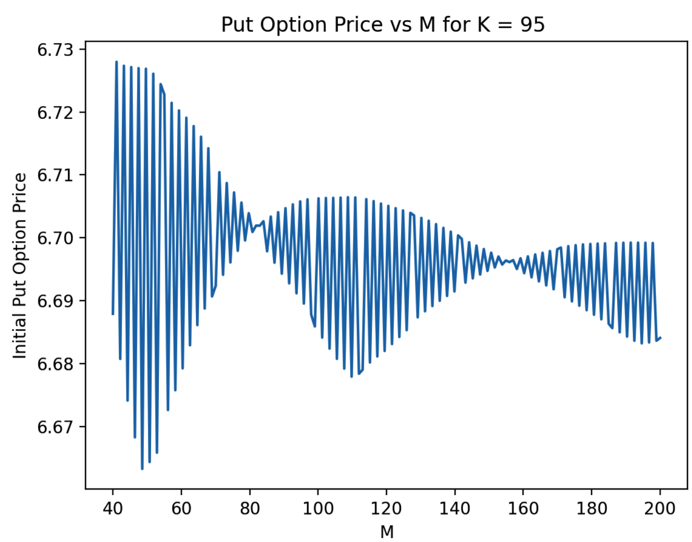

Initial Option Prices vs r
call option
put option
Initial option prices
r
0
10
20
30
40
50
60
0.0
0.2
0.4
0.6
0.8
1.0



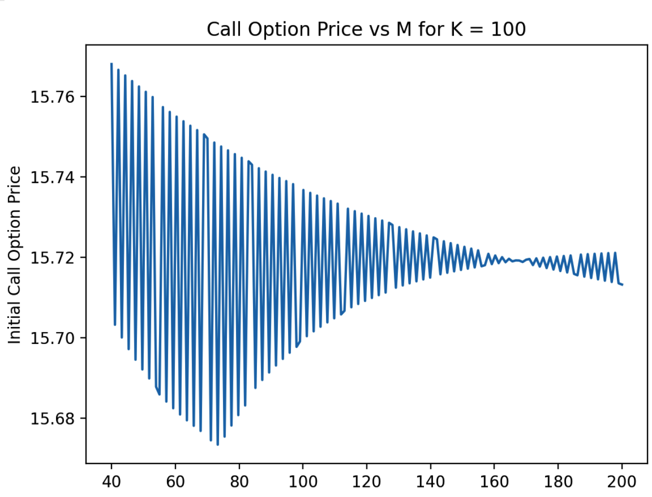
Initial Option Prices vs sigma
call option
put option
Initial option prices
sigma



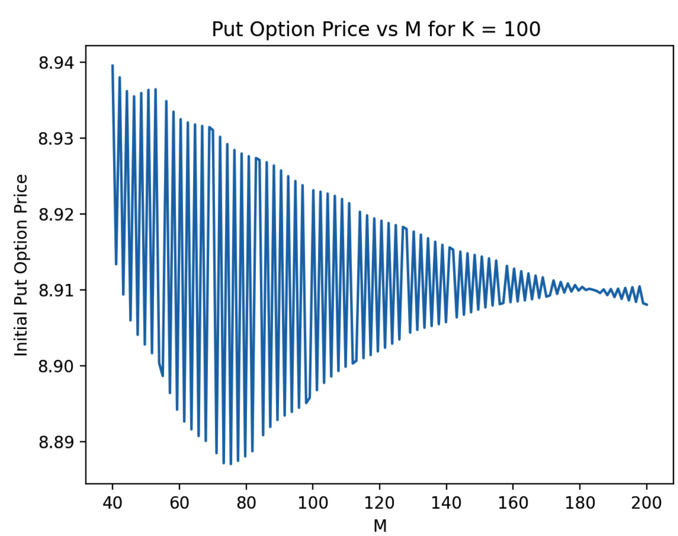
Call Option Price vs M for K = 95
Initial Call Option Price
M



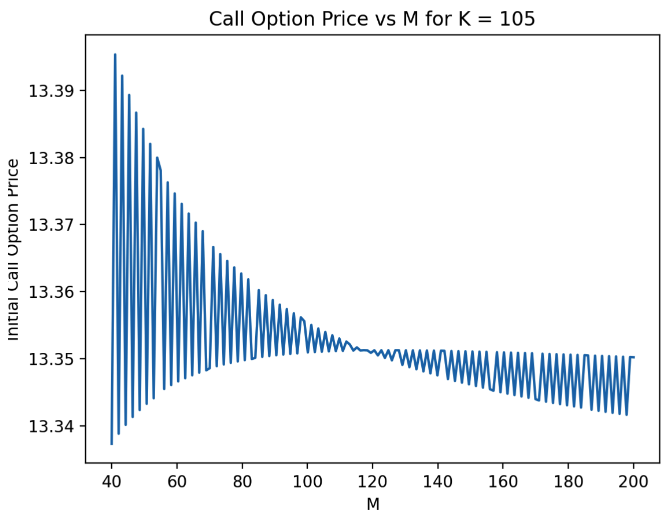
Put Option Price vs M for K = 95
Initial Put Option Price
M

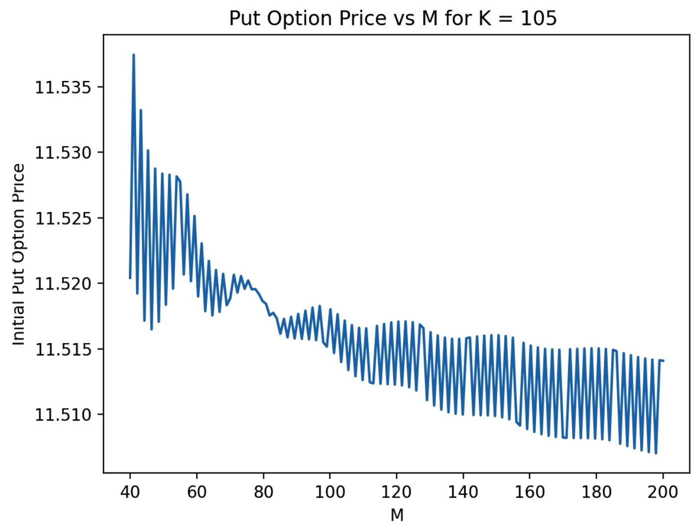


Call Option Price vs M for K = 100
Initial Call Option Price
M



Put Option Price vs M for K = 100
Initial Put Option Price
M





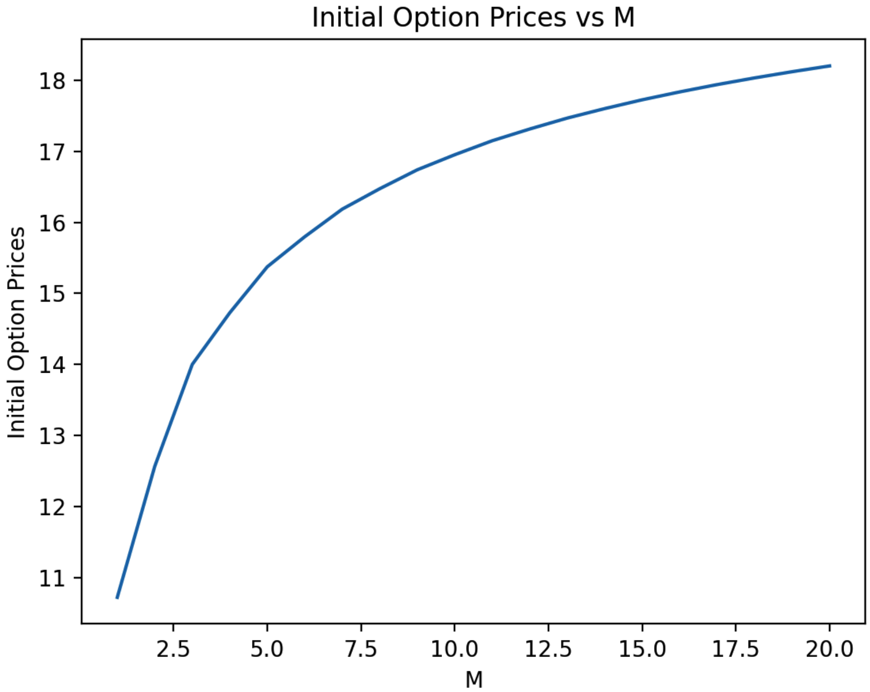
Question 2

Part A

```
**************************** Part A ******************************
For M=50 it is taking too much time so Putting Not Feasible

+-----+------------------------+-----------------------+
|   M | Initial Option Price   | Time Taken            |
+=====+========================+=======================+
|   5 | 15.372952215663782     | 3.886222839355469e-05 |
+-----+------------------------+-----------------------+
|  10 | 16.950340491777673     | 0.0008189678192138672 |
+-----+------------------------+-----------------------+
|  25 | 18.53378150009417      | 25.080140113830566    |
+-----+------------------------+-----------------------+
|  50 | Not Feasible           | Not Feasible          |
+-----+------------------------+-----------------------+
```

The initial option price will converge as the value of M is increased. The computation time increases exponentially as the value of M increases, therefore it becomes infeasible to calculate for larger values of M.

Part B



Part C

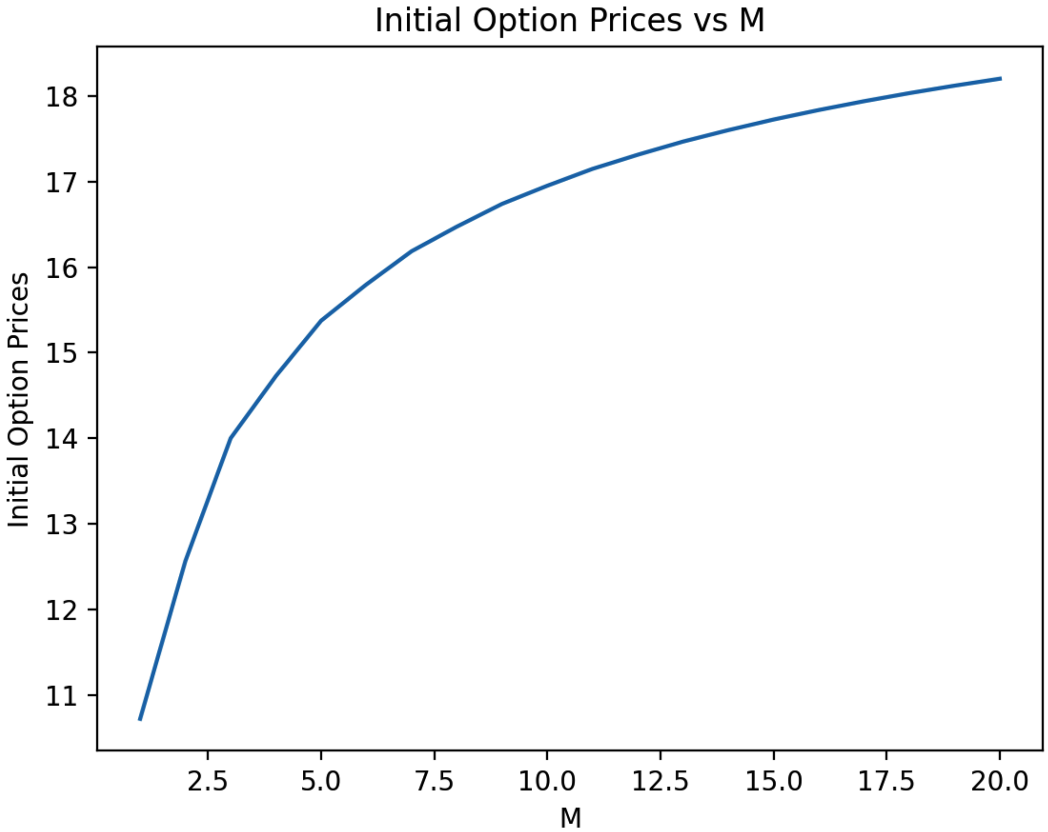
```
***************************** Part C ********************************
```

Time step	Values
t = 0	[15.382072999288122]
t = 1	[15.541029352487692, 15.719265966083048]
t = 2	[15.207899075118496, 16.375237171439704, 11.628823627215496, 20.315020328549444]
t = 3	[13.392862130341726, 17.512672765729953, 10.240943262976248, 23.035175288597717, 10.240943262976252, 13.39118453034756, 12.708324455749867, 28.57382863066069]
t = 4	[10.336645211210838, 16.87977920408257, 7.903986176166539, 27.68228703712279, 7.903986176166539, 12.907238157029042, 12.108163764549802, 34.70068887272636, 7.903986176166539, 12.907238157029049, 6.043836873231873, 21.16744936053216, 6.043836873231873, 19.77998149932738, 19.77998149932736, 38.2856636680285]
t = 5	[0.0, 21.002491662264447, 0.0, 34.29714522948986, 0.0, 16.05969832296735, 14.189941164644068, 42.06197481701972, 0.0, 16.05969832296735, 0.0, 26.225545739139193, 0.0, 24.601948051238253, 24.601948051238267, 45.914488453717624, 0.0, 16.05969832296735, 0.0, 26.225545739139207, 0.0, 12.280157724719814, 10.850435176426544, 32.162975578905915, 0.0, 12.280157724719814, 9.440589282577335, 30.75312968505669, 9.440589282577307, 30.753129685056678, 30.753129685056678, 47.04990888934698]

Question 3

Using Markov based optimization, we can now compute the option prices for larger values of M.

```
*********  Executing for M = 5  *********

No arbitrage exists for M = 5
Initial Price of Lookback Option        = 15.372952215663778
Execution Time                          = 0.00010275840759277344 sec



*********  Executing for M = 10  *********

No arbitrage exists for M = 10
Initial Price of Lookback Option        = 16.95034049177767
Execution Time                          = 0.0007631778717041016 sec



*********  Executing for M = 25  *********

No arbitrage exists for M = 25
Initial Price of Lookback Option        = 18.533781500094165
Execution Time                          = 0.05483078956604004 sec



*********  Executing for M = 50  *********

No arbitrage exists for M = 50
Initial Price of Lookback Option        = 19.390465235522452
Execution Time                          = 3.6521129608154297 sec
```


Initial Option Prices vs M
Initial Option Prices
M
18
17
16
15
14
13
12
11
2.5
5.0
7.5
10.0
12.5
15.0
17.5
20.0

```
(base) arushgupta@depressed-guy Fe2lab % python  /Users/arushgupta/Desktop/Fe2lab/08g.arushMA374lab03/q3.py
At t = 0
Intermediate state = (100, 100)         Price = 15.372952215663778

At t = 1
Intermediate state = (115.16135876866093, 115.16135876866093)           Price = 15.532131468492956
Intermediate state = (88.05891748599798, 100)           Price = 15.709699760878111

At t = 2
Intermediate state = (132.6213855344424, 132.6213855344424)             Price = 15.199750099616727
Intermediate state = (101.40984589384922, 115.16135876866093)           Price = 16.365773501799975
Intermediate state = (101.40984589384924, 101.40984589384924)           Price = 11.62259245758552
Intermediate state = (77.543729488058, 100)             Price = 20.30531014128848

At t = 3
Intermediate state = (152.7285895992882, 152.7285895992882)             Price = 13.386169289151374
Intermediate state = (116.78495645656189, 132.6213855344424)            Price = 17.50446467389843
Intermediate state = (116.78495645656187, 116.78495645656187)           Price = 10.235825536366997
Intermediate state = (89.3004125183424, 115.16135876866093)             Price = 23.026215406441317
Intermediate state = (116.78495645656189, 116.78495645656189)           Price = 10.235825536367
Intermediate state = (89.30041251834241, 101.40984589384924)            Price = 13.384908157013323
Intermediate state = (89.3004125183424, 100)            Price = 12.702323203700722
Intermediate state = (68.28416876545448, 100)           Price = 28.566489442465258

At t = 4
Intermediate state = (175.88431901075205, 175.88431901075205)           Price = 10.33248062285694
Intermediate state = (134.4911426927657, 152.7285895992882)             Price = 16.872978416162187
Intermediate state = (134.4911426927657, 134.4911426927657)             Price = 7.900801695311674
Intermediate state = (102.8395684421425, 132.6213855344424)             Price = 27.676760285887045
Intermediate state = (134.49114269276566, 134.49114269276566)           Price = 7.900801695311674
Intermediate state = (102.8395684421425, 116.78495645656187)            Price = 12.902037888217311
Intermediate state = (102.8395684421425, 115.16135876866093)            Price = 12.103285439254641
Intermediate state = (78.63697657418294, 115.16135876866093)            Price = 34.69646280474455
Intermediate state = (102.8395684421425, 116.78495645656189)            Price = 12.902037888217318
Intermediate state = (102.83956844214251, 102.83956844214251)           Price = 6.041401838252844
Intermediate state = (78.63697657418295, 101.40984589384924)            Price = 21.163223292550345
Intermediate state = (102.8395684421425, 102.8395684421425)             Price = 6.041401838252844
Intermediate state = (78.63697657418294, 100)           Price = 19.775755431345573
Intermediate state = (78.63697657418295, 100)           Price = 19.77575543134555
Intermediate state = (60.130299829171165, 100)          Price = 38.2824321763573

At t = 5
Intermediate state = (202.5507716337883, 202.5507716337883)             Price = 0.0
Intermediate state = (154.8818273484876, 175.88431901075205)            Price = 21.002491662264447
Intermediate state = (154.8818273484876, 154.8818273484876)             Price = 0.0
Intermediate state = (118.43144436979834, 152.7285895992882)            Price = 34.29714522948986
Intermediate state = (118.43144436979834, 134.4911426927657)            Price = 16.05969832296735
Intermediate state = (118.43144436979833, 132.6213855344424)            Price = 14.189941164644068
Intermediate state = (90.55941071742268, 132.6213855344424)             Price = 42.06197481701972
Intermediate state = (154.88182734848758, 154.88182734848758)           Price = 0.0
Intermediate state = (118.43144436979831, 134.49114269276566)           Price = 16.05969832296735
Intermediate state = (118.43144436979833, 118.43144436979833)           Price = 0.0
Intermediate state = (90.55941071742268, 116.78495645656187)            Price = 26.225545739139193
Intermediate state = (90.55941071742268, 115.16135876866093)            Price = 24.601948051238253
Intermediate state = (90.55941071742267, 115.16135876866093)            Price = 24.601948051238267
Intermediate state = (69.24687031494331, 115.16135876866093)            Price = 45.914488453717624
Intermediate state = (90.55941071742268, 116.78495645656189)            Price = 26.225545739139207
Intermediate state = (118.43144436979834, 118.43144436979834)           Price = 0.0
Intermediate state = (90.5594107174227, 102.83956844214251)             Price = 12.280157724719814
```

Therefore, the algorithm is efficient than the previous one.

The unoptimized algorithm exhibits exponential space complexity, whereas the Markov-based algorithm avoids this issue by leveraging memoization principles from dynamic programming. The unoptimized approach becomes impractical for small values of M, such as 50, as its time complexity can significantly escalate. In contrast, the Markov-based algorithm efficiently manages this challenge.

Question 4

```
(base) arushgupta@depressed-guy Fe2lab % python '/Users/arushgupta/Desktop/Fe2lab/08g.arushMA374lab03/q4.py'
```

M	call option price	Computational time without Markov	Computational time with Markov
1	18.4088	3.1948089599609375e-05	8.10623e-06
5	16.2001	3.0279159545898438e-05	2.26498e-05
10	15.7497	0.0005240440368652344	6.60419e-05
20	15.7788	0.4526650905609131	0.000218153
25	15.7469	15.749392986297607	0.000551939
30	15.7751	540.9872000217438	0.000702143
50	15.7612	Not Feasible	0.0034833

Here, the time complexity for the naive approach is O(2^n) whereas after using Markov based optimization (DP) the time complexity reduces to O(n^2) thereby allowing us to calculate for larger values of M

The unoptimised algorithm has exponential time and space complexity. The efficient algorithm has quadratic space and time complexity (in M),