Problem 2.2

1 Problem Description

(1) Hypothetically, the number of coronavirus patients during the first 45 days of the outbreak is given by $y(t_n+1) = y(t_n)(1+x(t_n))$, where $t_n = 1, 2, 3, ...44, 45, ...89, 90$, and $y(t_1) = 10$ and $x(t_1)$ is a number chose randomly from a pool of 90 numbers that are normally distributed with $\mu = 0.18$ and $\sigma = 0.08$. Compute and graph the number of infected patients in the first 45 days.
(2) Do the same for the second 45 days, with $\mu = -0.24$ and $\sigma = 0.04$.
(3) Interpolate the data points $t = 9, 18, 27, 36, 45$ by a $y(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4$
(4) Fit the results of the 2nd 45 days in $y(t) = \alpha e^{\beta t}$.
(5) Plot all results in one figure.

2 Algorithm Description and Pseudo-Code

2.1 Random Number Generation

To generate uniformely distributed random numbers, I used the following pseudo random number generator:

$$x_{n+1} = (x_n + x_{n-1}) \; mod \; 1.0 = \begin{cases} x_n + x_{n-1} - 1.0 & if \; (x_n + x_{n-1}) > 1.0 \\ x_n + x_{n-1} & Otherwise \end{cases}$$

Source: Professor Deng's lecture notes (Lecture 2)

2.2 Generating Normal Distribution

I used the Box-Muller transform on each pair of numbers x_1 and x_2 to generate a normal distribution:

$x_1 = \sqrt{-2 \ln x_1} \cos 2\pi x_2$

$x_2 = \sqrt{-2 \ln x_1} \sin 2\pi x_2$

Then, to achieve the required μ and σ, I transformed x_1 and x_2 by the following:

$z_1 = \mu + x_1 * \sigma$
$z_2 = \mu + x_2 * \sigma$

2.3 Interpolation

I interpolated using 5 evenly-selected data points (t = 9, 18, 27, 36, and 45) by a polynomial $y(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4$.

To do this I formed a system of equations using the known values at the selected data points. Then I was able to solve for the coefficients in the polynomial using linear algebra.

Pseudo-code:

```
A = [
    [9^4, 9^3, 9^2, 9^1, 9^0],
    [18^4, 18^3, 18^2, 18^1, 18^0],
    [27^4, 27^3, 27^2, 27^1, 27^0],
    [36^4, 36^3, 36^2, 36^1, 36^0],
    [45^4, 45^3, 45^2, 45^1, 45^0]
]

B = [y(9), y(18), y(27), y(36), y(45)]

coefficients = dot product of inverseOf(A) and B
```

2.4 Curve-Fitting

I fit the data for the second 45 days to the function $y(t) = \alpha e^{\beta t}$ using the non-linear least squares method.

3 Results

Note: Left column is the day number, right column is number of infected patients on that day. The graphs are attached as PNG files since they were hard to see when I embedded them directly in this PDF.

2.2.1

Day 1	10
Day 2	11
Day 3	18
Day 4	17
Day 5	24
Day 6	32
Day 7	30
Day 8	35
Day 9	41
Day 10	44
Day 11	52
Day 12	36
Day 13	46
Day 14	60
Day 15	73
Day 16	106
Day 17	80
Day 18	105
Day 19	139
Day 20	180
Day 21	227
Day 22	221
Day 23	320

Day 24	558
Day 25	902
Day 26	806
Day 27	1171
Day 28	1015
Day 29	354
Day 30	535
Day 31	454
Day 32	525
Day 33	796
Day 34	676
Day 35	745
Day 36	705
Day 37	786
Day 38	1027
Day 39	1412
Day 40	1393
Day 41	2134
Day 42	3517
Day 43	3370
Day 44	4819
Day 45	4342

2.2.2

Day 46	4926
Day 47	2008
Day 48	1547
Day 49	1200
Day 50	597
Day 51	567
Day 52	336
Day 53	272
Day 54	268
Day 55	129
Day 56	125
Day 57	71
Day 58	47
Day 59	31
Day 60	32
Day 61	23
Day 62	22
Day 63	12
Day 64	3
Day 65	3
Day 66	2
Day 67	2
Day 68	2

Day 69	3
Day 70	2
Day 71	2
Day 72	2
Day 73	2
Day 74	1
Day 75	1
Day 76	1
Day 77	1
Day 78	1
Day 79	1
Day 80	1
Day 81	1
Day 82	1
Day 83	1
Day 84	1
Day 85	1
Day 86	1
Day 87	1
Day 88	2
Day 89	2
Day 90	2

2.2.3

Interpolated polynomial:

$y(t) = .0467408424x^4 - 4.63786008x^3 + 160.615226x^2 - 2207.50000x + 997600000$

Day 1	7924.524107097373
Day 2	6167.105878168883
Day 3	4677.600823045194
Day 4	3430.986231773558
Day 5	2403.361174617652
Day 6	1571.946502057588
Day 7	915.0848447899061
Day 8	412.2406137275757
Day 9	44.0
Day 10	-207.9290250469894
Day 11	-360.7167098511309
Day 12	-430.4115226337344
Day 13	-431.94015139967087
Day 14	-379.10750393739727
Day 15	-284.5967078189133
Day 16	-159.9691103998266
Day 17	-15.66427881927666
Day 18	139.00000000000728
Day 19	295.82771935173514
Day 20	447.744652746027
Day 21	588.7983539094639
Day 22	714.1581567850517
Day 23	820.1151755321916

Day 24	904.0823045267462
Day 25	964.5942183610096
Day 26	1001.3073718437226
Day 27	1014.9999999999927
Day 28	1007.572118071439
Day 29	982.0455215160109
Day 30	942.5637860082425
Day 31	894.3922674388959
Day 32	843.9181019153693
Day 33	798.6502057613179
Day 34	767.2192755169599
Day 35	759.3777879388435
Day 36	786.0000000000073
Day 37	859.0819488898924
Day 38	991.7414520144448
Day 39	1198.2181069959406
Day 40	1493.873291673015
Day 41	1895.1901641009827
Day 42	2419.7736625514153
Day 43	3086.3505055123896
Day 44	3914.769191688254
Day 45	4926.000000000007

2.2.4

Fitted equation:

$y(t) = 117.94464445124133e^{-2.506498978471238t}$

Day 46	520.7309189138546
Day 47	209.84895258648592
Day 48	207.1095152135583
Day 49	415.6308052571721
Day 50	133.87220041464712
Day 51	328.0233243472632
Day 52	190.40940054181166
Day 53	123.4690494766615
Day 54	434.9328336517064
Day 55	129.16323563529127
Day 56	349.23297656996283
Day 57	284.6189015903966
Day 58	283.5826925450199
Day 59	113.04190972614892
Day 60	257.35119115574804
Day 61	141.09325131963183
Day 62	368.78124570604876
Day 63	831.2724607087118
Day 64	122.98222727498262
Day 65	347.06249386652297
Day 66	270.4515548094896
Day 67	281.45254294134327
Day 68	111.44479123633825

Day 69	318.10656044112875
Day 70	148.05134806502443
Day 71	176.0009256908402
Day 72	346.6747012197403
Day 73	526.5791339693726
Day 74	237.36252917506624
Day 75	217.85485019992703
Day 76	484.13672331211245
Day 77	248.5348376997649
Day 78	553.3129580343355
Day 79	125.18164922362864
Day 80	377.29760225453293
Day 81	245.5344093477934
Day 82	325.95921651410674
Day 83	240.97372937399265
Day 84	176.08005122647327
Day 85	151.39518950666442
Day 86	271.55711194916415
Day 87	106.07562845439033
Day 88	380.12720848219436
Day 89	188.90569779337577
Day 90	203.00865218178518

4 Brief comments on performance & other questions

One phenomenon that can be observed is that the curve-fitted data appears to be less accurate than the interpolated data, though both are fairly inaccurate when compared to the original data. Additionally, different methods of interpolation or curve-fitting may yield more accurate results; for example, using spline interpolation may result in a more accurate interpolation of the data. All of the calculations being done here are iterative and are dependant on the size of their input, therefore we can estimate the running time of them as $O(N)$.