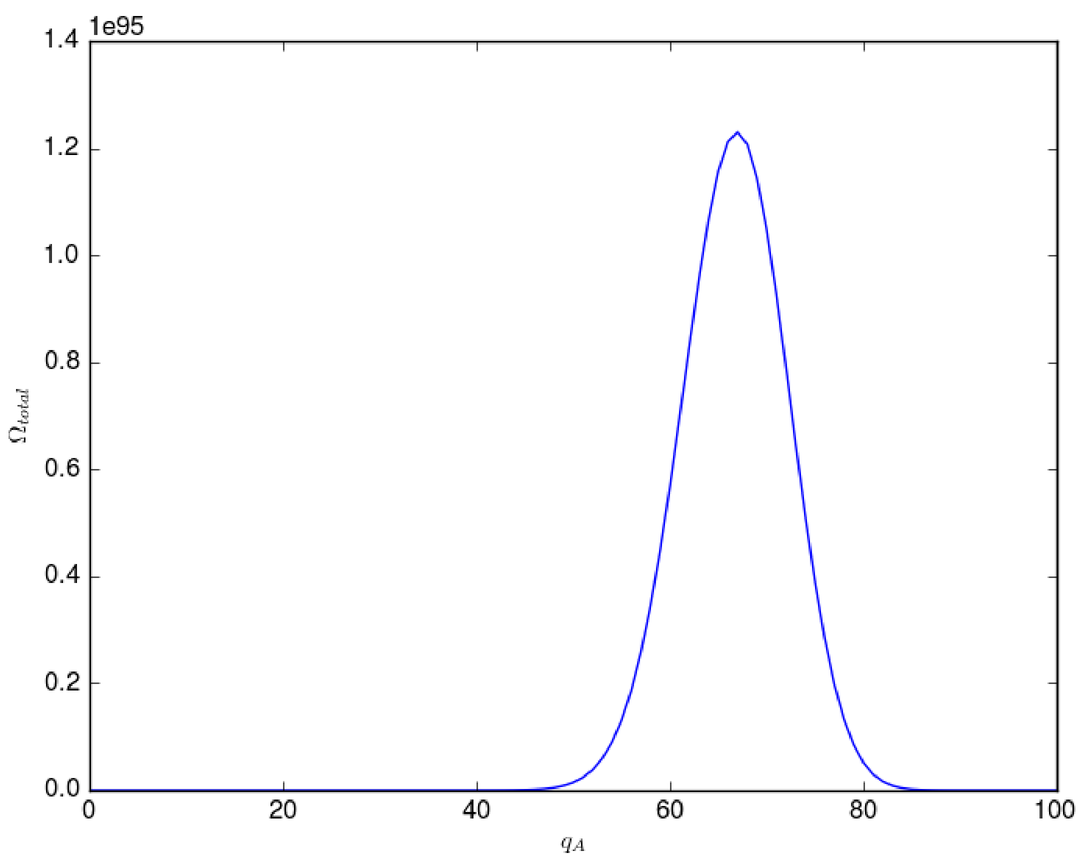
Matt Meyers PHYS 309-010 10/7/15

A)

q_a	omega_a	q_b	omega_b	omega_tot
0	1.00e+00	100	4.53e+58	4.53e+58
1	2.00e+02	99	2.28e+58	4.55e+60
2	2.01e+04	98	1.14e+58	2.29e+62
3	1.35e+06	97	5.66e+57	7.66e+63
4	6.87e+07	96	2.80e+57	1.92e+65
5	2.80e+09	95	1.38e+57	3.86e+66
6	9.57e+10	94	6.75e+56	6.46e+67
7	2.82e+12	93	3.29e+56	9.27e+68
8	7.29e+13	92	1.59e+56	1.16e+70
9	1.68e+15	91	7.67e+55	1.29e+71
10	3.52e+16	90	3.67e+55	1.29e+72
11	6.72e+17	89	1.75e+55	1.18e+73
12	1.18e+19	88	8.28e+54	9.79e+73
13	1.93e+20	87	3.90e+54	7.51e+74
14	2.93e+21	86	1.82e+54	5.35e+75
15	4.18e+22	85	8.48e+53	3.55e+76
16	5.62e+23	84	3.92e+53	2.20e+77
17	7.14e+24	83	1.80e+53	1.28e+78
18	8.61e+25	82	8.20e+52	7.06e+78
19	9.88e+26	81	3.71e+52	3.67e+79
20	1.08e+28	80	1.67e+52	1.81e+80
21	1.13e+29	79	7.47e+51	8.47e+80
22	1.14e+30	78	3.31e+51	3.77e+81
23	1.10e+31	77	1.46e+51	1.61e+82
24	1.02e+32	76	6.39e+50	6.53e+82
25	9.15e+32	75	2.78e+50	2.54e+83
26	7.92e+33	74	1.20e+50	9.47e+83
27	6.63e+34	73	5.12e+49	3.39e+84
28	5.37e+35	72	2.17e+49	1.17e+85
29	4.22e+36	71	9.14e+48	3.86e+85
30	3.23e+37	70	3.82e+48	1.23e+86
31	2.39e+38	69	1.58e+48	3.78e+86
32	1.73e+39	68	6.50e+47	1.12e+87
33	1.21e+40	67	2.65e+47	3.21e+87
34	8.32e+40	66	1.07e+47	8.88e+87
35	5.56e+41	65	4.27e+46	2.38e+88
36	3.63e+42	64	1.69e+46	6.15e+88
37	2.32e+43	63	6.65e+45	1.54e+89
38	1.44e+44	62	2.58e+45	3.73e+89
39	8.82e+44	61	9.95e+44	8.78e+89
40	5.27e+45	60	3.79e+44	2.00e+90
41	3.08e+46	59	1.43e+44	4.42e+90

42	1.77e+47	58	5.35e+43	9.46e+90
43	9.96e+47	57	1.98e+43	1.97e+91
44	5.50e+48	56	7.22e+42	3.97e+91
45	2.98e+49	55	2.61e+42	7.78e+91
46	1.59e+50	54	9.31e+41	1.48e+92
47	8.31e+50	53	3.29e+41	2.73e+92
48	4.28e+51	52	1.15e+41	4.90e+92
49	2.17e+52	51	3.95e+40	8.55e+92
50	1.08e+53	50	1.34e+40	1.45e+93
51	5.29e+53	49	4.50e+39	2.38e+93
52	2.55e+54	48	1.49e+39	3.80e+93
53	1.21e+55	47	4.87e+38	5.91e+93
54	5.68e+55	46	1.57e+38	8.91e+93
55	2.62e+56	45	4.97e+37	1.30e+94
56	1.20e+57	44	1.55e+37	1.86e+94
57	5.37e+57	43	4.78e+36	2.57e+94
58	2.38e+58	42	1.45e+36	3.44e+94
59	1.04e+59	41	4.31e+35	4.48e+94
60	4.49e+59	40	1.26e+35	5.67e+94
61	1.91e+60	39	3.63e+34	6.95e+94
62	8.06e+60	38	1.03e+34	8.27e+94
63	3.35e+61	37	2.85e+33	9.54e+94
64	1.38e+62	36	7.75e+32	1.07e+95
65	5.59e+62	35	2.07e+32	1.16e+95
66	2.25e+63	34	5.40e+31	1.21e+95
67	8.91e+63	33	1.38e+31	1.23e+95
68	3.50e+64	32	3.45e+30	1.21e+95
69	1.36e+65	31	8.43e+29	1.15e+95
70	5.22e+65	30	2.01e+29	1.05e+95
71	1.99e+66	29	4.67e+28	9.28e+94
72	7.48e+66	28	1.06e+28	7.92e+94
73	2.79e+67	27	2.33e+27	6.50e+94
74	1.03e+68	26	5.00e+26	5.14e+94
75	3.75e+68	25	1.04e+26	3.91e+94
76	1.36e+69	24	2.10e+25	2.85e+94
77	4.87e+69	23	4.09e+24	1.99e+94
78	1.73e+70	22	7.72e+23	1.33e+94
79	6.09e+70	21	1.40e+23	8.54e+93
80	2.12e+71	20	2.46e+22	5.21e+93
81	7.34e+71	19	4.13e+21	3.03e+93
82	2.51e+72	18	6.64e+20	1.67e+93
83	8.54e+72	17	1.02e+20	8.73e+92
84	2.88e+73	16	1.50e+19	4.31e+92
85	9.62e+73	15	2.08e+18	2.00e+92

86	3.19e+74	14	2.74e+17	8.74e+91
87	1.05e+75	13	3.40e+16	3.56e+91
88	3.42e+75	12	3.94e+15	1.35e+91
89	1.11e+76	11	4.26e+14	4.71e+90
90	3.55e+76	10	4.26e+13	1.51e+90
91	1.13e+77	9	3.91e+12	4.43e+89
92	3.58e+77	8	3.26e+11	1.17e+89
93	1.12e+78	7	2.44e+10	2.74e+88
94	3.50e+78	6	1.61e+09	5.64e+87
95	1.08e+79	5	9.20e+07	9.97e+86
96	3.33e+79	4	4.42e+06	1.47e+86
97	1.02e+80	3	1.72e+05	1.75e+85
98	3.08e+80	2	5.05e+03	1.56e+84
99	9.27e+80	1	1.00e+02	9.27e+82
100	2.77e+81	0	1.00e+00	2.77e+81



B) The most probable macrostate corresponds to q_A = 67 and q_B = 33. The probability of this is 0.0732. The least probable macrostate corresponds to q_A = 0 and q_B = 100. The probability of this is 2.69×10^{-38}.

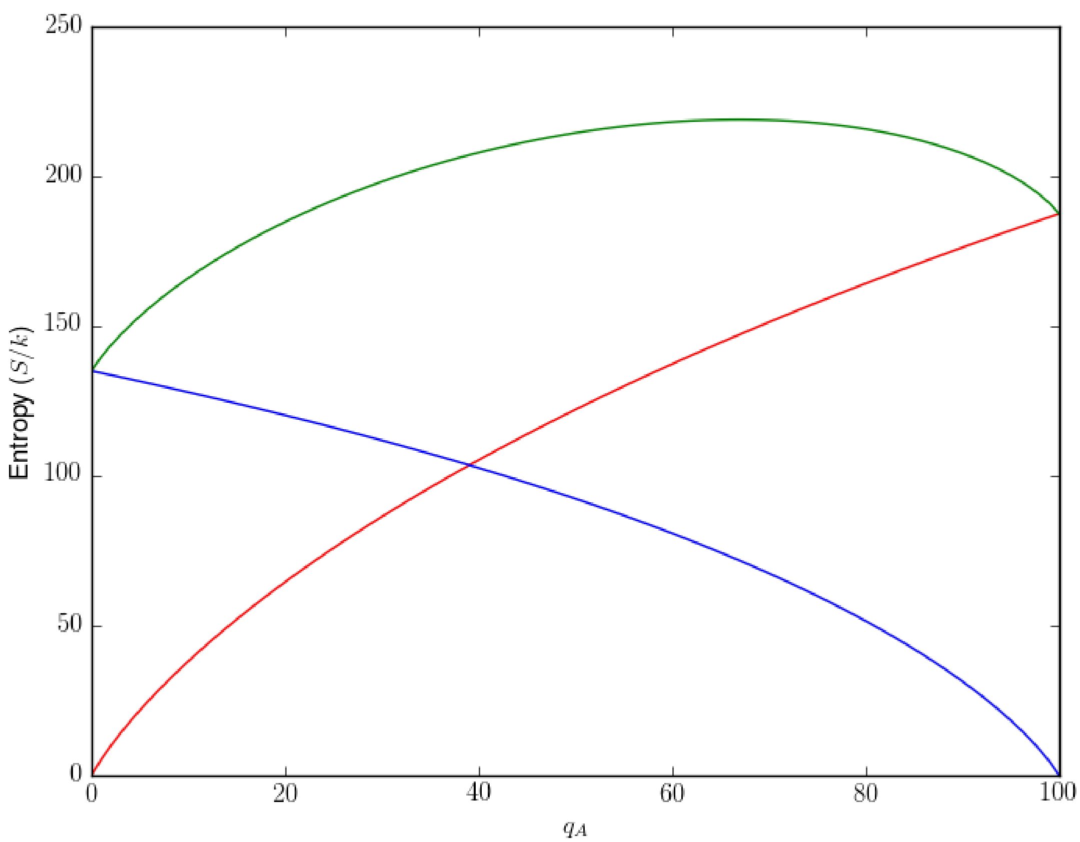
C)

q_a	omega_a	S_a/k	q_b	omega_b	S_b/k	omega_tot	S_tot/k
0	1.00e+00	0.00	100	4.53e+58	135.06	4.53e+58	135.06
1	2.00e+02	5.30	99	2.28e+58	134.37	4.55e+60	139.67
2	2.01e+04	9.91	98	1.14e+58	133.68	2.29e+62	143.59
3	1.35e+06	14.12	97	5.66e+57	132.98	7.66e+63	147.10
4	6.87e+07	18.05	96	2.80e+57	132.28	1.92e+65	150.32
5	2.80e+09	21.75	95	1.38e+57	131.57	3.86e+66	153.32
6	9.57e+10	25.28	94	6.75e+56	130.85	6.46e+67	156.14
7	2.82e+12	28.67	93	3.29e+56	130.14	9.27e+68	158.80
8	7.29e+13	31.92	92	1.59e+56	129.41	1.16e+70	161.33
9	1.68e+15	35.06	91	7.67e+55	128.68	1.29e+71	163.74
10	3.52e+16	38.10	90	3.67e+55	127.94	1.29e+72	166.04
11	6.72e+17	41.05	89	1.75e+55	127.20	1.18e+73	168.25
12	1.18e+19	43.92	88	8.28e+54	126.45	9.79e+73	170.37
13	1.93e+20	46.71	87	3.90e+54	125.70	7.51e+74	172.41
14	2.93e+21	49.43	86	1.82e+54	124.94	5.35e+75	174.37
15	4.18e+22	52.09	85	8.48e+53	124.17	3.55e+76	176.26
16	5.62e+23	54.69	84	3.92e+53	123.40	2.20e+77	178.09
17	7.14e+24	57.23	83	1.80e+53	122.62	1.28e+78	179.85
18	8.61e+25	59.72	82	8.20e+52	121.84	7.06e+78	181.56
19	9.88e+26	62.16	81	3.71e+52	121.05	3.67e+79	183.20
20	1.08e+28	64.55	80	1.67e+52	120.25	1.81e+80	184.80
21	1.13e+29	66.90	79	7.47e+51	119.44	8.47e+80	186.34
22	1.14e+30	69.21	78	3.31e+51	118.63	3.77e+81	187.84
23	1.10e+31	71.47	77	1.46e+51	117.81	1.61e+82	189.29
24	1.02e+32	73.70	76	6.39e+50	116.98	6.53e+82	190.69
25	9.15e+32	75.90	75	2.78e+50	116.15	2.54e+83	192.05
26	7.92e+33	78.05	74	1.20e+50	115.31	9.47e+83	193.36
27	6.63e+34	80.18	73	5.12e+49	114.46	3.39e+84	194.64
28	5.37e+35	82.27	72	2.17e+49	113.60	1.17e+85	195.87
29	4.22e+36	84.33	71	9.14e+48	112.74	3.86e+85	197.07
30	3.23e+37	86.37	70	3.82e+48	111.86	1.23e+86	198.23
31	2.39e+38	88.37	69	1.58e+48	110.98	3.78e+86	199.35
32	1.73e+39	90.35	68	6.50e+47	110.09	1.12e+87	200.44
33	1.21e+40	92.30	67	2.65e+47	109.19	3.21e+87	201.49
34	8.32e+40	94.22	66	1.07e+47	108.29	8.88e+87	202.51
35	5.56e+41	96.12	65	4.27e+46	107.37	2.38e+88	203.49
36	3.63e+42	98.00	64	1.69e+46	106.45	6.15e+88	204.44
37	2.32e+43	99.85	63	6.65e+45	105.51	1.54e+89	205.36
38	1.44e+44	101.68	62	2.58e+45	104.57	3.73e+89	206.25
39	8.82e+44	103.49	61	9.95e+44	103.61	8.78e+89	207.10
40	5.27e+45	105.28	60	3.79e+44	102.65	2.00e+90	207.93
41	3.08e+46	107.05	59	1.43e+44	101.67	4.42e+90	208.72

42	1.77e+47	108.79	58	5.35e+43	100.69	9.46e+90	209.48
43	9.96e+47	110.52	57	1.98e+43	99.69	1.97e+91	210.21
44	5.50e+48	112.23	56	7.22e+42	98.69	3.97e+91	210.91
45	2.98e+49	113.92	55	2.61e+42	97.67	7.78e+91	211.59
46	1.59e+50	115.59	54	9.31e+41	96.64	1.48e+92	212.23
47	8.31e+50	117.25	53	3.29e+41	95.60	2.73e+92	212.84
48	4.28e+51	118.89	52	1.15e+41	94.54	4.90e+92	213.43
49	2.17e+52	120.51	51	3.95e+40	93.48	8.55e+92	213.98
50	1.08e+53	122.11	50	1.34e+40	92.40	1.45e+93	214.51
51	5.29e+53	123.70	49	4.50e+39	91.31	2.38e+93	215.01
52	2.55e+54	125.28	48	1.49e+39	90.20	3.80e+93	215.48
53	1.21e+55	126.84	47	4.87e+38	89.08	5.91e+93	215.92
54	5.68e+55	128.38	46	1.57e+38	87.95	8.91e+93	216.33
55	2.62e+56	129.91	45	4.97e+37	86.80	1.30e+94	216.71
56	1.20e+57	131.43	44	1.55e+37	85.64	1.86e+94	217.06
57	5.37e+57	132.93	43	4.78e+36	84.46	2.57e+94	217.39
58	2.38e+58	134.42	42	1.45e+36	83.26	3.44e+94	217.68
59	1.04e+59	135.89	41	4.31e+35	82.05	4.48e+94	217.94
60	4.49e+59	137.35	40	1.26e+35	80.82	5.67e+94	218.18
61	1.91e+60	138.80	39	3.63e+34	79.58	6.95e+94	218.38
62	8.06e+60	140.24	38	1.03e+34	78.31	8.27e+94	218.56
63	3.35e+61	141.67	37	2.85e+33	77.03	9.54e+94	218.70
64	1.38e+62	143.08	36	7.75e+32	75.73	1.07e+95	218.81
65	5.59e+62	144.48	35	2.07e+32	74.41	1.16e+95	218.89
66	2.25e+63	145.87	34	5.40e+31	73.07	1.21e+95	218.94
67	8.91e+63	147.25	33	1.38e+31	71.70	1.23e+95	218.95
68	3.50e+64	148.62	32	3.45e+30	70.32	1.21e+95	218.93
69	1.36e+65	149.98	31	8.43e+29	68.91	1.15e+95	218.88
70	5.22e+65	151.32	30	2.01e+29	67.47	1.05e+95	218.79
71	1.99e+66	152.66	29	4.67e+28	66.01	9.28e+94	218.67
72	7.48e+66	153.98	28	1.06e+28	64.53	7.92e+94	218.51
73	2.79e+67	155.30	27	2.33e+27	63.02	6.50e+94	218.32
74	1.03e+68	156.60	26	5.00e+26	61.48	5.14e+94	218.08
75	3.75e+68	157.90	25	1.04e+26	59.91	3.91e+94	217.81
76	1.36e+69	159.18	24	2.10e+25	58.31	2.85e+94	217.49
77	4.87e+69	160.46	23	4.09e+24	56.67	1.99e+94	217.13
78	1.73e+70	161.73	22	7.72e+23	55.00	1.33e+94	216.73
79	6.09e+70	162.99	21	1.40e+23	53.30	8.54e+93	216.28
80	2.12e+71	164.24	20	2.46e+22	51.56	5.21e+93	215.79
81	7.34e+71	165.48	19	4.13e+21	49.77	3.03e+93	215.25
82	2.51e+72	166.71	18	6.64e+20	47.95	1.67e+93	214.65
83	8.54e+72	167.93	17	1.02e+20	46.07	8.73e+92	214.00
84	2.88e+73	169.15	16	1.50e+19	44.15	4.31e+92	213.30
85	9.62e+73	170.35	15	2.08e+18	42.18	2.00e+92	212.53

86	3.19e+74	171.55	14	2.74e+17	40.15	8.74e+91	211.70
87	1.05e+75	172.74	13	3.40e+16	38.06	3.56e+91	210.80
88	3.42e+75	173.92	12	3.94e+15	35.91	1.35e+91	209.83
89	1.11e+76	175.10	11	4.26e+14	33.69	4.71e+90	208.78
90	3.55e+76	176.26	10	4.26e+13	31.38	1.51e+90	207.65
91	1.13e+77	177.42	9	3.91e+12	28.99	4.43e+89	206.42
92	3.58e+77	178.57	8	3.26e+11	26.51	1.17e+89	205.08
93	1.12e+78	179.72	7	2.44e+10	23.92	2.74e+88	203.63
94	3.50e+78	180.86	6	1.61e+09	21.20	5.64e+87	202.05
95	1.08e+79	181.98	5	9.20e+07	18.34	9.97e+86	200.32
96	3.33e+79	183.11	4	4.42e+06	15.30	1.47e+86	198.41
97	1.02e+80	184.22	3	1.72e+05	12.05	1.75e+85	196.28
98	3.08e+80	185.33	2	5.05e+03	8.53	1.56e+84	193.86
99	9.27e+80	186.43	1	1.00e+02	4.61	9.27e+82	191.04
100	2.77e+81	187.53	0	1.00e+00	0.00	2.77e+81	187.53

The red curve corresponds to A, blue corresponds to B, and green corresponds to the total entropy.



This program was coded in Python 2.7

```
# Matt Meyers, PHYS 309-010

import numpy as np
import matplotlib.pyplot as plt
import math

# Assigning constants for number of oscillators and total energy
N_A = 200
N_B = 100
Q = 100

# Putting macrostates for a and b into lists
q_a = np.arange(0, Q + 1)
q_b = Q - q_a

# Calculating omega for a and b and assigning these values to lists
omegaA = np.empty(shape=[Q + 1])
for i in range(0, Q + 1):
   omegaA[i] = (math.factorial(q_a[i] + N_A - 1)) / (math.factorial(q_a[i]) * math.factorial(N_A - 1))

omegaB = np.empty(shape=[Q + 1])
for i in range(0, Q + 1):
   omegaB[i] = (math.factorial(q_b[i] + N_B - 1)) / (math.factorial(q_b[i]) * math.factorial(N_B - 1))

# Omega total is the product of omega_a and omega_b
omegaTot = omegaA * omegaB

#Printing table for testing
for i in range(0,Q +1):
   print(q_a[i],"\t","{:.2e}".format(omegaA[i]),"\t",q_b[i],"\t",
        "{:.2e}".format(omegaB[i]),"\t","{:.2e}".format(omegaTot[i]))

# Plotting Omega_total versus number of macrostates for a
plt.plot(q_a, omegaTot)
plt.rc('text', usetex=True)
plt.xlabel(r"$q_A$")
plt.ylabel(r"$\Omega_{total}$")
plt.savefig("MicrostatesPlot.png")
plt.show()

# The most probable macrostate would be q_a = 67.  The probability of this would be the following.
mostProbable = omegaTot[67] / sum(omegaTot)
```

```
print("{:.2e}".format(mostProbable))

# The least probable macrostate would be q_a = 0.  The probability of this would be the following.
leastProbable = omegaTot[0] / sum(omegaTot)
print("{:.2e}".format(leastProbable))

# Calculating entropy for a and b.  Then saving these values to lists  in terms of Boltzmann's constant, the entropy is
# the natural log of the number of microstates.
s_a = np.empty(shape=[Q + 1])
for i in range(0, Q + 1):
   s_a[i] = math.log(omegaA[i])
s_b = np.empty(shape=[Q + 1])
for i in range(0, Q + 1):
   s_b[i] = math.log(omegaB[i])
s_tot = np.empty(shape=[Q + 1])
for i in range(0, Q + 1):
   s_tot[i] = math.log(omegaTot[i])

# Plotting the entropies against the number of macrostates for a. A is the red curve, B is blue, and the total is green.
plt.plot(q_a, s_a, 'r', q_a, s_b, 'b', q_a, s_tot, 'g')
plt.rc('text', usetex=True)
plt.xlabel(r"$q_A$")
plt.ylabel(r"Entropy ($S/k$)")
plt.savefig("EntropyPlot.png")
plt.show()

# Formatting the data for omega and writing to a text file.
file = open("MicrostateTable.txt", "w")
for i in range(0,102):
   if i == 0:
      file.write("q_a\tomega_a\t\tq_b\tomega_b\t\tomega_tot")
      file.write("\n")
   else:
      file.write(str(q_a[i-1])+"\t"+str("{:.2e}".format(omegaA[i-1]))+"\t"+str(q_b[i-1])+"\t"+
            str("{:.2e}".format(omegaB[i-1]))+"\t"+str("{:.2e}".format(omegaTot[i-1])))
      file.write("\n")

# Formatting the entropy data and printing to a text file.
file = open("EntropyTable.txt", "w")
for i in range(0,102):
   if i == 0:
      file.write("q_a\tomega_a\t\tS_a/k\tq_b\tomega_b\t\tS_b/k\t\tomega_tot\tS_tot/k")
```

```
        file.write("\n")
    else:
        file.write(str(q_a[i-1])+"\t"+str("{:.2e}".format(omegaA[i-1]))+"\t"+str("{:.2f}".format(s_a[i-1]))
                +"\t"+str(q_b[i-1])+"\t"+str("{:.2e}".format(omegaB[i-1]))+"\t"+str("{:.2f}".format(s_b[i-
1]))+"\t\t"
                +str("{:.2e}".format(omegaTot[i-1]))+"\t"+str("{:.2f}".format(s_tot[i-1])))
        file.write("\n")
```