Sample output from my solution to Problem #1:
(yours should match the format: the times depend on your machine's speed).

```
Nearest Neighbor, size = 100
Analysis of 5 timings
avg = 0.00097   min = 0.00085  max = 0.00117  span = 33.0%

   Time Ranges
8.49e-04<>8.81e-04[ 20.0%]|**************************************************
8.81e-04<>9.13e-04[ 20.0%]|**************************************************
9.13e-04<>9.45e-04[ 20.0%]|**************************************************
9.45e-04<>9.77e-04[  0.0%]|A
9.77e-04<>1.01e-03[ 20.0%]|**************************************************
1.01e-03<>1.04e-03[  0.0%]|
1.04e-03<>1.07e-03[  0.0%]|
1.07e-03<>1.11e-03[  0.0%]|
1.11e-03<>1.14e-03[  0.0%]|
1.14e-03<>1.17e-03[  0.0%]|
1.17e-03<>1.20e-03[ 20.0%]|**************************************************

Nearest Neighbor, size = 200
Analysis of 5 timings
avg = 0.00209   min = 0.00201  max = 0.00219  span = 8.7%

   Time Ranges
2.01e-03<>2.03e-03[ 40.0%]|**************************************************
2.03e-03<>2.05e-03[  0.0%]|
2.05e-03<>2.06e-03[  0.0%]|
2.06e-03<>2.08e-03[  0.0%]|
2.08e-03<>2.10e-03[ 20.0%]|*************************A
2.10e-03<>2.12e-03[  0.0%]|
2.12e-03<>2.14e-03[  0.0%]|
2.14e-03<>2.15e-03[  0.0%]|
2.15e-03<>2.17e-03[ 20.0%]|*************************
2.17e-03<>2.19e-03[  0.0%]|
2.19e-03<>2.21e-03[ 20.0%]|*************************

Nearest Neighbor, size = 400
Analysis of 5 timings
avg = 0.00433   min = 0.00426  max = 0.00440  span = 3.4%

   Time Ranges
4.26e-03<>4.27e-03[ 20.0%]|*************************
4.27e-03<>4.29e-03[  0.0%]|
4.29e-03<>4.30e-03[  0.0%]|
4.30e-03<>4.32e-03[  0.0%]|
4.32e-03<>4.33e-03[ 20.0%]|*************************
4.33e-03<>4.34e-03[ 40.0%]|**************************************************A
4.34e-03<>4.36e-03[  0.0%]|
4.36e-03<>4.37e-03[  0.0%]|
4.37e-03<>4.39e-03[  0.0%]|
4.39e-03<>4.40e-03[  0.0%]|
4.40e-03<>4.42e-03[ 20.0%]|*************************

Nearest Neighbor, size = 800
Analysis of 5 timings
avg = 0.00968   min = 0.00947  max = 0.00995  span = 5.0%

   Time Ranges
9.47e-03<>9.52e-03[ 20.0%]|*************************
```

```
9.52e-03<>9.57e-03[  0.0%]|
9.57e-03<>9.61e-03[  0.0%]|
9.61e-03<>9.66e-03[ 20.0%]|*************************
9.66e-03<>9.71e-03[ 40.0%]|**************************************************A
9.71e-03<>9.76e-03[  0.0%]|
9.76e-03<>9.81e-03[  0.0%]|
9.81e-03<>9.85e-03[  0.0%]|
9.85e-03<>9.90e-03[  0.0%]|
9.90e-03<>9.95e-03[  0.0%]|
9.95e-03<>1.00e-02[ 20.0%]|*************************

Nearest Neighbor, size = 1600
Analysis of 5 timings
avg = 0.02231   min = 0.02169  max = 0.02313  span = 6.5%

   Time Ranges
2.17e-02<>2.18e-02[ 20.0%]|*************************
2.18e-02<>2.20e-02[  0.0%]|
2.20e-02<>2.21e-02[ 40.0%]|**************************************************
2.21e-02<>2.23e-02[  0.0%]|
2.23e-02<>2.24e-02[  0.0%]|A
2.24e-02<>2.26e-02[  0.0%]|
2.26e-02<>2.27e-02[ 20.0%]|*************************
2.27e-02<>2.28e-02[  0.0%]|
2.28e-02<>2.30e-02[  0.0%]|
2.30e-02<>2.31e-02[  0.0%]|
2.31e-02<>2.33e-02[ 20.0%]|*************************

Nearest Neighbor, size = 3200
Analysis of 5 timings
avg = 0.04936   min = 0.04533  max = 0.05466  span = 18.9%

   Time Ranges
4.53e-02<>4.63e-02[ 40.0%]|**************************************************
4.63e-02<>4.72e-02[  0.0%]|
4.72e-02<>4.81e-02[ 20.0%]|*************************
4.81e-02<>4.91e-02[  0.0%]|
4.91e-02<>5.00e-02[  0.0%]|A
5.00e-02<>5.09e-02[  0.0%]|
5.09e-02<>5.19e-02[  0.0%]|
5.19e-02<>5.28e-02[  0.0%]|
5.28e-02<>5.37e-02[ 20.0%]|*************************
5.37e-02<>5.47e-02[  0.0%]|
5.47e-02<>5.56e-02[ 20.0%]|*************************

Nearest Neighbor, size = 6400
Analysis of 5 timings
avg = 0.10750   min = 0.10005  max = 0.11775  span = 16.5%

   Time Ranges
1.00e-01<>1.02e-01[ 40.0%]|**************************************************
1.02e-01<>1.04e-01[  0.0%]|
1.04e-01<>1.05e-01[  0.0%]|
1.05e-01<>1.07e-01[  0.0%]|
1.07e-01<>1.09e-01[ 20.0%]|*************************A
1.09e-01<>1.11e-01[ 20.0%]|*************************
1.11e-01<>1.12e-01[  0.0%]|
1.12e-01<>1.14e-01[  0.0%]|
1.14e-01<>1.16e-01[  0.0%]|
1.16e-01<>1.18e-01[  0.0%]|
1.18e-01<>1.20e-01[ 20.0%]|*************************
```

```
Nearest Neighbor, size = 12800
Analysis of 5 timings
avg = 0.21930   min = 0.21385  max = 0.22416  span = 4.7%

   Time Ranges
2.14e-01<>2.15e-01[ 40.0%]|**************************************************
2.15e-01<>2.16e-01[  0.0%]|
2.16e-01<>2.17e-01[  0.0%]|
2.17e-01<>2.18e-01[  0.0%]|
2.18e-01<>2.19e-01[  0.0%]|
2.19e-01<>2.20e-01[  0.0%]|A
2.20e-01<>2.21e-01[ 20.0%]|*************************
2.21e-01<>2.22e-01[  0.0%]|
2.22e-01<>2.23e-01[  0.0%]|
2.23e-01<>2.24e-01[ 20.0%]|*************************
2.24e-01<>2.25e-01[ 20.0%]|*************************

Nearest Neighbor, size = 25600
Analysis of 5 timings
avg = 0.48683   min = 0.46404  max = 0.50300  span = 8.0%

   Time Ranges
4.64e-01<>4.68e-01[ 20.0%]|*************************
4.68e-01<>4.72e-01[  0.0%]|
4.72e-01<>4.76e-01[ 20.0%]|*************************
4.76e-01<>4.80e-01[  0.0%]|
4.80e-01<>4.84e-01[  0.0%]|
4.84e-01<>4.87e-01[  0.0%]|A
4.87e-01<>4.91e-01[  0.0%]|
4.91e-01<>4.95e-01[  0.0%]|
4.95e-01<>4.99e-01[ 40.0%]|**************************************************
4.99e-01<>5.03e-01[  0.0%]|
5.03e-01<>5.07e-01[ 20.0%]|*************************
```

Sample output from my solution to Problem #2:
(yours should match the format: the times/counts depend on your machine’s speed).

```
Fri May 31 08:15:00 2019    test_profile

         3078237 function calls (3045471 primitive calls) in 0.788 seconds

   Ordered by: call count
   List reduced from 20 to 12 due to restriction <12>

   ncalls  tottime  percall  cumtime  percall filename:lineno(function)
   959030    0.042    0.000    0.042    0.000 {built-in method builtins.len}
   581867    0.063    0.000    0.063    0.000 nearestneighbor.py:9(swap)
   411862    0.018    0.000    0.018    0.000 {built-in method builtins.abs}
   374783    0.018    0.000    0.018    0.000 {method 'append' of 'list' objects}
   184320    0.010    0.000    0.010    0.000 nearestneighbor.py:64(<lambda>)
   174080    0.010    0.000    0.010    0.000 nearestneighbor.py:63(<lambda>)
    61884    0.034    0.000    0.040    0.000 nearestneighbor.py:37(dist)
    61884    0.005    0.000    0.005    0.000 {built-in method math.sqrt}
    55544    0.145    0.000    0.208    0.000 nearestneighbor.py:8(partition)
  32767/1    0.320    0.000    0.787    0.787 nearestneighbor.py:36(closest_2d)
    32766    0.007    0.000    0.007    0.000 nearestneighbor.py:38(<listcomp>)
    32766    0.013    0.000    0.025    0.000 nearestneighbor.py:38(min_none)


Fri May 31 08:15:00 2019    test_profile

         3078237 function calls (3045471 primitive calls) in 0.788 seconds

   Ordered by: internal time
   List reduced from 20 to 12 due to restriction <12>

   ncalls  tottime  percall  cumtime  percall filename:lineno(function)
  32767/1    0.320    0.000    0.787    0.787 nearestneighbor.py:36(closest_2d)
    55544    0.145    0.000    0.208    0.000 nearestneighbor.py:8(partition)
   581867    0.063    0.000    0.063    0.000 nearestneighbor.py:9(swap)
    32766    0.062    0.000    0.083    0.000 {method 'sort' of 'list' objects}
   959030    0.042    0.000    0.042    0.000 {built-in method builtins.len}
    61884    0.034    0.000    0.040    0.000 nearestneighbor.py:37(dist)
    16383    0.019    0.000    0.227    0.000 nearestneighbor.py:21(select)
   411862    0.018    0.000    0.018    0.000 {built-in method builtins.abs}
   374783    0.018    0.000    0.018    0.000 {method 'append' of 'list' objects}
    16383    0.014    0.000    0.014    0.000 nearestneighbor.py:44(<listcomp>)
    32766    0.013    0.000    0.025    0.000 nearestneighbor.py:38(min_none)
   174080    0.010    0.000    0.010    0.000 nearestneighbor.py:63(<lambda>)
```