Testing the speed of the insertion sort algorithm

Homework #2

By

John Robertson

CS 303 Algorithms and Data Structures

September 4, 2019

1. **Problem Specification**

This assignment is supposed to measure the speed of insertion sorts on large inputs.

2. **Program Design**

This program has a file with the insertion sort function and a driver file to test it.

The following steps were required to develop this program:

1. Implement insertion sort as the ins\_sort() function.
2. Import the above into the driver.
3. Import the files in a loop in order to run ins\_sort() on their contents.
4. Print how long doing that takes.

3. **Testing Plan**

In order to have small inputs to test on, I tested ins\_sort() in its own file against Python’s built in sort() function to see if it is correct. After being satisfied with the results, I went straight to testing on the input files.

4. **Test Cases**

I initially tested the inputs on lists eight elements long of random values one through twenty. I then moved up to four hundred elements and a range of four hundred, repeating this one thousand times and comparing the output with the sort() built in function. I next tested on zero elements, one element, and a descending order array; these arrays sorted by my insertion sort function returned the same array that sort() returned.

The rest of the cases are from the input files.

5. **Analysis and Conclusions**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Number of inputs | 100 | 1000 | 5000 | 10000 | 50000 | 100000 | 500000 |
| Time in seconds | 0.0024714000000000125 | 0.2704601 | 0.7821910999999999 | 27.0296733 | 400.4907278 (7 minutes) | 1343.2281662 (23 minutes) | 36636.953898700005 (10 hours) |

As can be shown by the chart and the graph, the growth is exponential. Time 2 is more than double time 1, and so on. For very large inputs that need to be sorted, insertion sort takes too much time. For large inputs, a faster algorithm may be needed, perhaps sacrificing space.

6. **References**

The following references were used:

Dr. Unan’s CS303\_02\_Insertion\_Sort slide

https://stackoverflow.com/questions/1614236/in-python-how-do-i-convert-all-of-the-items-in-a-list-to-floats

The C implementation of insertion sort I wrote for CS 330 based on the pdf Dr. Unan gave in that class