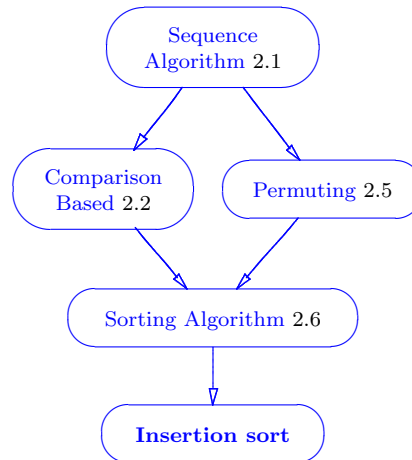


2.7 Insertion Sort

Section authors: Bart Ingleston, John Schmiederer, Richard Spackmann



Refinement of: Sequence Sorting Algorithm (§2.6), therefore of Comparison Based (§2.2), Permuting (§2.5), Sequence Algorithm (§2.1).

Prototype: `template<class RandomAccessIterator>`
`void insertion_sort(`
 `RandomAccessIterator first,`
 `RandomAccessIterator last)`

Effects: Standard effects of a Sequence Sorting Algorithm (§2.6). In brief: the elements in `[first, last)` after execution are a permutation of the original element in the range, and they are in nondecreasing order according the comparison operator.

Asymptotic complexity: Let $N = \text{last} - \text{first}$.

- Best case (almost sorted data): $O(N)$
- Average case (random data): $O(N^2)$
- Worst case: $O(N^2)$

Complexity in terms of operation counts:

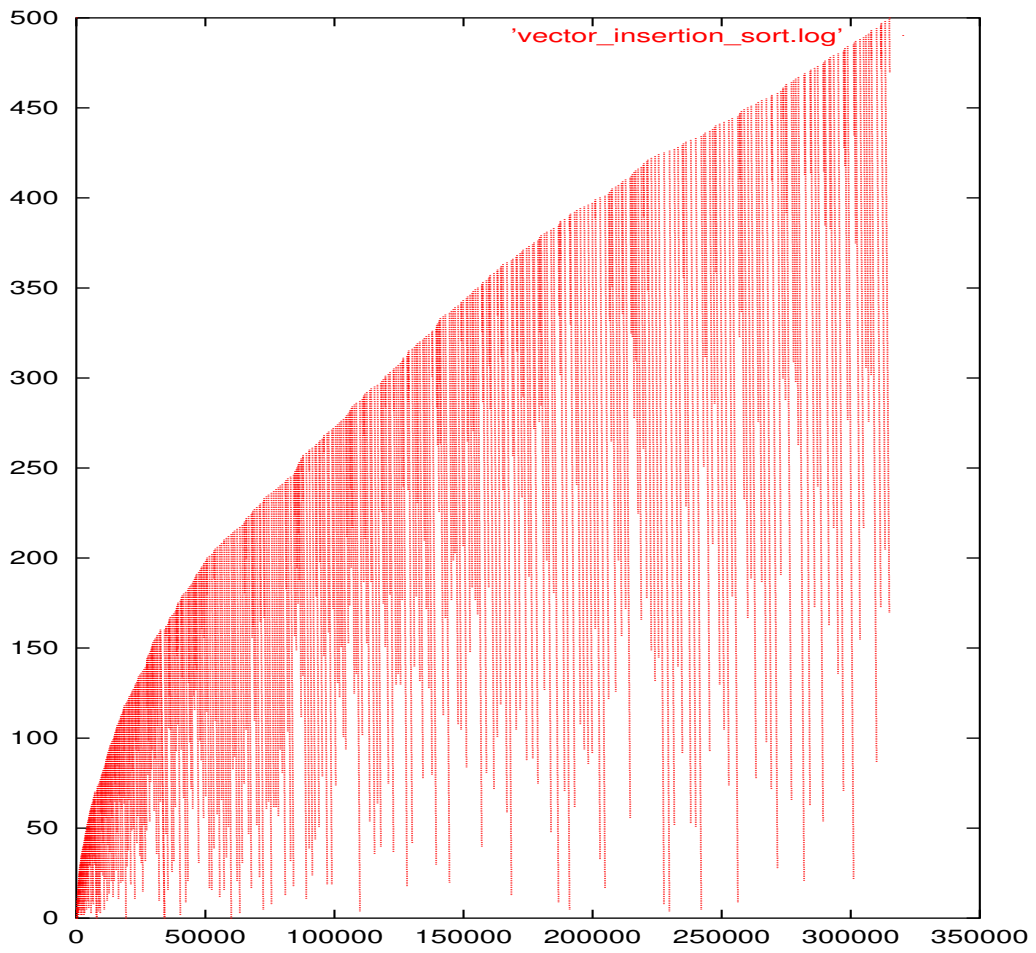
- Average case (Random data):

Table 1: Performance of Insertion Sort on Random Sequences (Sizes and Operations Counts in Multiples of 1,000)

Size	Comparisons	Assignments	Iterator Ops	Integer Ops	Total Ops
0.01	0.042	0.032	0.192	0.004	0.27
0.02	0.347	0.137	0.728	0.028	1.028
0.04	1.16	0.545	2.627	0.014	3.697
0.08	3.729	1.759	8.91	0.106	12.55
0.16	13.089	6.444	31.655	0.016	44.405
0.32	53.698	26.285	133.04	0.711	186.686
0.64	208.759	103.899	518.068	0.278	725.768
1.28	804.147	400.042	2005.88	2.686	2810.03
2.56	3218.13	1605.68	8034.75	4.14	11251.8
5.12	13118.6	6553.5	32772.5	6.366	45887.1
10.24	52935.3	26456.6	132289	11.807	185216
20.48	211044	105490	527540	42.943	738585

- Found Coefficients:
 - Value comparisons: $0.2513N^2 + 5.4887N + 9181.6$
 - Value assignments: $0.5027N^2 + 12.5058N + 17849$
 - Iterator operations: $1.2567N^2 + 25.5456N - 43971$
 - Integer operations: $.0001N^2 + 0.5357N + 471.75$
- See also Sorting Algorithm Operation Counts (§2.91) for sample counts on random data for other sorting algorithms. See Appendix A for Insertion Sort operation count graphs.

Iterator trace plot:



500 random elements sorted by insertion sort as implemented in SGI STL.

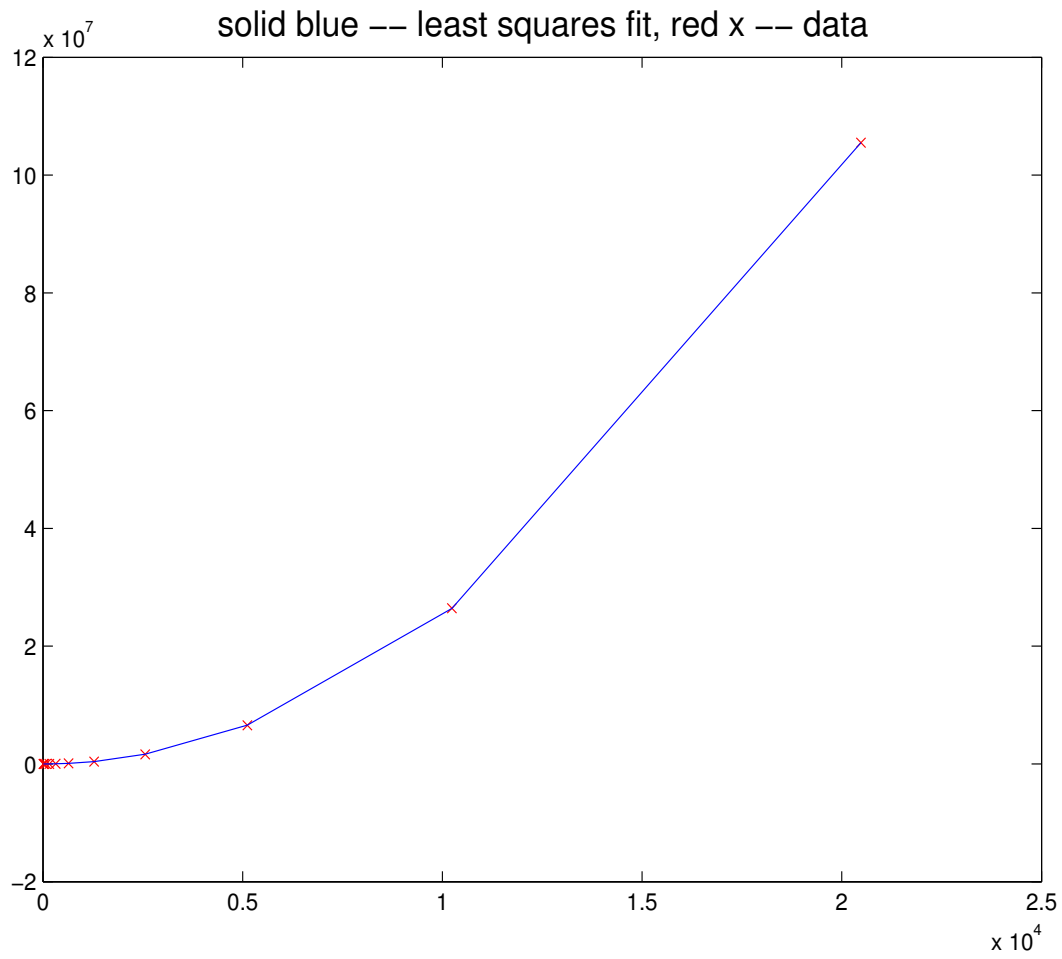
Algorithm animations: The Animator

The Animator is a Web-based, Java Applet algorithm animator developed at Hope College by Peter Brummund. The Animator implements a few common sequence algorithms and provides a good visual representation of the algorithms. It allows the user to watch the values be sorted while stepping through the implemented java source code. Insertion sort is one of the implemented sequence sorting algorithms. Then Animator can be found at

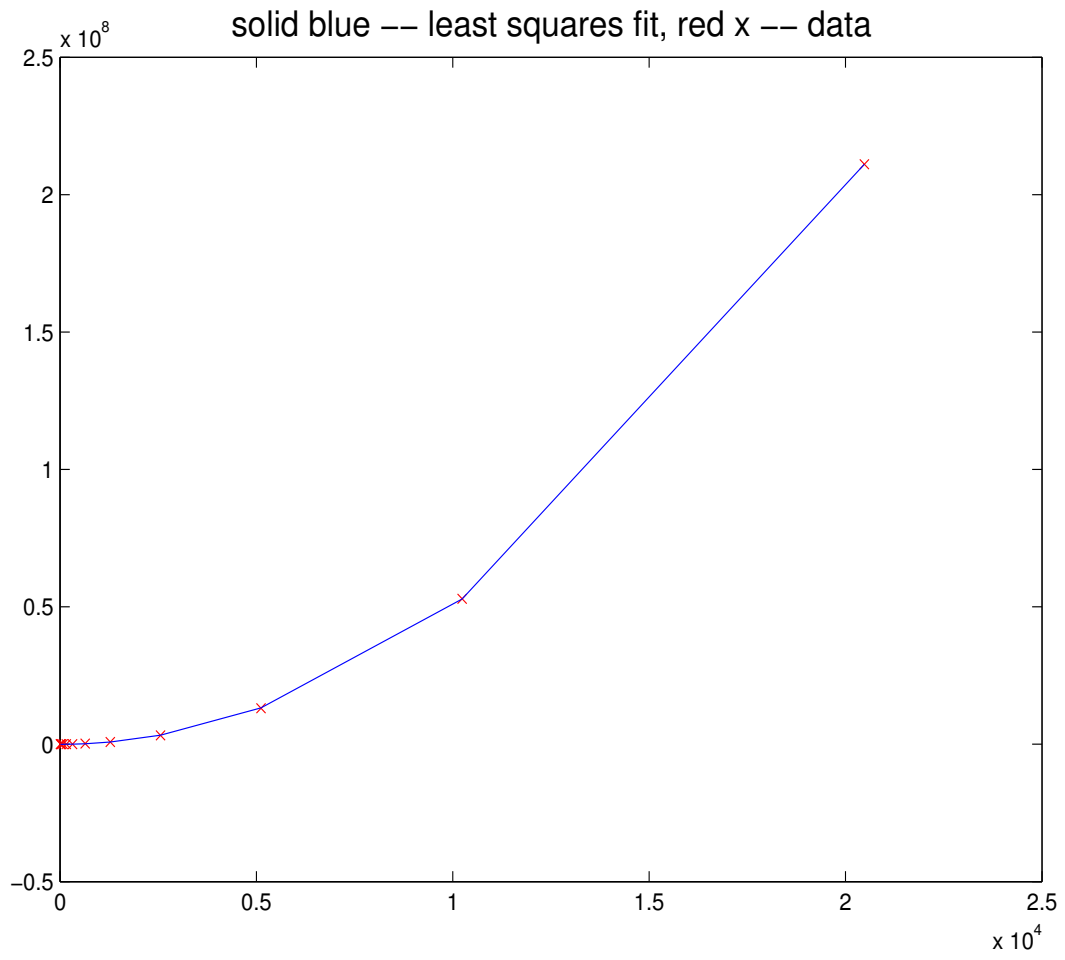
<http://www.cs.hope.edu/alganim/animator/Animator.html>.

Appendix A: Operation Count Graphs (using counters.h)

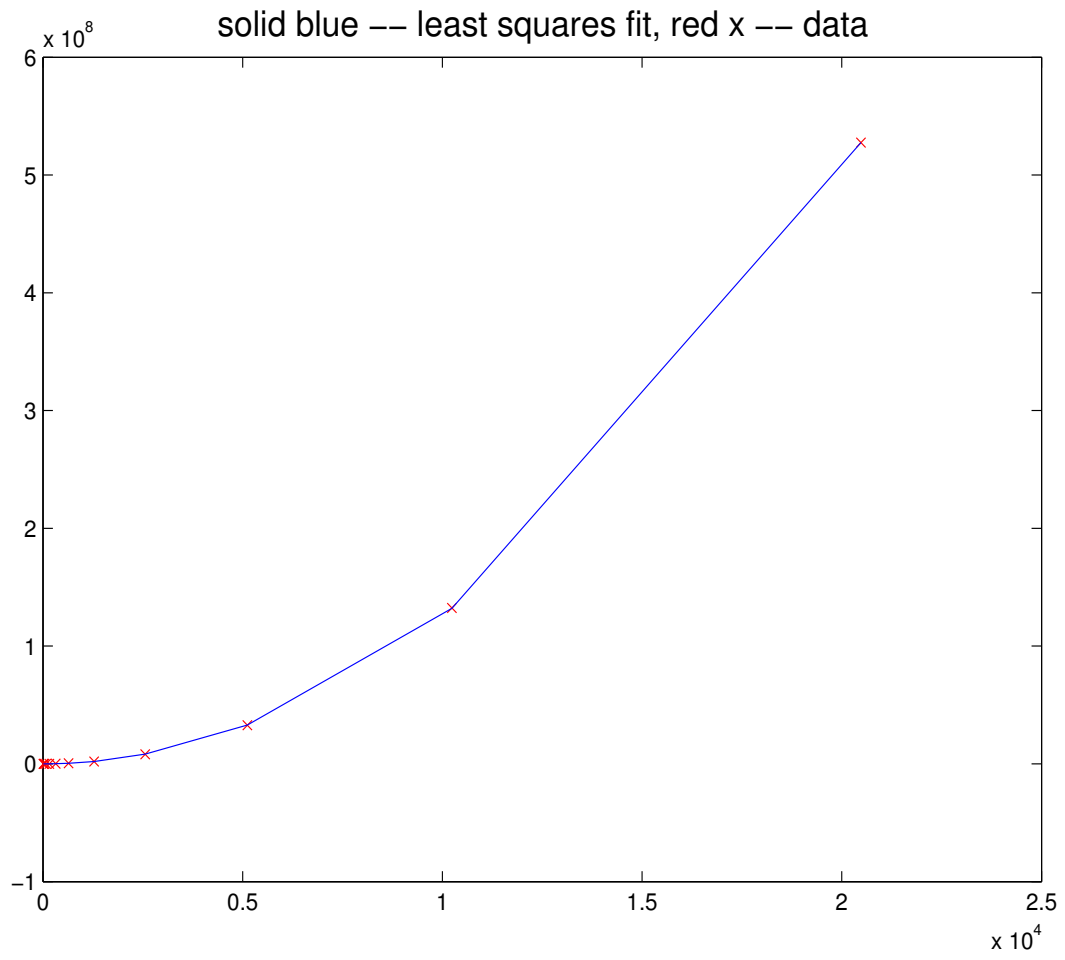
Value Assignments:



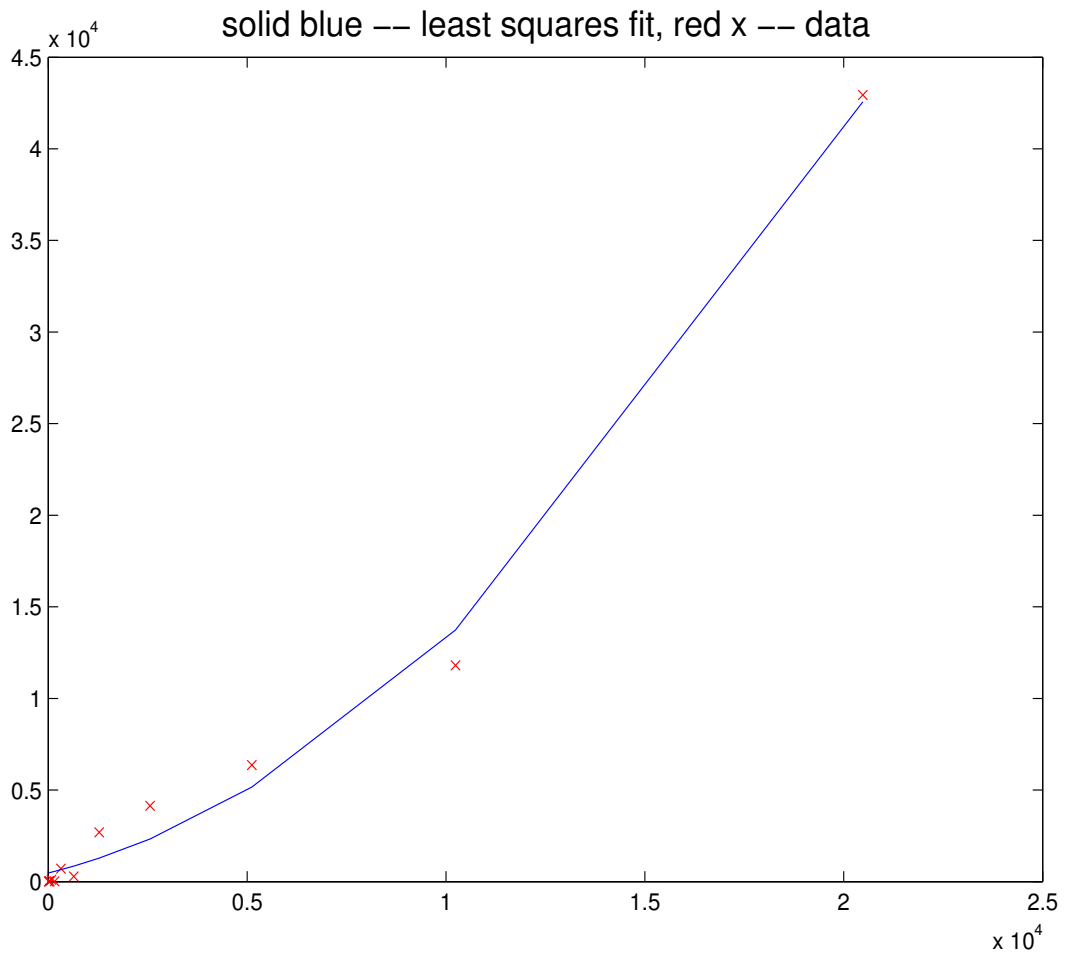
Value Comparisons:



Iterator Operations:



Integer Operations:



Total Operations:

