Principles for Experimental Algorithmics

Michael T. Goodrich

based, in part, on the following papers:

Towards a Discipline of Experimental Algorithmics, by Bernard M.E. Moret
A Theoretician’s Guide to the Experimental Analysis of Algorithms, by David S. Johnson
Experimental Algorithmics

- Experimental Algorithmics studies algorithms and data structures by joining experimental studies with the traditional theoretical analyses.
  - Scientists do experiments because they have no choice
  - In experimental algorithmics we combine theoretical analysis with experimentation.
Experimental Algorithmics

• Experimentation with algorithms and data structures can prove to be indispensable for the following tasks:
  – The assessment of heuristics for hard problems
  – The characterization of asymptotic behavior of complex algorithms
  – The comparison of competing designs for tractable problems
  – The formulation of new combinatorial conjectures
  – The evaluation of optimization criteria
  – The transfer of research results from paper to production code
Perform Worthwhile Experiments

• Ask questions worth asking
  – New problems
  – New algorithms
  – New types of input distributions
  – New types of computer hardware
Measuring Actual Performance

• Random instances should be motivated from real-world data
• Also use real-world data when possible

http://dimacs.rutgers.edu/programs/challenge/

http://snap.stanford.edu/
Testing the Quality of Solutions

• Find a parameter that can be effectively tested experimentally
  – Waste in bin packing
  – Closeness to a known lower bound
Experimental Setup

- Have clear objectives
- Gather data to answer the questions posed
- Choose hardware appropriately
- Code solutions consistently to allow for good conclusions
- Generate useful problem instances
- Analyze your data
Understand Your Hardware: The Memory Hierarchy

- The trade-off of size and speed

Diagram:
- Disk: 1 TB, 10 ms
- Memory: 4 GB, 100 ns
- L3 cache: 5 ns
- L2 cache: 0.5 ns
- L1 cache: core
Ensure Reproducibility

• Unless it is truly confidential, post your code for others to use.
• For random data, post how you generated it
• For real-world data, post how to get it
Ensure Comparability

• Perform all experiments on the same hardware
• Report the type of hardware used
• Code all algorithms with the same level of code optimizations and tuning
Present your Data in Meaningful Ways

- Use tables only for small data sets

<table>
<thead>
<tr>
<th>Algorithm</th>
<th>100</th>
<th>316</th>
<th>1,000</th>
<th>3,162</th>
<th>10,000</th>
<th>31,623</th>
<th>100,000</th>
<th>316,227</th>
<th>1,000,000</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>0.00</td>
<td>0.02</td>
<td>0.08</td>
<td>0.29</td>
<td>1.05</td>
<td>5.46</td>
<td>23.0</td>
<td>89.6</td>
<td>377</td>
</tr>
<tr>
<td>B</td>
<td>0.00</td>
<td>0.03</td>
<td>0.11</td>
<td>0.35</td>
<td>1.38</td>
<td>6.50</td>
<td>30.6</td>
<td>173.3</td>
<td>669</td>
</tr>
<tr>
<td>C</td>
<td>0.01</td>
<td>0.06</td>
<td>0.21</td>
<td>0.71</td>
<td>2.79</td>
<td>10.98</td>
<td>42.7</td>
<td>329.5</td>
<td>1253</td>
</tr>
<tr>
<td>D</td>
<td>0.02</td>
<td>0.09</td>
<td>0.43</td>
<td>1.64</td>
<td>6.98</td>
<td>37.51</td>
<td>192.4</td>
<td>789.7</td>
<td>5465</td>
</tr>
<tr>
<td>E</td>
<td>0.03</td>
<td>0.14</td>
<td>0.57</td>
<td>2.14</td>
<td>10.42</td>
<td>55.36</td>
<td>369.4</td>
<td>5775.0</td>
<td>33414</td>
</tr>
</tbody>
</table>