Programming Equations

- **Algorithms + Data Structures = Programs**
  - Niklaus Wirth, old view of programming and compsci

- **How old are algorithms?**
  - Euclid: find greatest common divisor, GCD (56, 217)
  - Who cares?

- **What about sorting? Why is it important?**
  - Old: Bubble sort, selection sort, insertion sort
  - Old: Heapsort, mergesort, quicksort
  - New: introsort, hybrid mergesort, timsort!

Tim Peters: Zen of Python

- Beautiful is better than ugly.
- Explicit is better than implicit.
- Simple is better than complex.
- Complex is better than complicated.
- Flat is better than nested.
- Readability counts.
- Special cases aren't special enough to break the rules.
- In the face of ambiguity, refuse the temptation to guess.

What did I learn today?

- **timsort is standard on...**
  - Python as of version 2.3, Android, Java 7
    - Adaptive, stable, natural mergesort with supernatural performance

- **What is mergesort?**
  - Stable, what does that mean?
  - Divide and Conquer, what does that mean?
  - Nothing is faster, what does that mean?
  - Quicksort is faster, what does that mean?
  - See Merge.py

Comparing Algorithms

- **Searching a list of N elements for a specific value**
  - Worst case is ....

- **Doing binary search (guess a number), sorted list**
  - Worst case is ...

- **Sorting using merge sort**
  - Worst case is ...

- **Naïve finding which ball collides with specific ball**
  - Worst case is ...
Jim Gray (Turing 1998)

- Bubble sort is a good argument for analyzing algorithm performance. It is a perfectly correct algorithm. But it's performance is among the worst imaginable. So, it crisply shows the difference between correct algorithms and good algorithms.

 Italics (italics ola's)
Brian Reid (Hopper Award 1982)

Feah. I love bubble sort, and I grow weary of people who have nothing better to do than to preach about it. Universities are good places to keep such people, so that they don't scare the general public.

(continued)

Brian Reid (Hopper 1982)

I am quite capable of squaring N with or without a calculator, and I know how long my sorts will bubble. I can type every form of bubble sort into a text editor from memory. If I am writing some quick code and I need a sort quick, as opposed to a quick sort, I just type in the bubble sort as if it were a statement. I'm done with it before I could look up the data type of the third argument to the quicksort library.

I have a dual-processor 1.2 GHz Powermac and it sneers at your N squared for most interesting values of N. And my source code is smaller than yours.

Brian Reid who keeps all of his bubbles sorted anyhow.

Niklaus Wirth (Turing award 1984)

I have read your article and share your view that Bubble Sort has hardly any merits. I think that it is so often mentioned, because it illustrates quite well the principle of sorting by exchanging.

I think BS is popular, because it fits well into a systematic development of sorting algorithms. But it plays no role in actual applications. Quite in contrast to C which without merit (and its derivative Java), among programming codes.

Owen O’ Malley

- Debugging can be frustrating, but very rewarding when you find the cause of the problem. One of the nastiest bugs that I’ve found when I was at NASA and the Mars Explorer Rovers had just landed on Mars.

http://sortbenchmark.org/

Hadoop sets Terabyte sort record
- Java
- 34252 nodes
- Greedy!