# Scientific Computing

Ramani Duraiswami

# Prerequisites

- Undergraduate numerical methods or numerical analysis course (460 or 466)
- Background Needed
  - Linear Algebra
  - Numerical Analysis
  - Programming
    - Matlab, C/C++ and/or FORTRAN 9x
  - Experience and/or interest in scientific computing
- Participation essential!
- Web site and Piazza for information

# Homework

- Will try to have it at least every other week
- Will not be excessive
- Essential for learning --- must do as opposed to just read.
- Homework handed out last class of a week.
- Due last class of next week
- Thanksgiving week no homework
- 25%

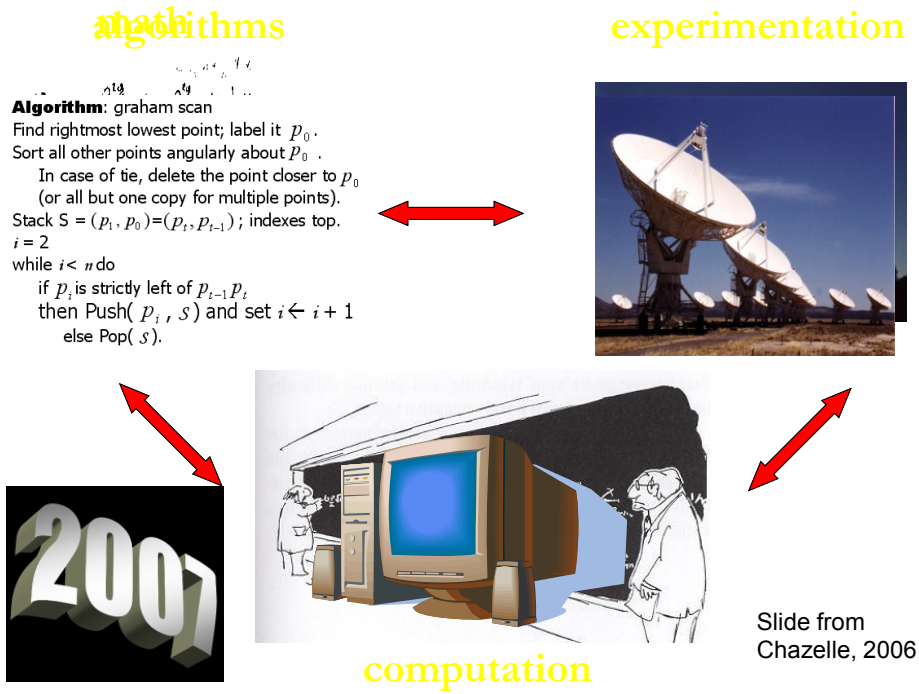
# Projects & Exams

- There will be a final project that will require you to implement a case study as in the book
  - account for 25% of the grade.
- Project to be chosen latest by October 15.
  - If you already have a project in mind you can discuss it with me
- Exams
  - No final
  - first exam worth 20%, October 10
  - Second exam worth 25 %, November 21

# Scientific Computing

## Big Picture

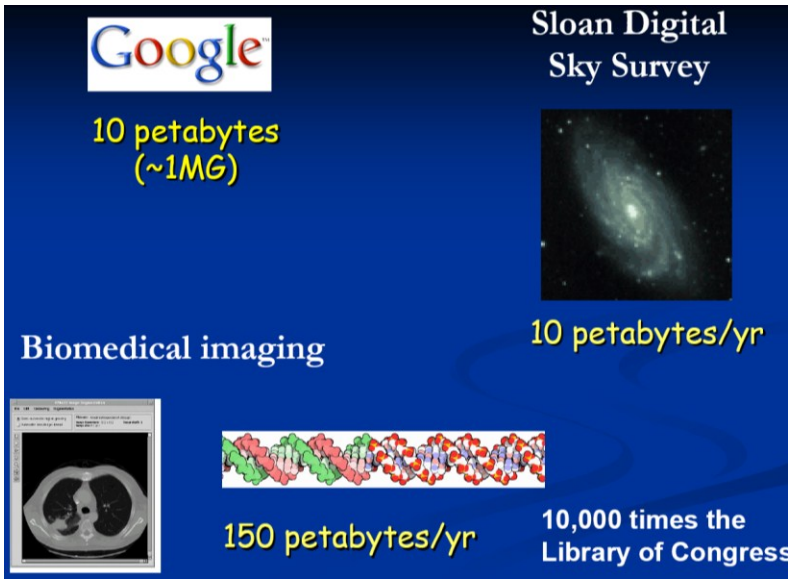
- Object of all science
- Efficiency and better understanding
- Scientific Method: Experiment/Hypothesis
- Now Simulation/Hypothesis




Slide from Chazelle, 2006

# Problem Sizes Continue to Grow in all Fields

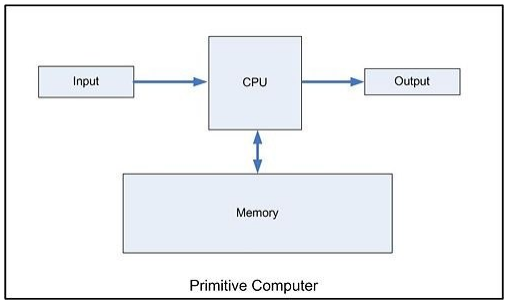
- Sensors are getting varied and cheaper; and storage is getting cheaper
- Cameras, microphones
- Text (all the newspapers, books, technical papers)
- Genome data
- Medical/biological data (X-Ray, PET, MRI, Ultrasound, Electron microscopy …)
- Climate (Temperature, Salinity, Pressure, Wind, Oxygen content, …)
- Finer detail in meshes



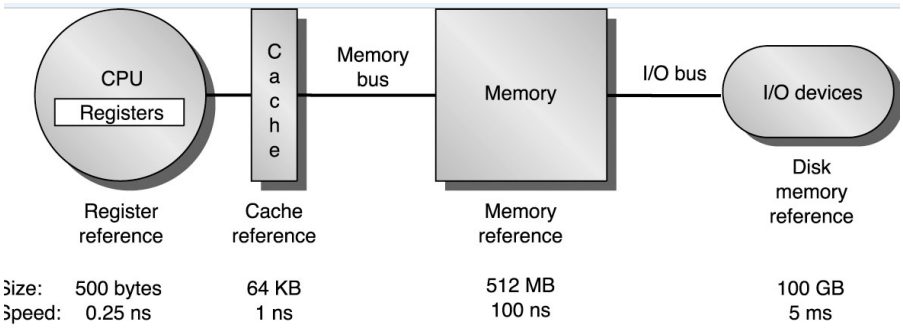

- 2006, B. Chazelle

# Good workmen know their tools

- Primitive model



# Memory Hierarchy




## Norvig: Numbers every programmer should know

- L1 cache reference 0.5 ns
  - Branch mispredict 5 ns
- L2 cache reference 7 ns 14x L1 cache
- Mutex lock/unlock 25 ns
- Main memory reference 100 ns 20x L2 cache, 200x L1 cache
- Compress 1K bytes with Zippy 3,000 ns
- Send 1K bytes over 1 Gbps network 10,000 ns 0.01 ms
- Read 4K randomly from SSD* 150,000 ns 0.15 ms
- Read 1 MB sequentially from memory 250,000 ns 0.25 ms
- Round trip within same datacenter 500,000 ns 0.5 ms
- Read 1 MB sequentially from SSD* 1,000,000 ns 1 ms 4X memory
- Disk seek 10,000,000 ns 10 ms 20x datacenter roundtrip
- Read 1 MB sequentially from disk 20,000,000 ns 20 ms 80x memory, 20X SSD
- Send packet CA->Netherlands->CA 150,000,000 ns 150 ms

# Numbers every programmer should know

- http://www.eecs.berkeley.edu/~rcs/research/interactive_latency.html