

## Outline

- Announcements:
  - HW III due today!
- Trend I: Parallelism
- Trend II: Matlab
- Where to go from here

---

---

---

---

---

---

---

---

## Trend I: Parallelism

- Computers get faster every year
- but, scientists' appetites will always exceed available resources
  - want to run higher resolution
  - want to run for longer
- One solution is parallelism:
  - divide problem into N pieces
  - give each piece to a separate computer
  - Theoretically, could run N-times faster

---

---

---

---

---

---

---

---

## Easy Parallelism

- Example: run small program on lots of different inputs
  - can get final answer in  $1/N$  times if you have N computers
  - Any "parallel" computer (Cray, Velocity, cluster, internet) will work
  - Example: [SETI@home](#)
    - Problem is to analyze lots of radio data for ET
    - data is divided into small chunks which are sent to computers around the world
    - these computers run SETI screen saver which analyzes small chunks

---

---

---

---

---

---

---

---

## Hard Parallelism

- When divided into pieces, most scientific problems require data from other pieces
  - Need a way of sending data from one piece to another
    - shared memory
    - network
  - Requires programming (tell computer what information to share and when)
  - Requires special systems with fast communication
    - Dual processor PCs---->256 processor SGI or IBM systems
    - Clusters of smaller units like Cornell's Velocity

---

---

---

---

---

---

---

---

## Trend II: Matlab

- Matlab is an environment for scientific computing
  - programming language
  - graphics
  - lots of built-in functions (linear algebra, statistics, ODE solvers, etc.)
- It is very easy to do computational science in Matlab
  - easy to program
  - easy to debug
  - easy to analyze data

---

---

---

---

---

---

---

---

## Matlab

- Matlab's only disadvantage: slower than compiled languages, but
  - Much faster development (prototyping)
  - Fewer bugs (no need to write PCGN, could just call CGM)
  - Easier to validate
- Alternatives to Matlab
  - IDL
  - S+

---

---

---

---

---

---

---

---

## Where to go from here

<b>If interested in</b>	<b>then take</b>
numerical methods	CS 322, or CS 621-24
parallelism	CTC workshops <a href="http://www.tc.cornell.edu">www.tc.cornell.edu</a>
algorithms	CS409 (for scientists) CS481 (more theory)
Matlab	CIS401-402
More development	CIS404 Libraries--starts after spring break!

---

---

---

---

---

---

---

---