How computers work
Fundamentally, computers:

Data → Processing → Data
The romantic view

Data → Processing → Results!!
The romantic view, mixed with reality

- Your lab mates' data
- Your data
- Public data
- Processing
- …more data.
The optimistic view

Your lab mates' data

Your data → Processing → Results!!

Public data

Publication; PhD; fame; glory; money!!!
The sad reality

Your lab mates' data

Processing (memory/RAM, CPU)

Your data

Public data

Disk

Disk/Network

Disk

…more data.
Issues to be raised...

• How do we keep track of what we’ve done?
• How do we know that what we’ve done is right?
  – Correct process?
  – Computer actually worked?
• How can we do things faster/more efficiently?
• How do we tell the computer what to do, anyway??
Issues to be raised...

• How do we keep track of what we’ve done?
• How do we know that what we’ve done is right?
  – Correct process?
  – Computer actually worked?
• How can we do things faster/more efficiently?
• How do we tell the computer what to do, anyway??
Issues to be raised...

• How do we keep track of what we’ve done?
• How do we know that what we’ve done is right?
  – Correct process?
  – Computer actually worked?
• How can we do things faster/more efficiently?
• How do we tell the computer what to do, anyway??
Efficiency

• Cost/resource tradeoff.
• Faster and slower ways of doing things.
• Computers don’t work like people :)  
  – Simple, automated, explicit tasks => computer  
  – Judgement, intuition, creativity => human
Computer architecture

Computation is cheap

Working memory (RAM)
~1-100 gb
- expensive

100-1000 mb/s

~1-10 mb/s

Disk
(.25-20 tb)
cheap

Network
- cheap
Communication between CPUs is pretty cheap.
Computer architecture

Retrieving/sending data from/to the disk or the network is slow.
Computer architecture

Retrieving/sending data from/to the disk or the network is slow.

Working memory (RAM)
~1-100 gb
- expensive

100-1000 mb/s

~1-10 mb/s

Disk
(.25-20 tb)
cheap

Network
- cheap
Issues to be raised...

• How do we keep track of what we’ve done?
• How do we know that what we’ve done is right?
  – Correct process?
  – Computer actually worked?
• How can we do things faster/more efficiently?
• **How do we tell the computer what to do, anyway??**
Telling the computer what to do

• Translating your will => computer effort is technically challenging!
  – Shell commands (a few days to a few months)
  – Scripting (a few weeks to a few years)
  – Programming (a few months to a lifetime)

• Have to meet the computer halfway…

• Most important bit? Manipulating files and executing programs => “shell”