

CS IJC3  
~~Introduction to~~  
Computational Thinking

Christopher Anand  
[anandc@mcmaster.ca](mailto:anandc@mcmaster.ca)

<http://www.cas.mcmaster.ca/~anand/CSIJC32015.html>

# World's Total CPU Power: One Human Brain

By [Ars Technica](#) February 11, 2011 | 1:30 pm | Categories: [Tech](#)



*By John Timmer, Ars Technica*

# Kurzweil predicts personal computers with the power of the human brain by 2025



Tweet

8



Like



0



Share

*This is the second post in a series summarising the key arguments of Ray Kurzweil's **The Singularity is near: When humans transcend biology**. The first post entitled Ray Kurzweil, The Singularity and the accelerating pace of progress can be found [here](#).*

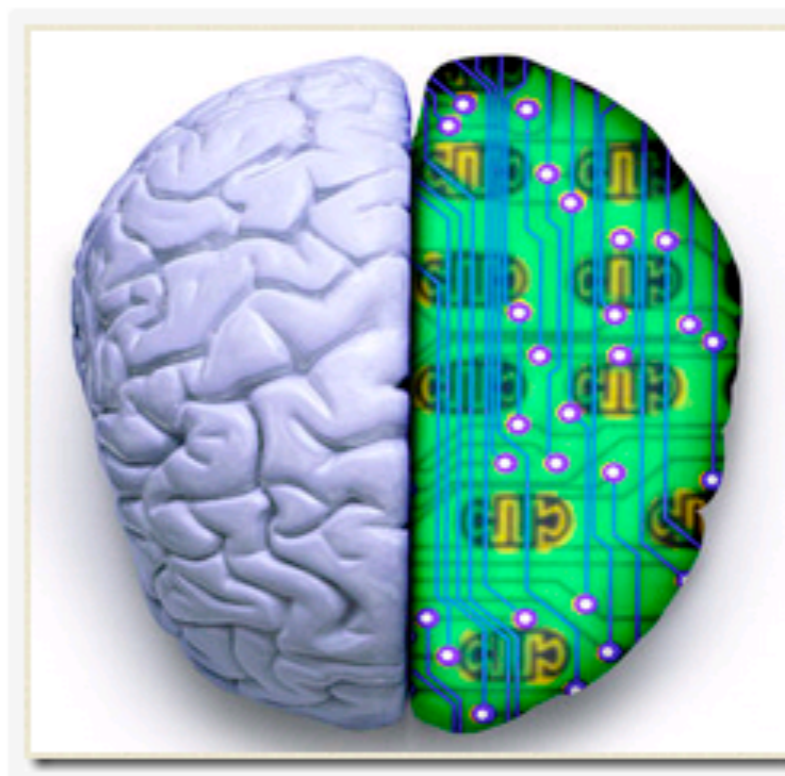
---

The second idea I'm going to pick out from The Singularity is Kurzweil's prediction that by 2025 personal computers (I think costing less than \$1,000 in today's money) will have the power of the human brain.

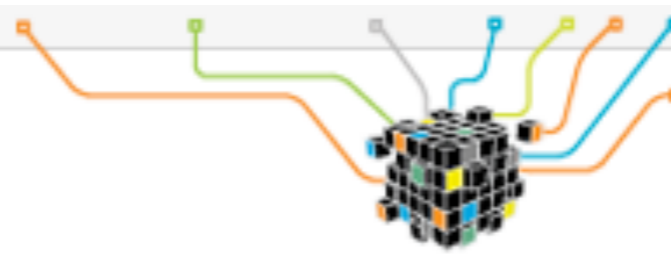
The first component to this prediction is an assessment of the computational power of the human brain. Kurzweil looks at a number of different ways to think about this question and they all yield estimates in the range of  $10(\text{exp } 14)$  to  $10(\text{exp } 15)$  calculations per second (henceforth cps) – that is *one hundred million trillion cps to one billion trillion cps*.

Today's personal computers, or at least those back in 2005 when The Singularity was published provided  $10(\text{exp } 9)$  cps and an extrapolation historical increases in computing power going forward yields the prediction that personal computers will have a capacity of  $10(\text{exp } 16)$  cps by 2025.

Kurzweil goes on to add substance to this prediction by discussing the technologies that will yield the increase in computing performance that he is predicting.







March 7, 2011, 3:56 PM

## Software Progress Beats Moore's Law

By STEVE LOHR

One of the old jokes in computing is that what the hardware giveth, the software taketh away.

The biblical vernacular is meant to convey the commandmentlike certainty of a set order of things. And the implication of the old saw is that chip-based progress is torrid and the engine of computing innovation, while the messy, unpredictable process of humans writing code is the laggard — the caboose of the innovation train.

In reporting [a Sunday Week in Review piece](#), I was pointed to research that sharply contradicts the conventional wisdom. It did not find its way into the more general article, but the research was intriguing, I thought, and its implication not widely appreciated.

[A report by an independent group of science and technology advisers to the White House](#), published last December, cited research showing that performance gains in doing computing tasks that result from improvements in software algorithms often far outpace the gains attributable to faster processors.

1,000 was attributable to faster processor speeds,  
43,000 was due to software algorithms

Martin Grottschel, a German scientist and mathematician. Yet a factor of

improvements in the efficiency of

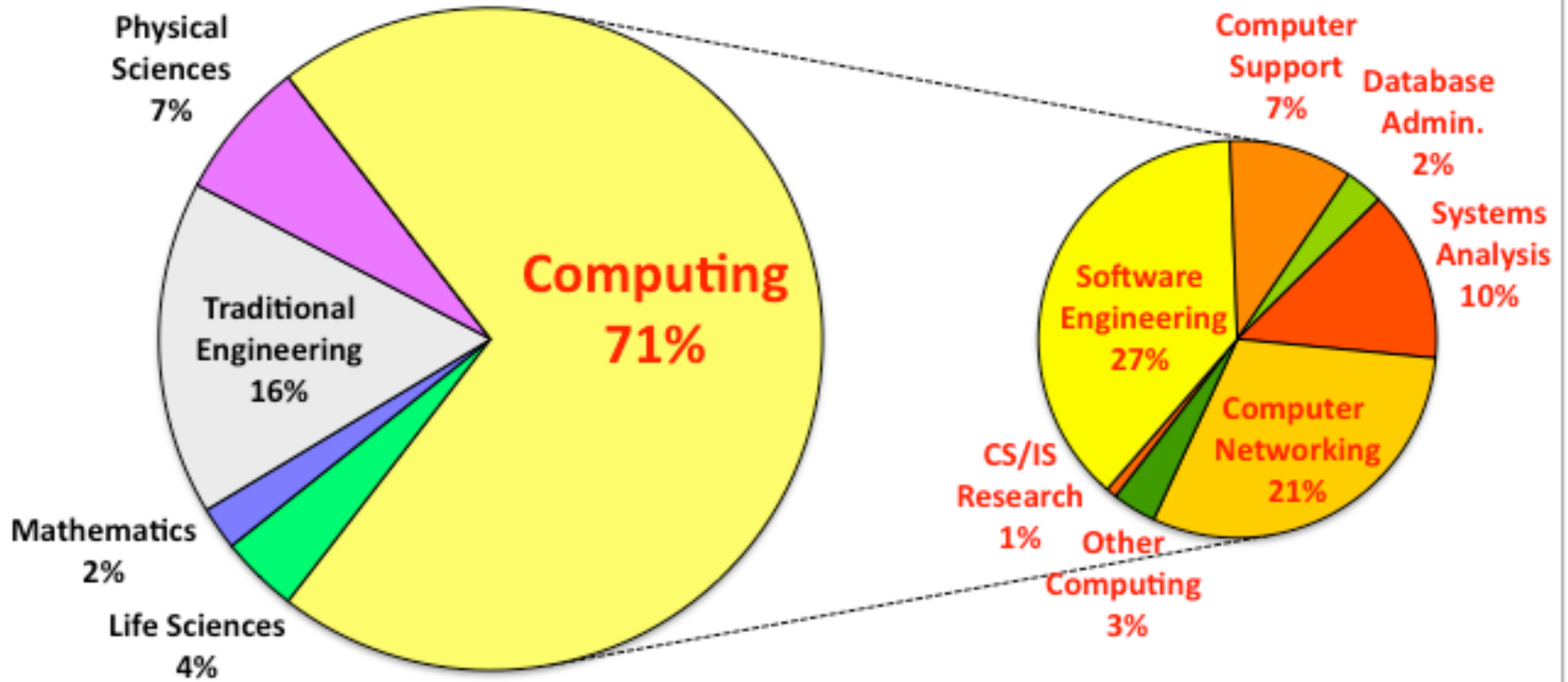


# The Revolution is Happening

- Be an *actor* or a *spectator*?



## Percentage Of New STEM Jobs By Area Through 2018



Data Source: US-BLS Employment Projections, 2008-2018 ([http://www.bls.gov/emp/ep\\_table\\_102.pdf](http://www.bls.gov/emp/ep_table_102.pdf)).

# Informatics *Literacy*

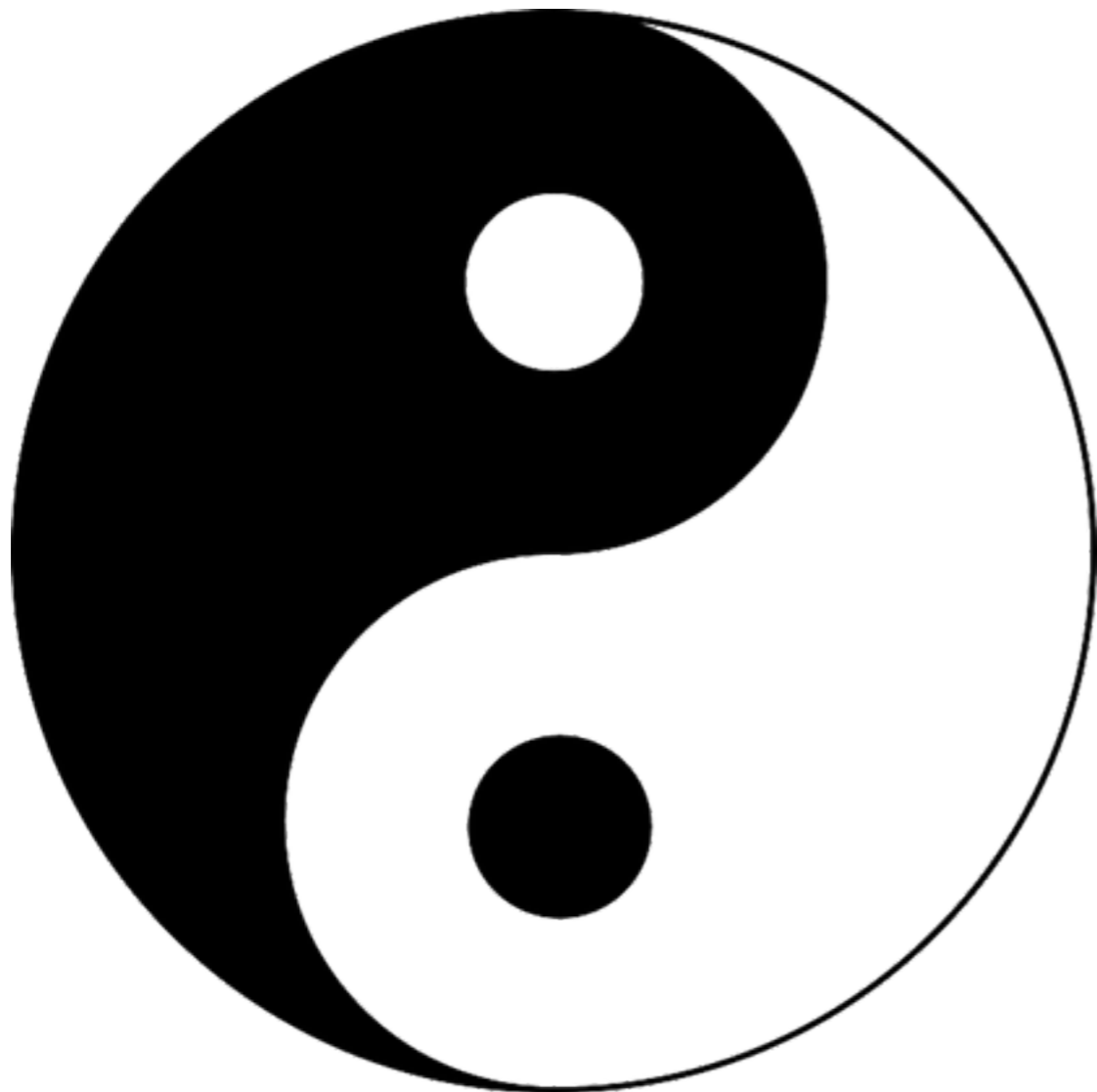
- how to recognize problems computers can solve
- how to decompose them into pieces
- how computers are organized
- how we map decomposed problems to computers



# Learning

- CS will challenge your ability to learn
- Machine Learning = Artificial Intelligence
  - greatest challenge in CS
  - already beats people in many tasks

# Induction - Deduction

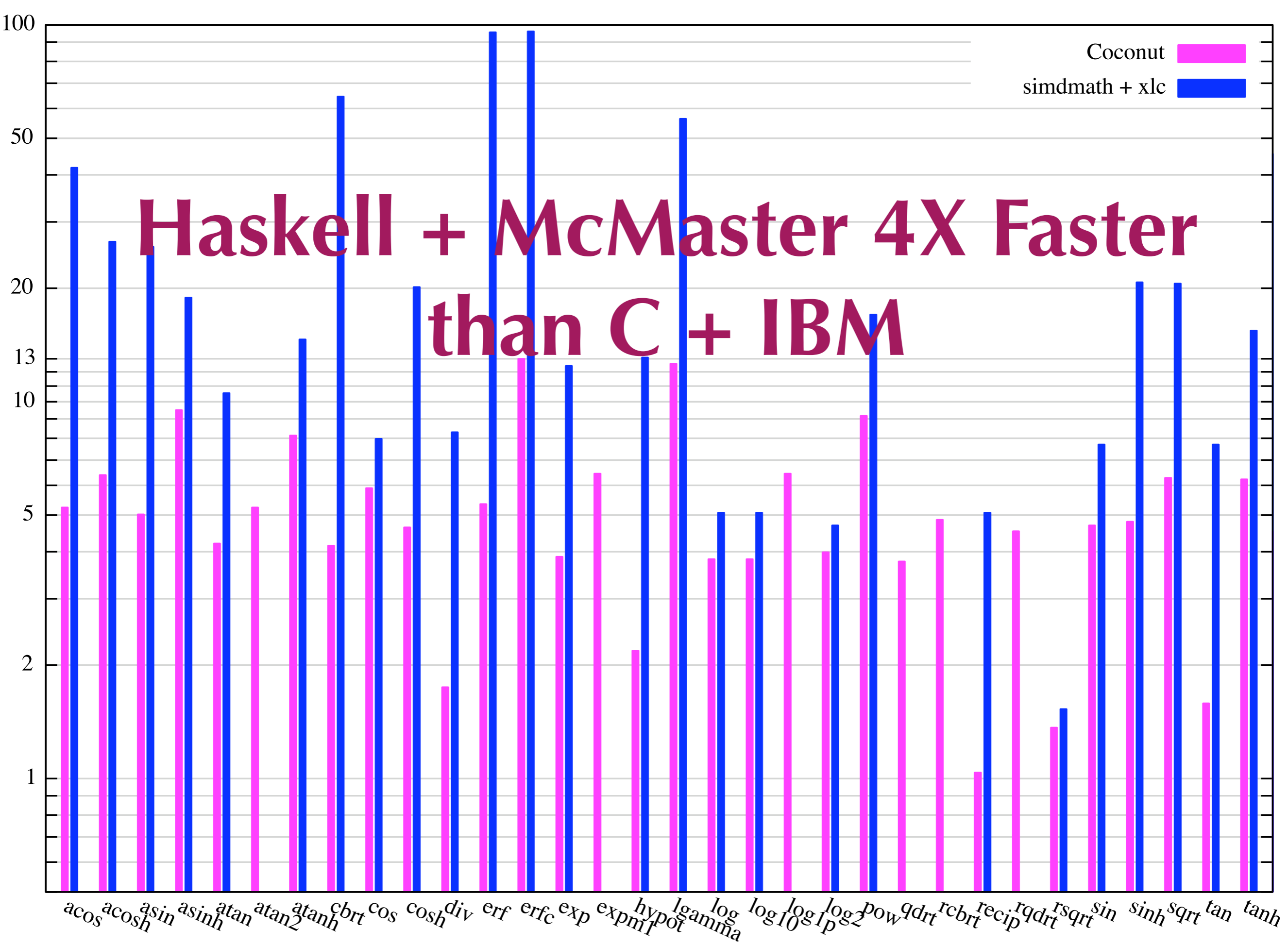


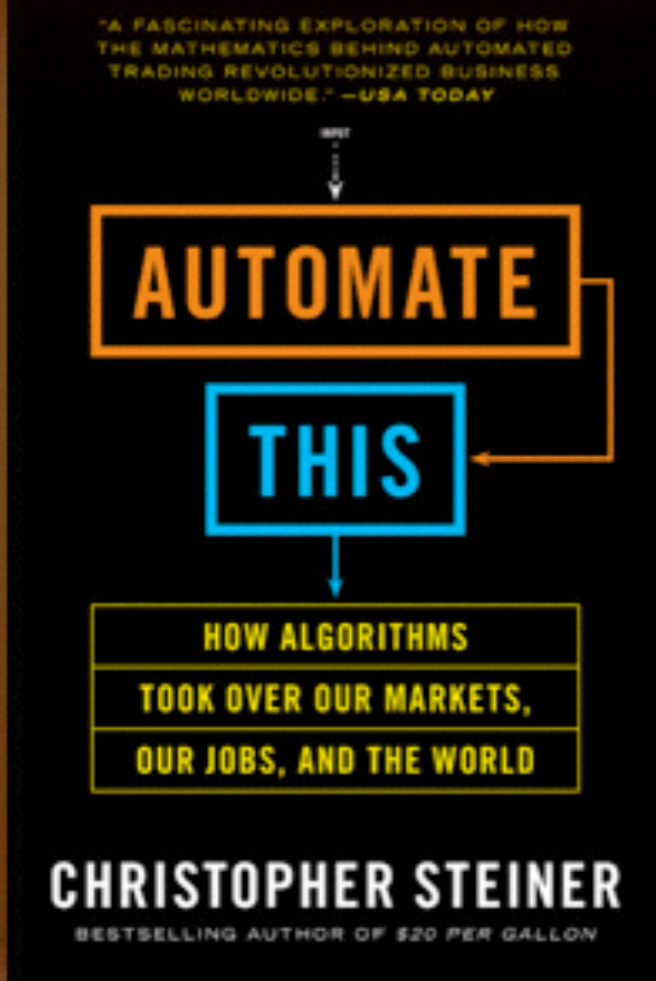
- virtuous circle
  - learn from examples
  - learn from trying
  - learn from organizing
  - learn from proving

# ELM - born 2012

- functional *reactive* programming for the web
- functional programming
  - *better* because
    - shortest distance from problem to solution
    - easy to apply logical thinking
  - *worse* because
    - needs better compiler
    - only well-educated people know about it
- other functional programming languages
  - Scala (big in big business computing, born 2003)
  - Swift (future of iOS and MacOS, born 2014)
  - Haskell (~~workhorse~~ racehorse of CS research, born 1990)







## Game Frame


Using Games as a Strategy for Success

Aaron Dignan >

Published: Mar 8, 2011  
Print Length: 224 Pages  
No Ratings

\$19.99    SAMPLE

Details    Ratings and Reviews    Rel



## Big Data

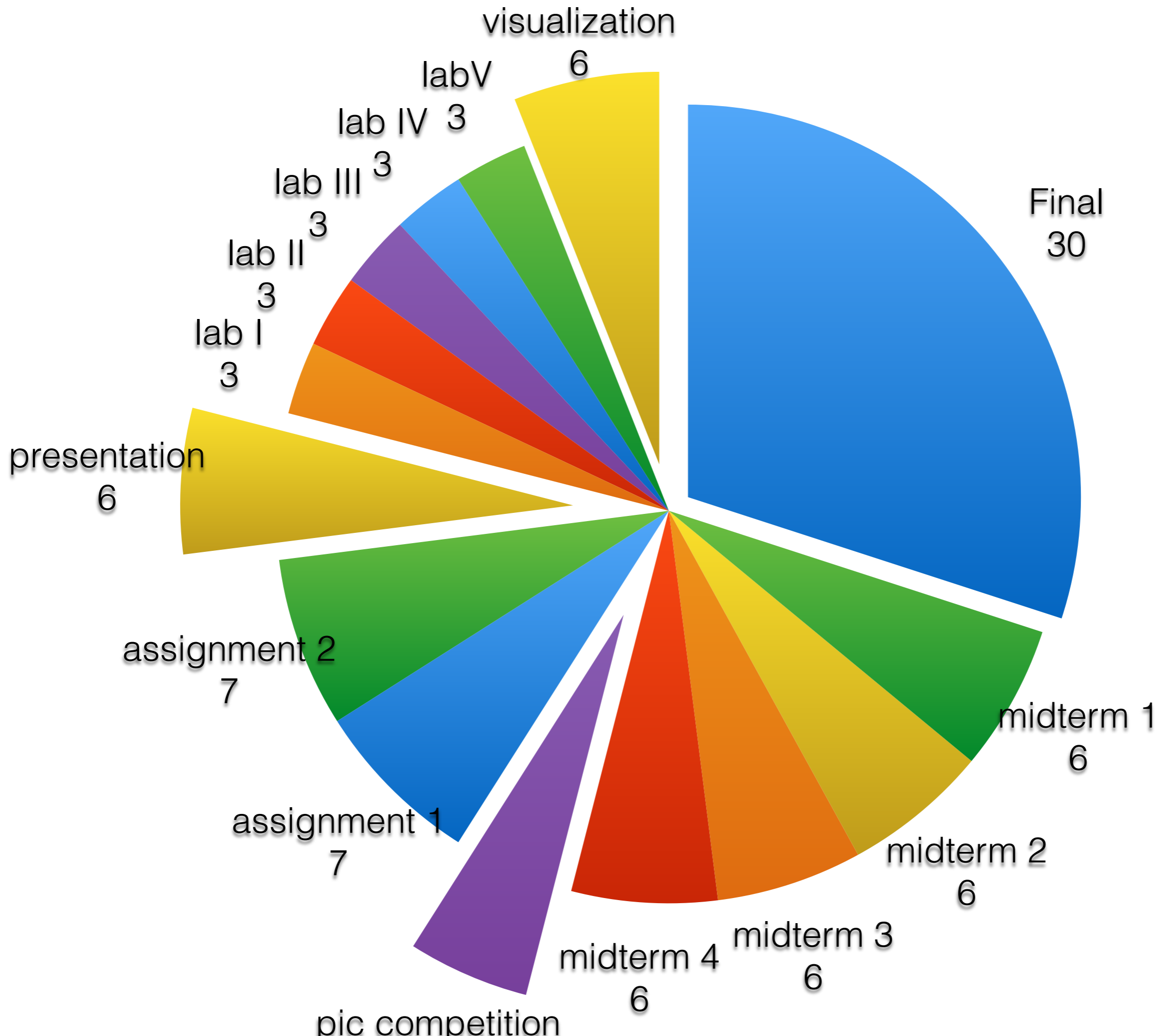
A Revolution That Will Transform How We Live, Work, and Think

Viktor Mayer-Schönberger & Kenneth Cukier >

Published: Mar 11, 2013  
Print Length: 240 Pages  
No Ratings

\$15.99    SAMPLE

11 Books to get you started!!!



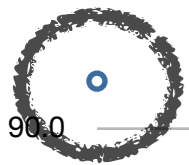


# Portfolio

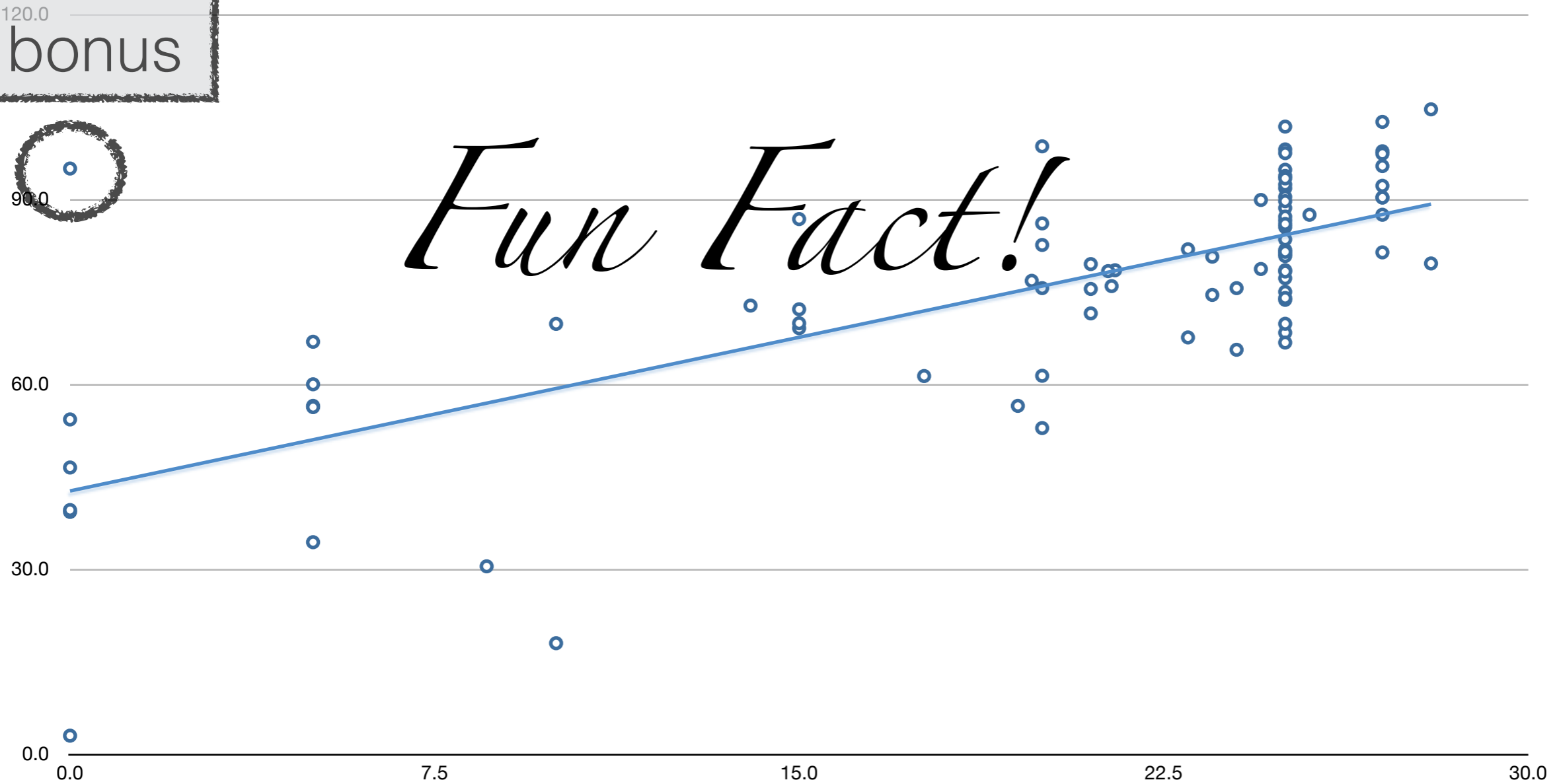
- your work
- shows what you can *do* with *knowledge*
- great for job interviews
- CS is now a portfolio program = *required*
- CS students need an advisor
  - ask me, especially if you want to help with Outreach  
— more about Outreach tomorrow

# Do Assignments Marks Go Up

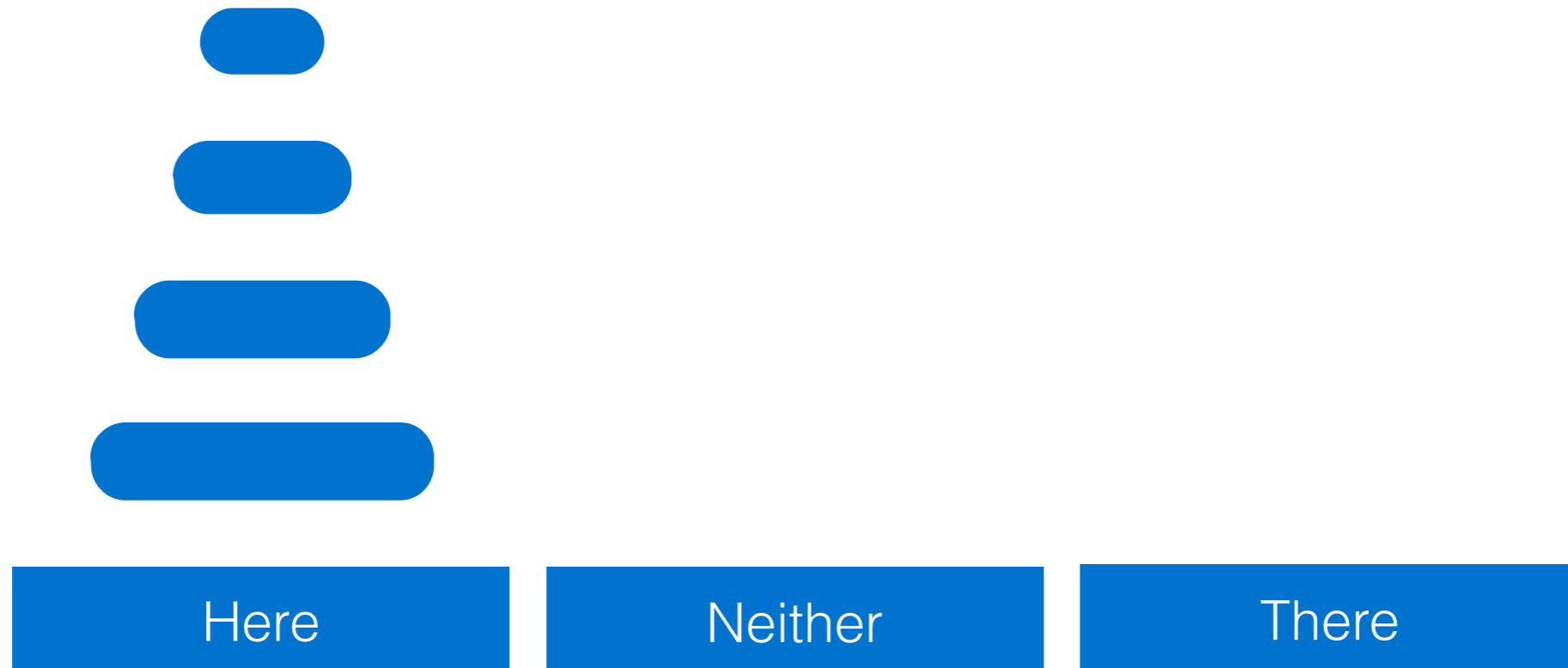
bored?  
ask for  
bonus



*Fun Fact!*



# Tower of Hanoi



Big under Small

One at a Time

from Here to There



# Homework

- Write out rules for getting from
  - Here
    - to
      - There