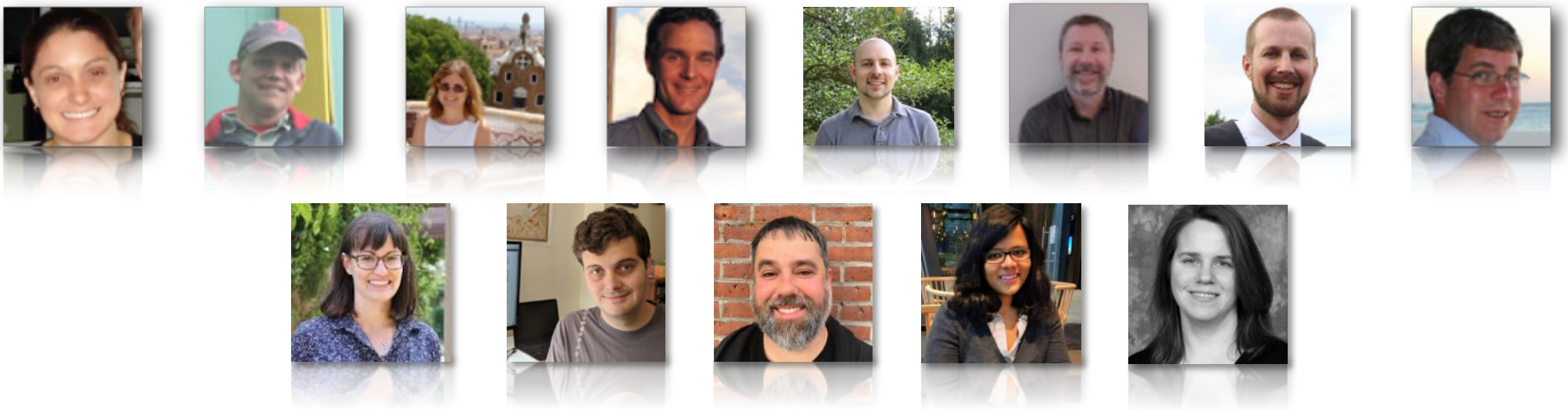
# Computer Science Faculty



- Diverse interests and background!
- Short "lightning" talks today
  - Should follow-up individually for more info!

# Summer Research in CS

- Spend 8-10 weeks working with individual faculty member on specific research problems
- Live in dorm with other students (not just CS students)
- Work in computer lab (mostly)
- "Flexible" schedule
- Social events (epic canoe trip)
- Chance to get to know us personally
- No classes = more time for faculty to focus on research

# Honors Thesis in CS

- Spend senior year working on independent research project
- Advised by faculty member
- Fall is spent reading
  - Become an expert in chosen area
- WSP is spent working…hard
- Spring is spent writing
  - Gathering results
  - Writing thesis
- Present results in May in public presentation
- Lots of work, but very rewarding
- Highly recommended for those considering grad school

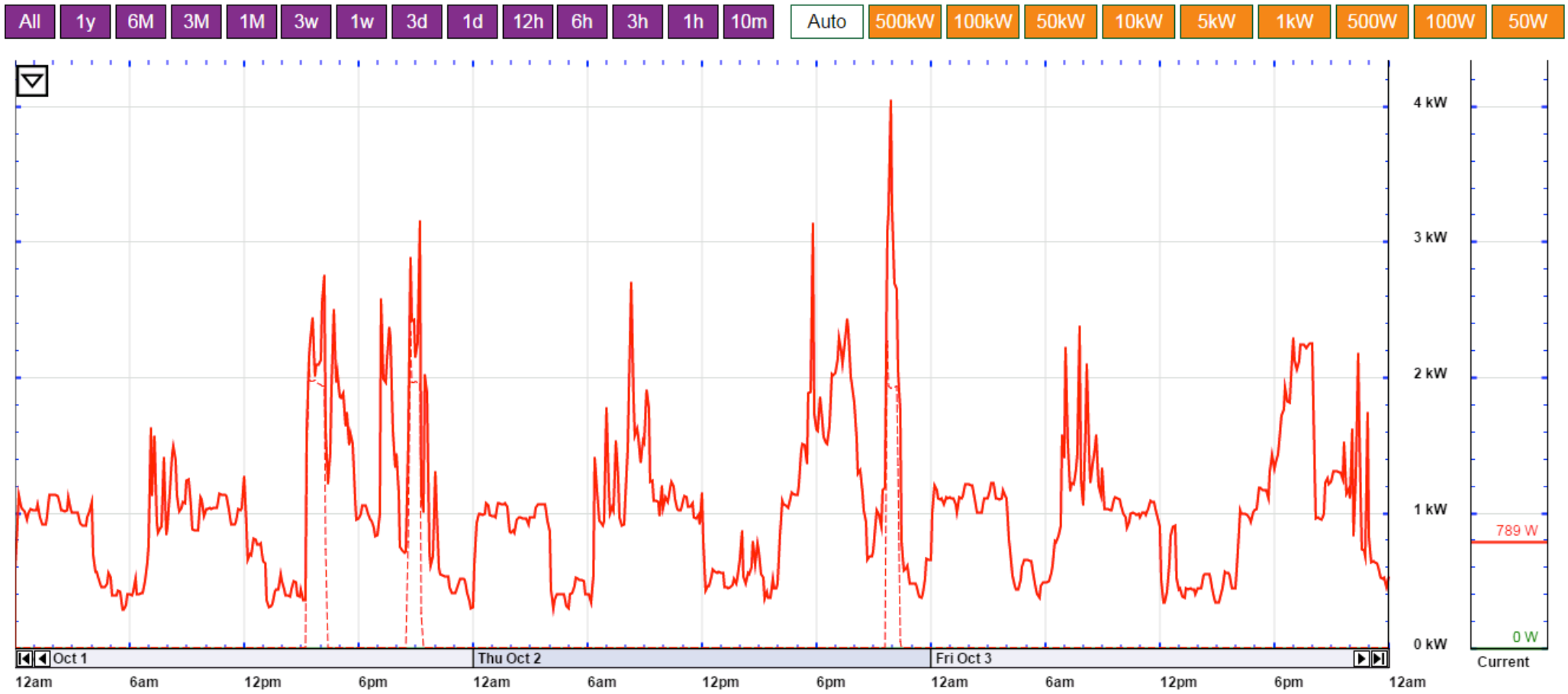
# Faculty research interests

# Jeannie Albrecht

- Jeannie works in Distributed System, Networks, IoT
- Lately, Smart Buildings
- Also K-8 CS ed



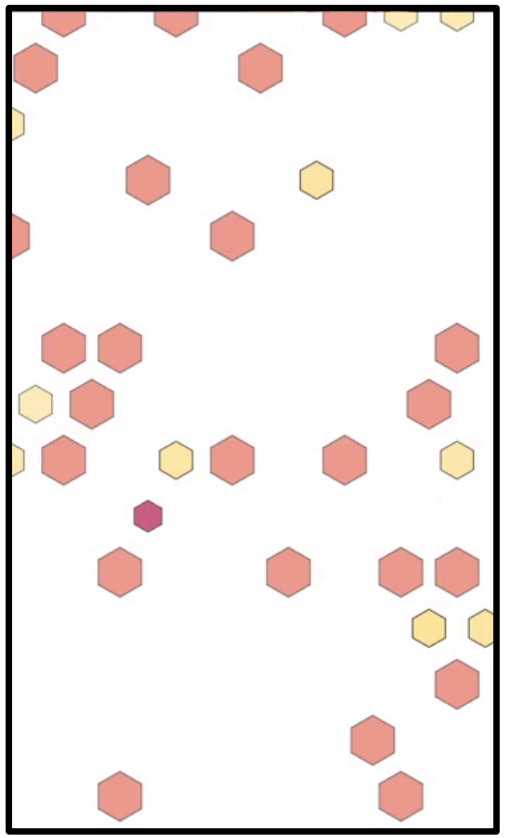

# Traditional Energy Use Visualizations

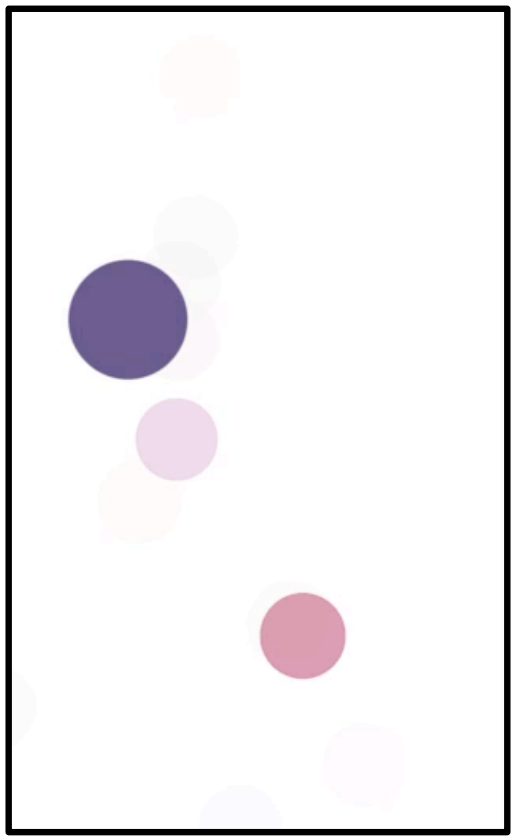


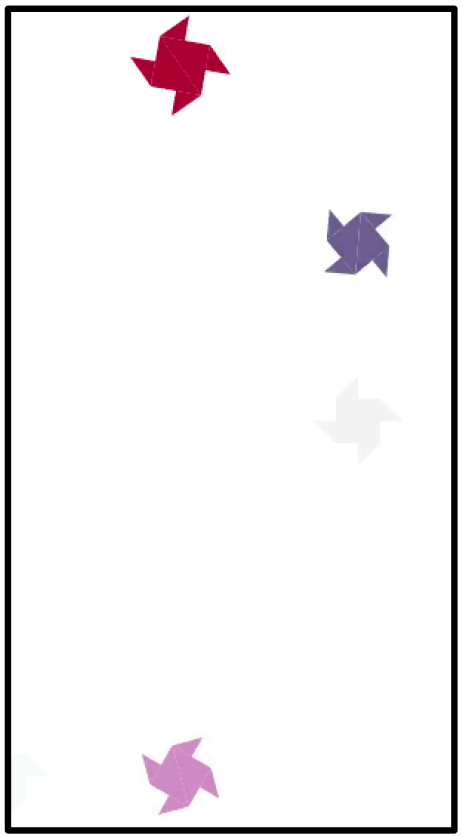

- Need something that is more intuitive…

# Research Goal:
# Make Buildings *and* Occupants Smart(er)

- Use *ambient* visualizations to help occupants better understand energy usage (in kitchen)
- Research students need to have at least taken CS 136







# Kitchen Simulator

# Duane Bailey

- Duane is a computer architect
- Works with hardware
- Often advises projects that use FPGAs
- FPGAs contain programmable logic blocks and interconnection circuits
- They can be programmed/ reprogrammed to the required functionality after manufacturing
- (kinda like custom built hardware)
- Need at least CSCI 237

# Dan Barowy: Spreadsheet Support for End-to-End Data Analysis

| data wrangling | data cleaning | debugging |
|---|---|---|
| FlashRelate | CheckCell | ExceLint |
| PLDI ‘15 | OOPSLA ‘14 | OOPSLA ‘18 |

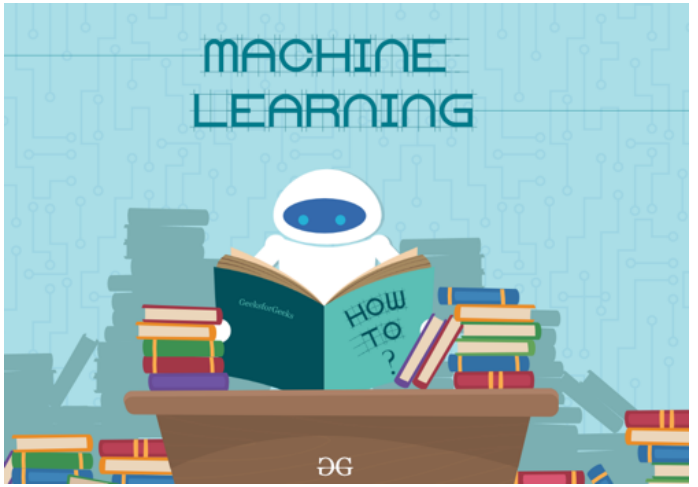
Background req’d: **136**!!! **334** or **376** preferred, but not required.

Daniel Barowy, Assistant Professor

Williams

# Andrea Danyluk

- Andrea works in Machine Learning
- Projects involve "Supervised Inductive Machine Learning"
- Examples:
  - Learn a predictive model from labeled training data
    - Will this cancer patient experience a recurrence after initial treatment? Is this banknote authentic or not? Who is the author of this Amazon review?
  - Recent excitement
    - Availability of data. Increased processing power.
- Interesting research problems
  - New algorithms for new types of problems. Model interpretability for humans. Models that balance accuracy and cost. And many more.




https://www.geeksforgeeks.org/machine-learning/

# Supervised Inductive Machine Learning: Summer Research at Williams with Andrea

Application-Driven

Model expressivity

Identify individuals


Identify keys to successful teaching and learning

What it's like:

- Learn application domain
- Learn ML algorithms and empirical methodology
- Gather, clean, and featurize data (at least CSCI 136)
- Run experiments
- Have fun!

# Steve Freund

- Steve works in Programming Languages
- Probably need to have taken at least 136 and probably 334

# Google Docs

2 Days Ago

UPDATE

* Bug fixes and performance improvements


# Amazon Prime Video

Jan 29, 2018

UPDATE

Bug fixes and performance improvements

# Google Sheets

2 Days Ago

UPDATE

* Bug fixes and performance improvements

Netflix

3 Days Ago

UPDATE

Bug fixes


# HBO GO: Stream with TV Package

3 Days Ago

UPDATE

- Bug fixes and performance improvements.

Instagram

Jan 29, 2018

UPDATE

General bug fixes and performance improvements


## Google Calendar

Today

UPDATE

- Bug fixes and performance improvements.

# Traditional Software Process

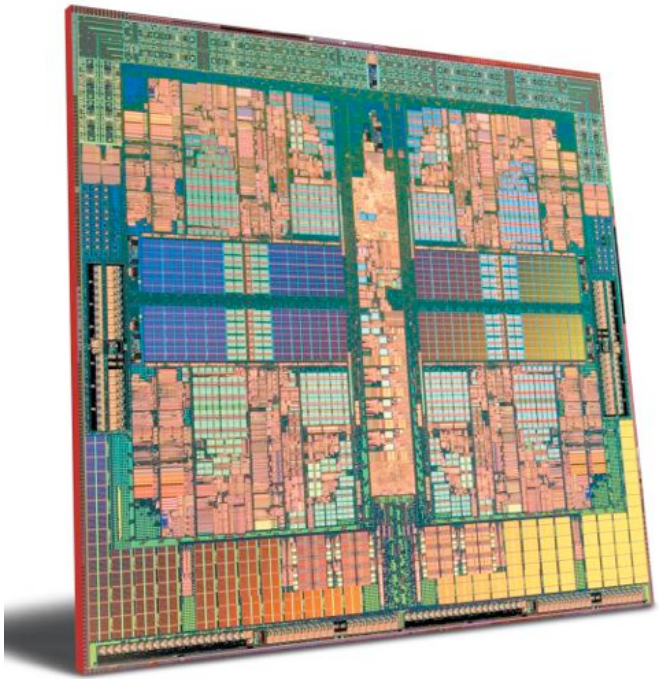
# With Better Programming Tools


Better:

- Programming Languages
- Verification Techniques
- Debuggers
- Programming Environments
- …

# Work on Programming Languages And Analysis Tools

- language design
- theoretical foundations
- proving theorems
- systems development
- performance modeling
- experimental validation



Hi folks! My name's Iris Howley.
I'm currently at the AAAI Conference on Artificial Intelligence, Ethics and Society with a couple of my Williams research students.

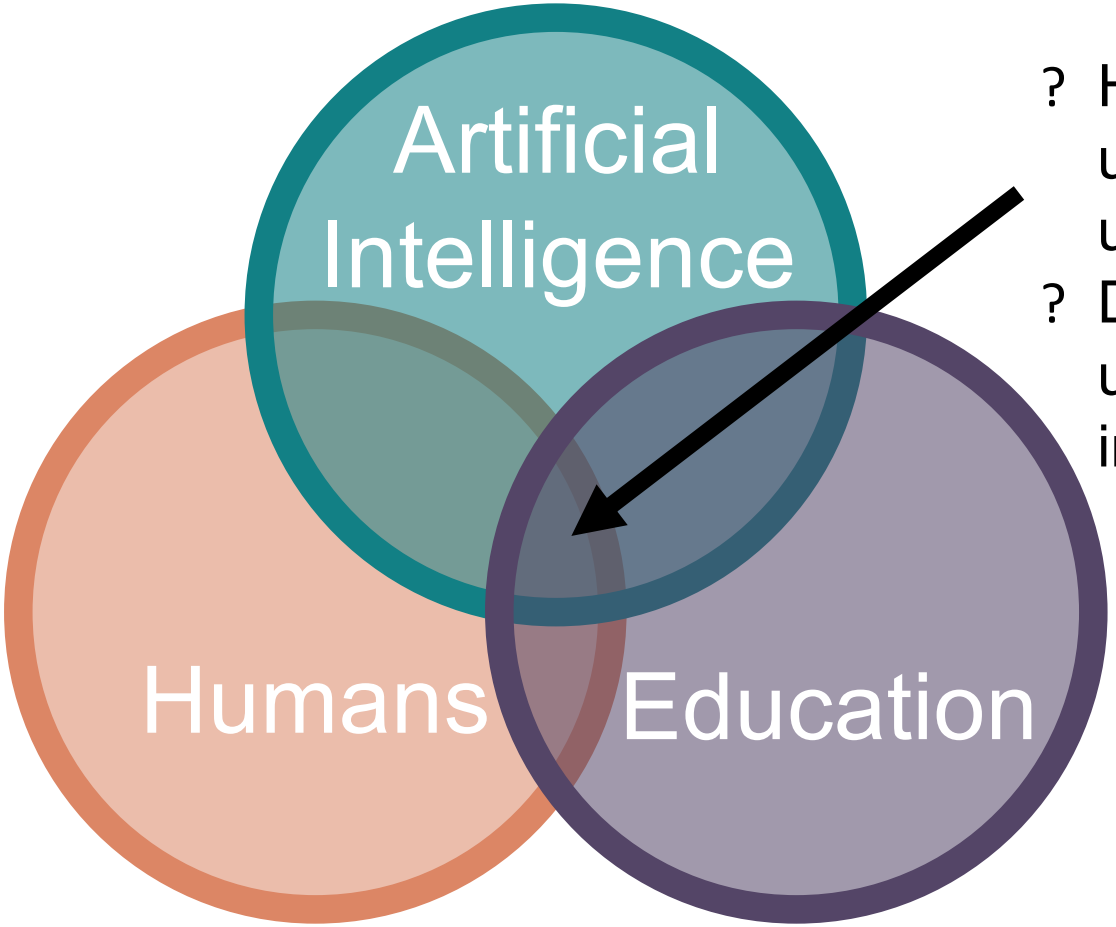
There's more info on doing research with me on **my website** under "Info for Students" so check that out & contact me to talk if you're interested.

MIT Tech Review recently wrote up some related work on my topic, here:

www.bit.ly/AIexplain

Iris Howley, iris@cs.williams.edu
Assistant Professor of Computer Science

**Iris works at the intersection of Human-Computer Interaction, Artificial Intelligence, and Education. Here are some questions she thinks about!**



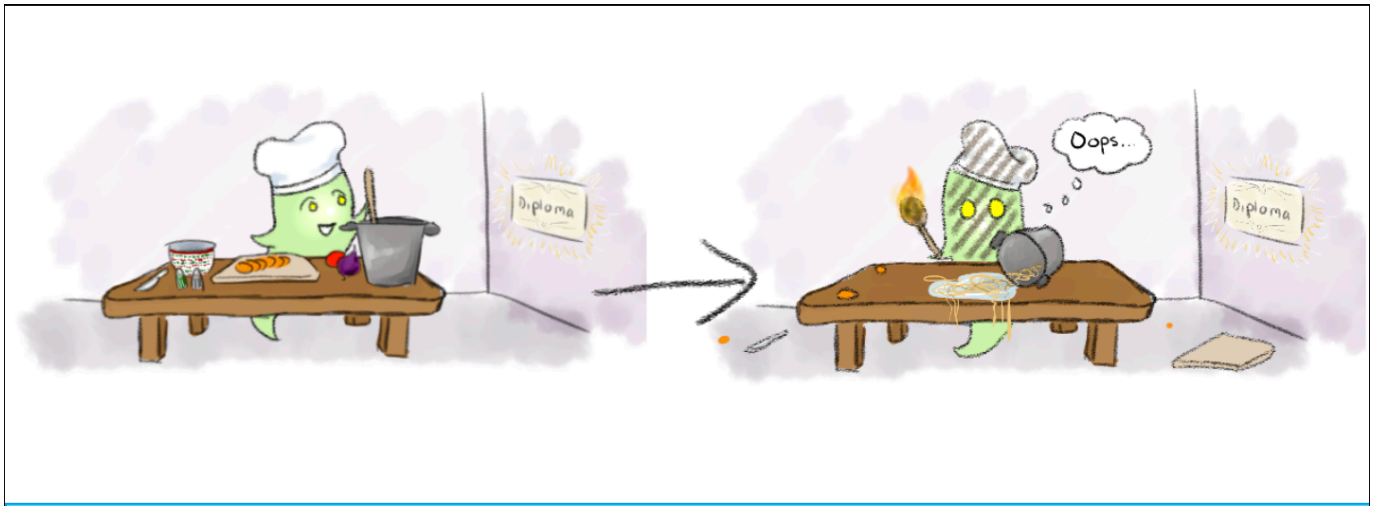

? How do users of ed tech systems understand the algorithms underlying their systems?

? Does increased algorithmic understanding lead to more informed decision-making?

**Skills We Learn in this Research**

- *Understand humans*: Interviews, User studies
- *Build systems to explain AI*: Javascript, web dev, computer languages
- *Evaluate*: Survey measures, behavioral measures

# Iterative User-Centered Design



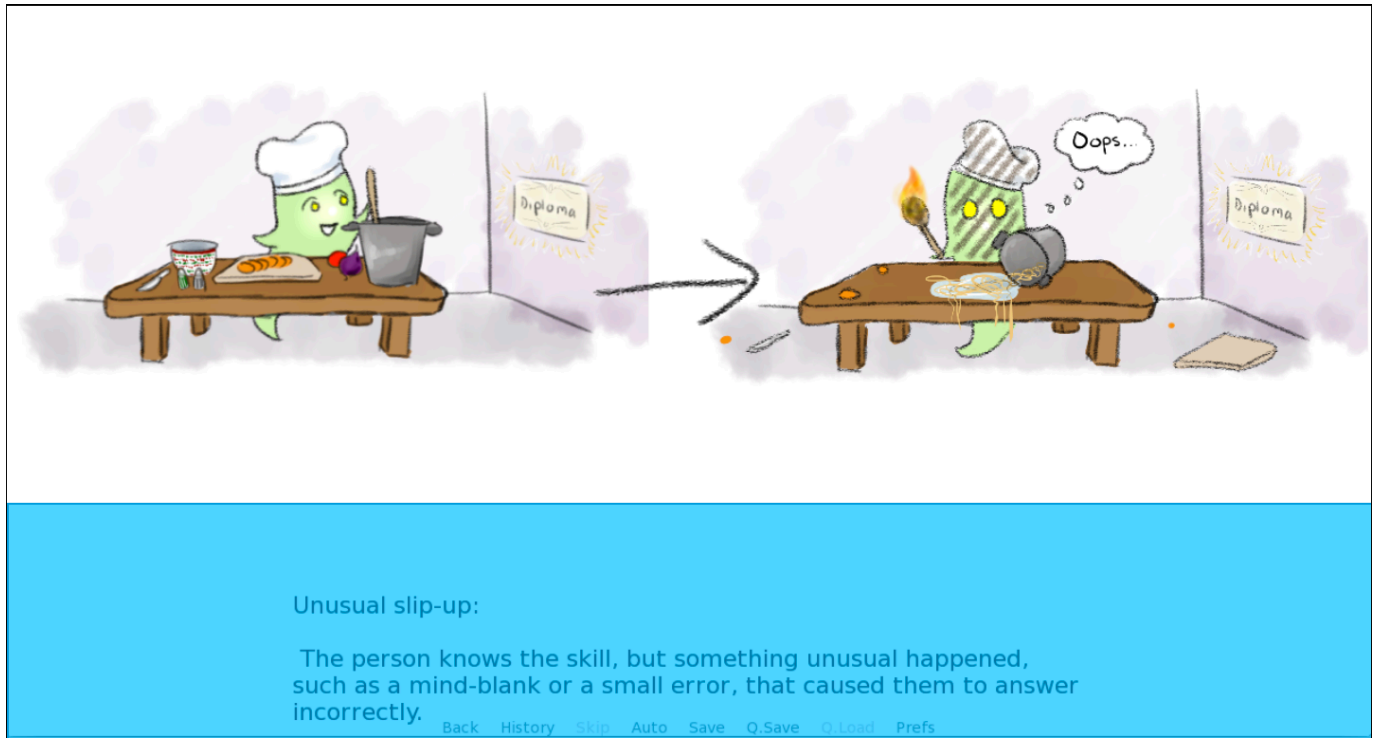

Unusual slip-up:

The person knows the skill, but something unusual happened, such as a mind-blank or a small error, that caused them to answer incorrectly.

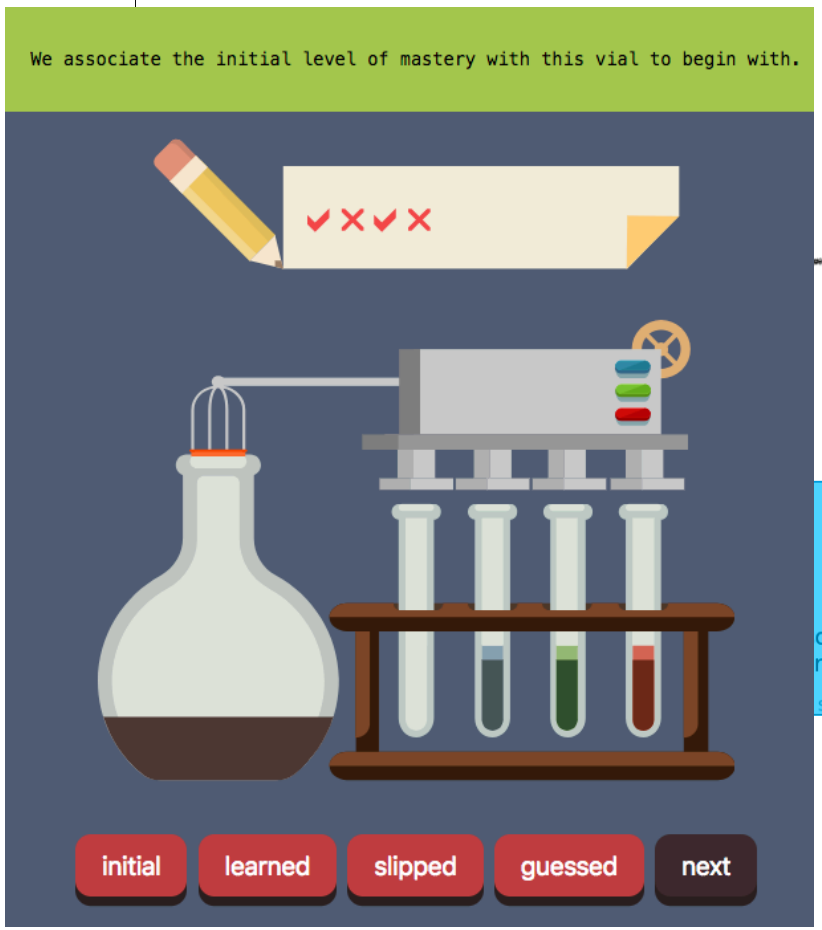
Back History Skip Auto Save Q.Save Q.Load Prefs

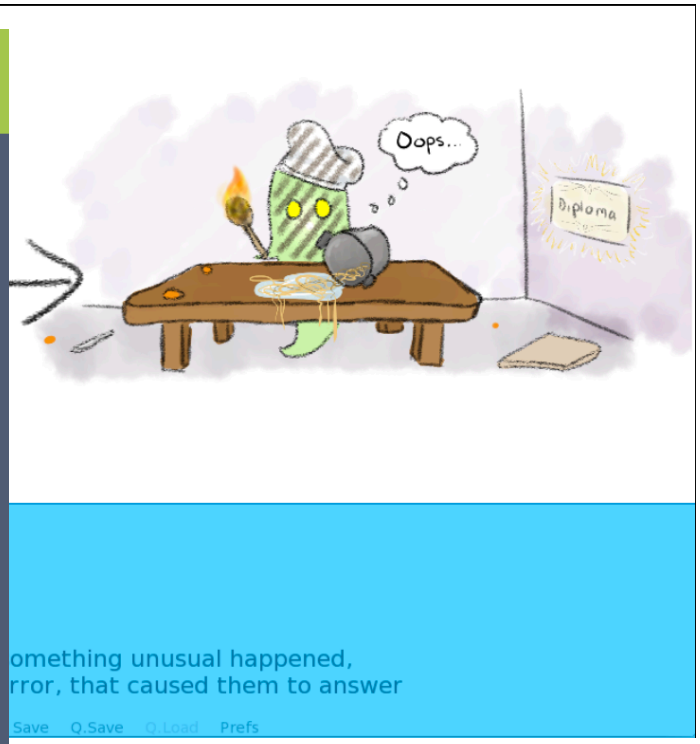
And now for some example of past summer research student work from my group!

# Iterative User-Centered Design

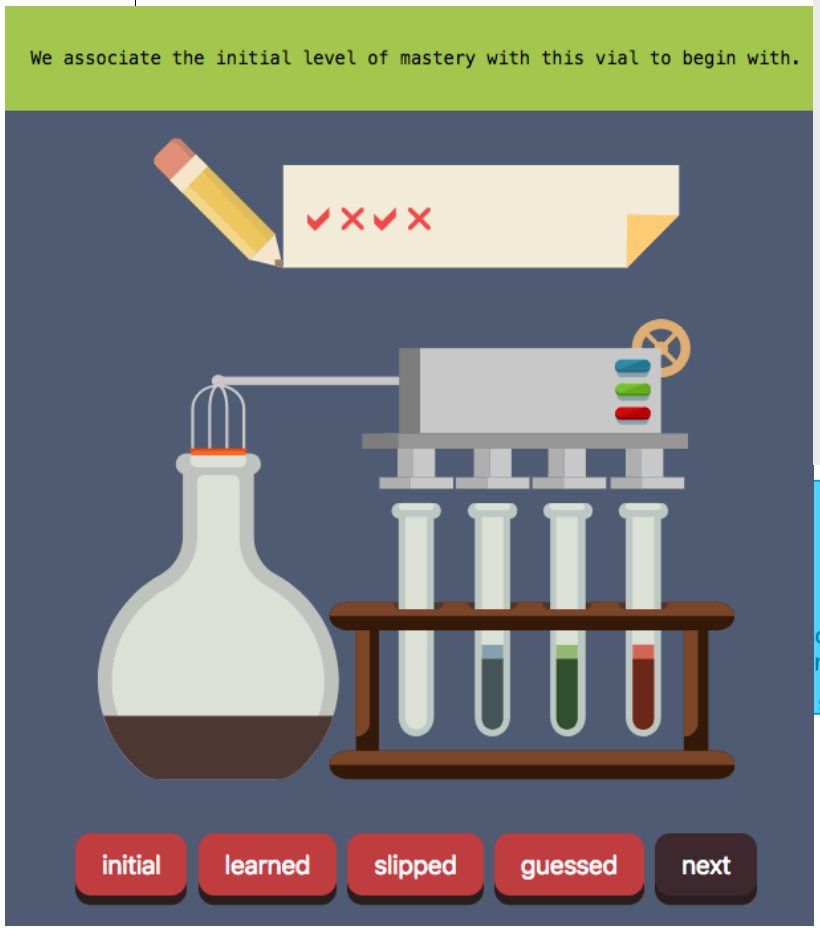


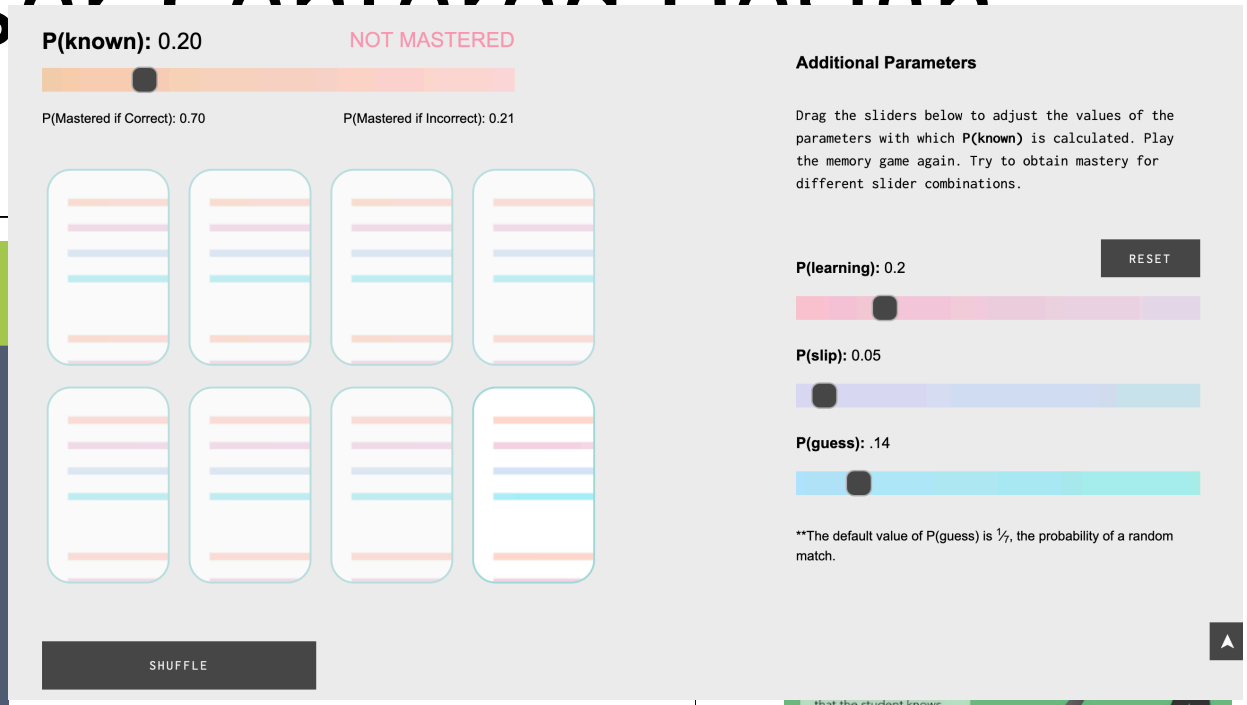

# Iterative User-Centered Design

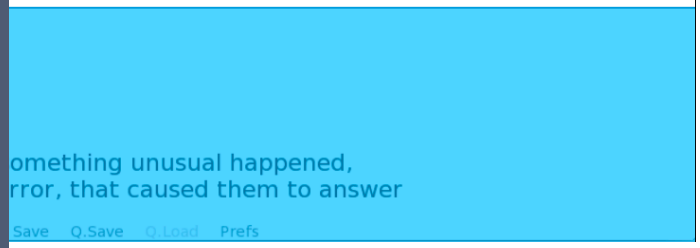


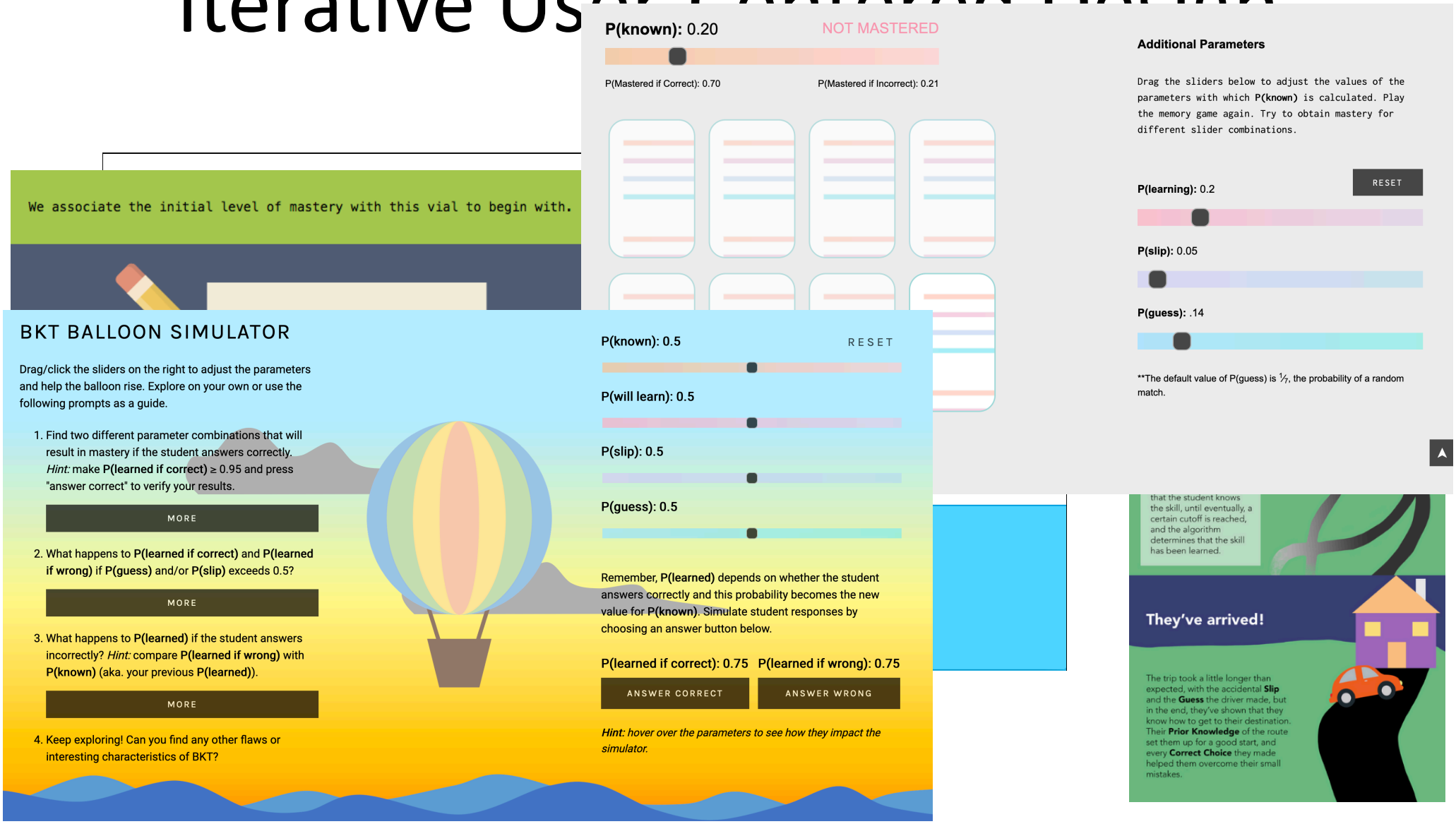

# Iterative User Centered Design



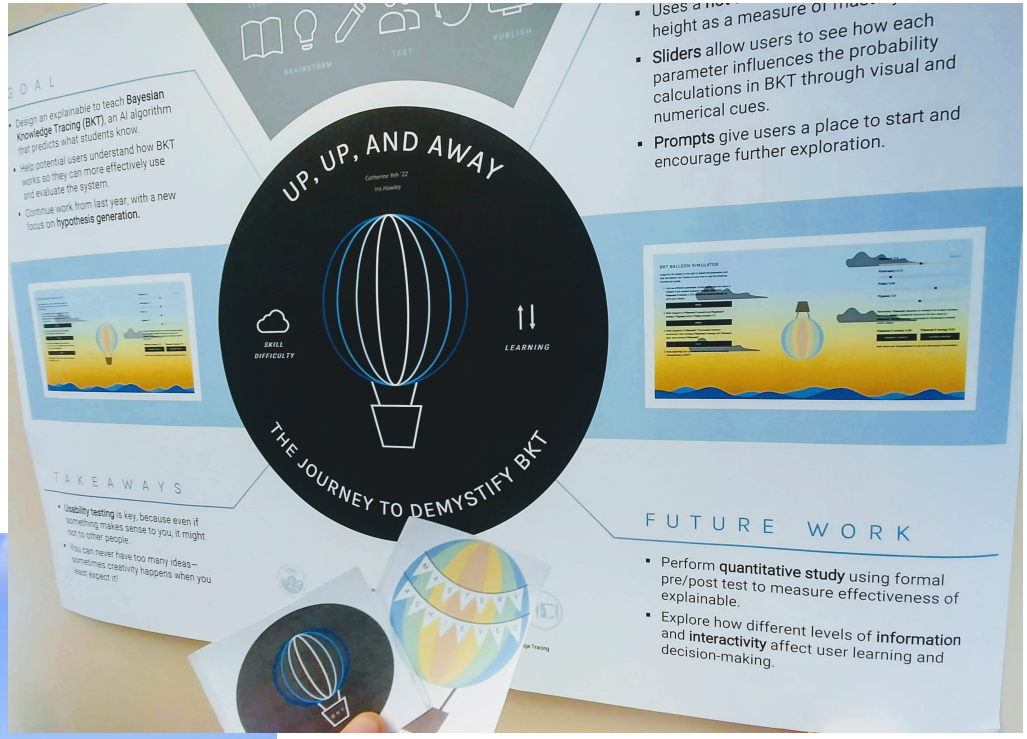




# Iterative User Centered Design



# Fun

Stickers coordinating with our research posters





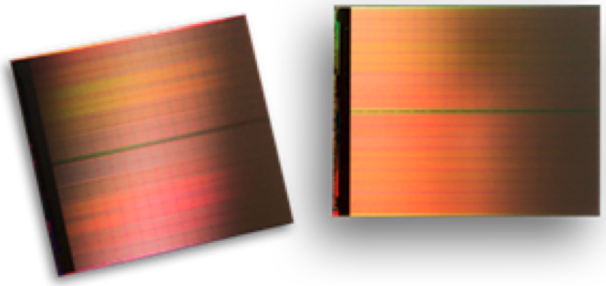
Making stickers to match our research posters!

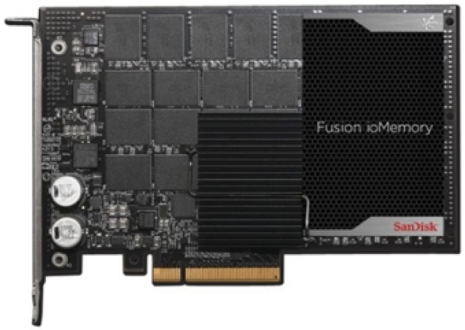
# Bill Jannen

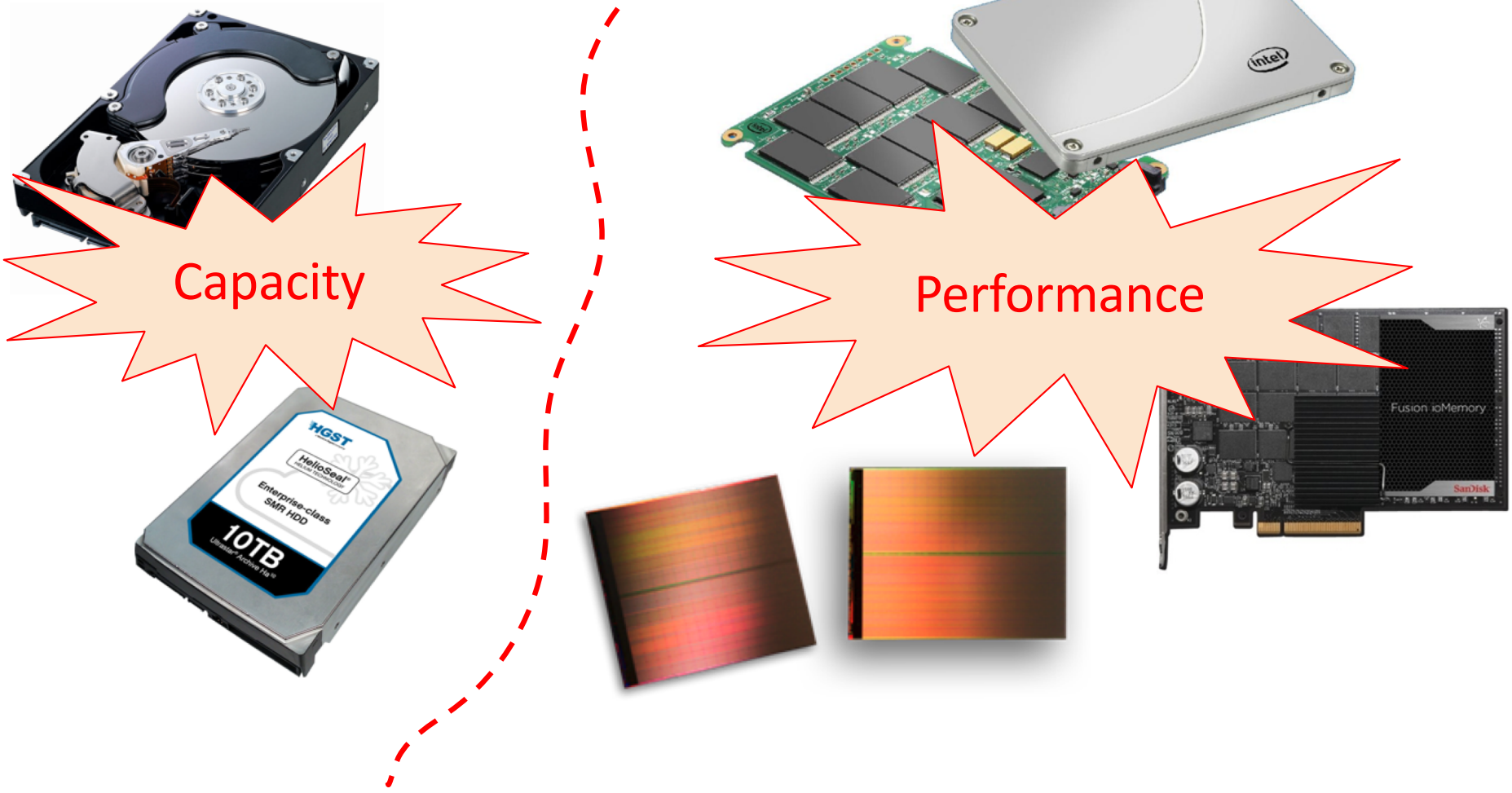

Bill works in Storage Systems

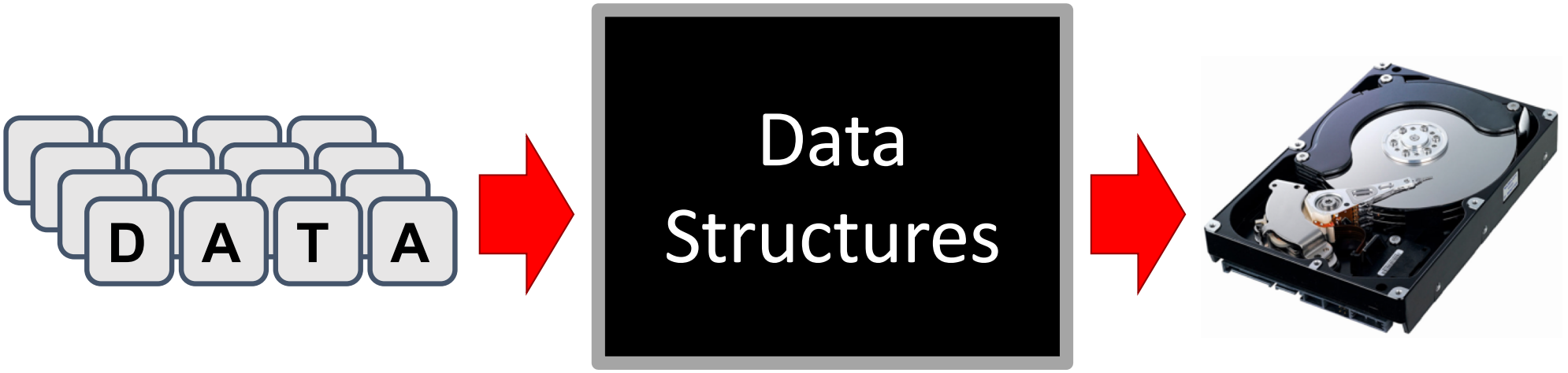

intel

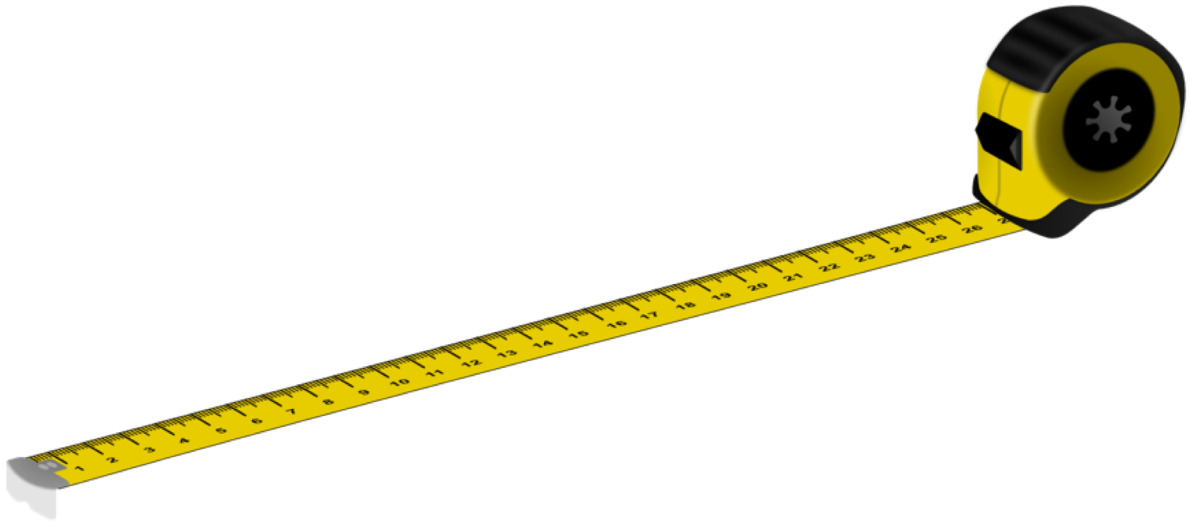

HGST
HelioSeal
Enterprise-class
SMR HDD
10TB




Fusion ioMemory
SanDisk

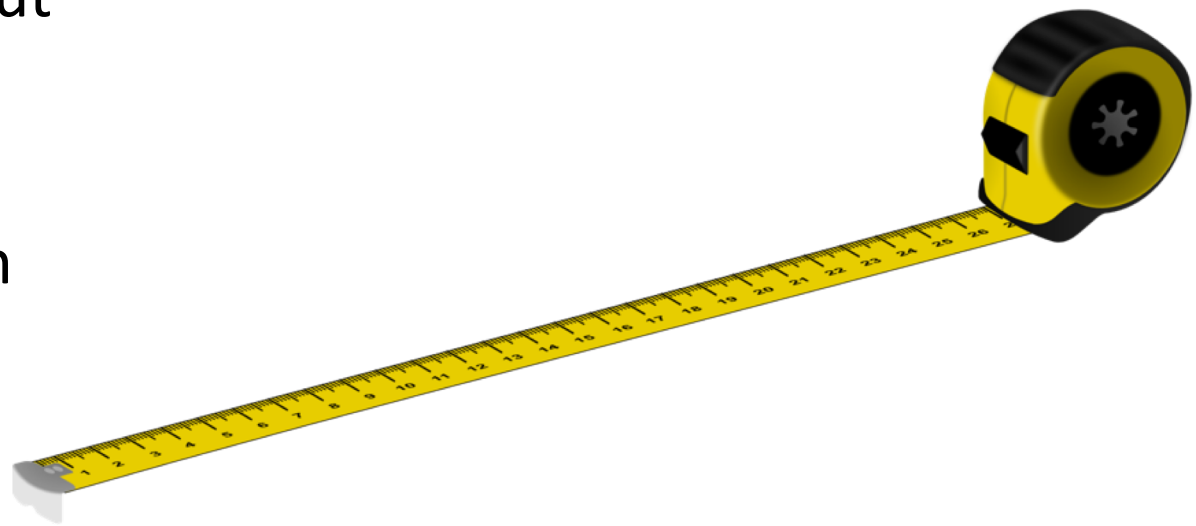

Capacity
Performance
HGST
HelioSeal
Enterprise-class
SMR HDD
10TB
intel
Fusion ioMemory
SanDisk


D A T A
Data
Structures



Write
Amplification

Throughput



Compression
Ratio

Utilization

Latency

IOPS

# Storage Research Goals

Bill tries to build systems that help users:

- Read and write their data faster (performance)
- Store more data in the same physical device (capacity)
- Ensure that data is never lost (safety)

He's interested in topics like:

- New HDD and SSD technologies
- File systems
- Key-value stores

- Data deduplication
- Write optimized data structures
- Filters

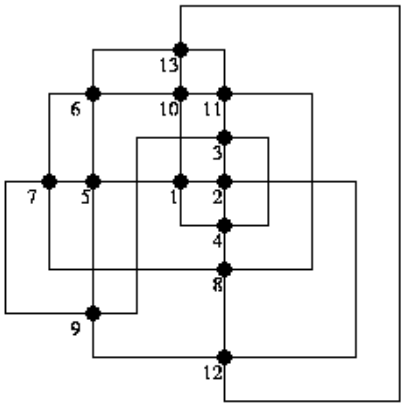
His research group has a website with their project info: `www.betrfs.org`

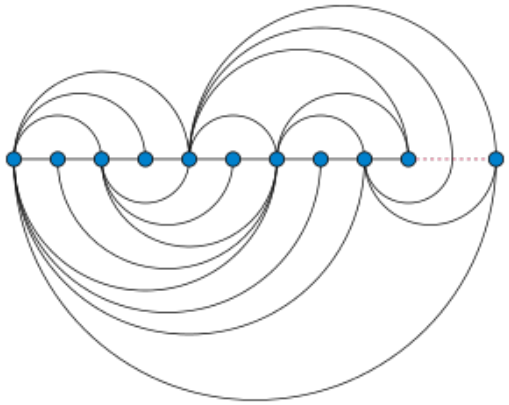
Probably need CSCI 136 and 237 (ideally)

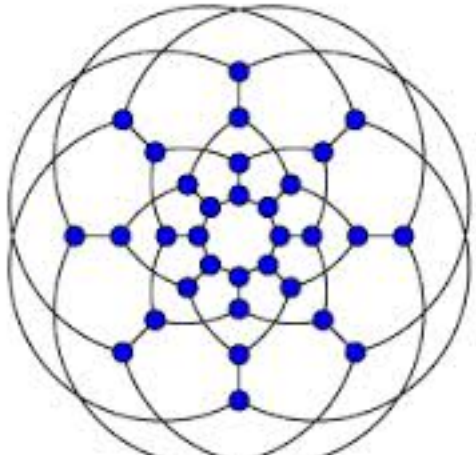
# Bill Lenhart

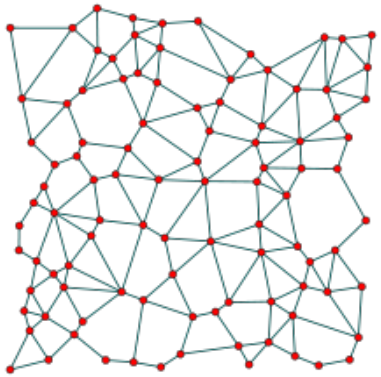
*How Bill Spends His Summer Vacations*

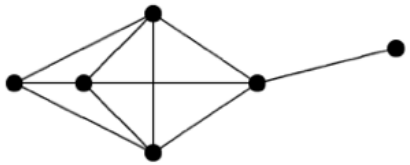
# *How To Draw a Graph*

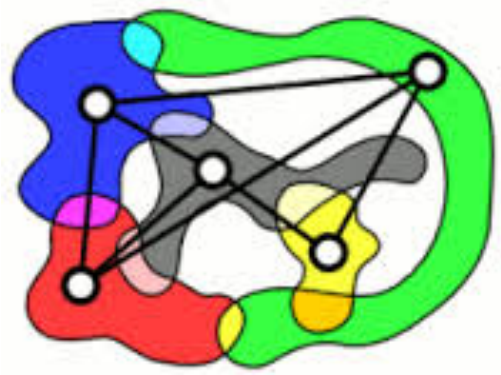


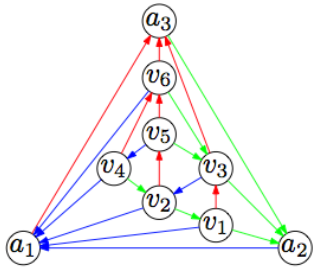



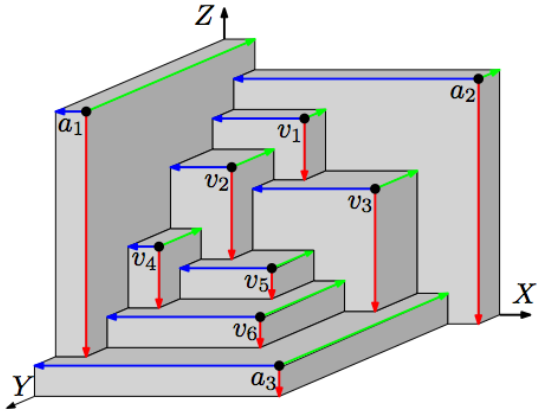




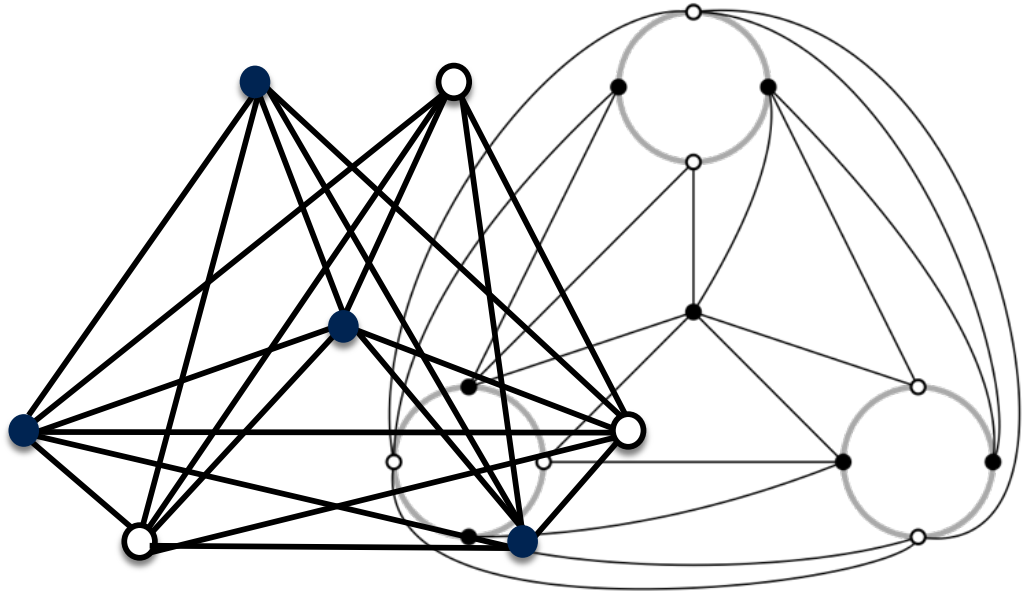






# A (2,2)-Planar Drawing

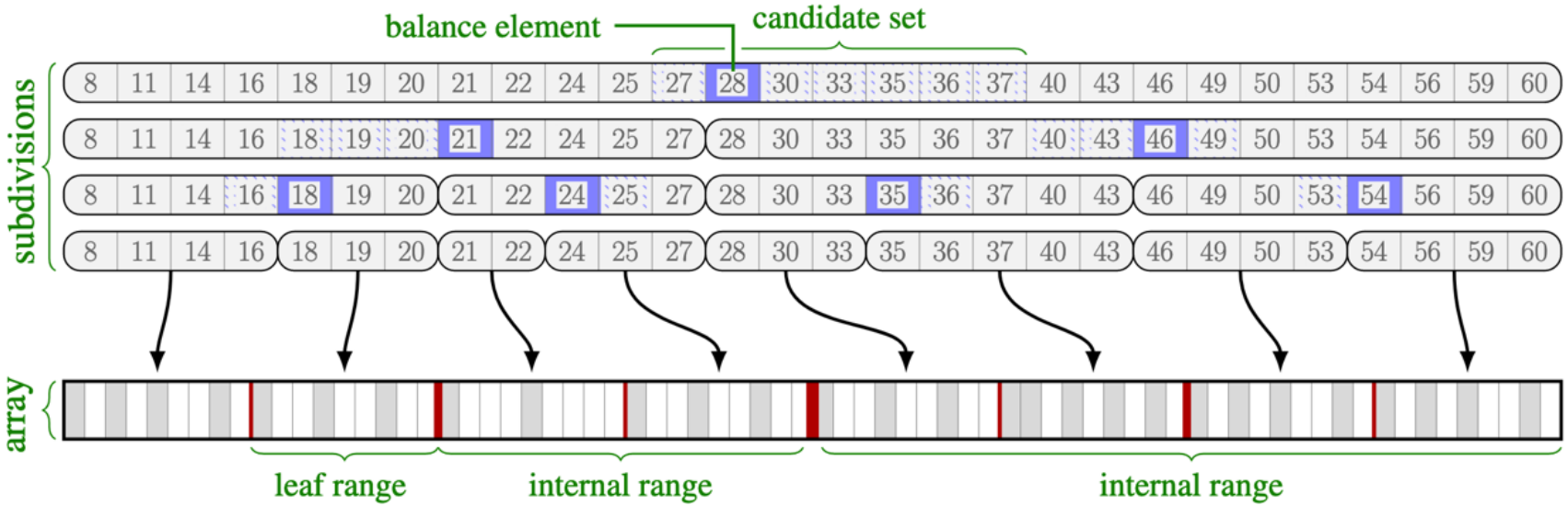


# The Game

- Identify interesting/useful/important class **C** of graphs
- Specify drawing model **M**
- Ask questions
  - Can G $\in$ **C** be drawn according to model **M**?
  - Can the drawing be computed efficiently?
  - Can the class $\mathbf{C_M}$ be easily characterized?
- Need CSCI 256 and strong background in math

# Sam McCauley

- Sam works on Data Structures
- How do we store large amounts of data so that we can quickly and efficiently access it?

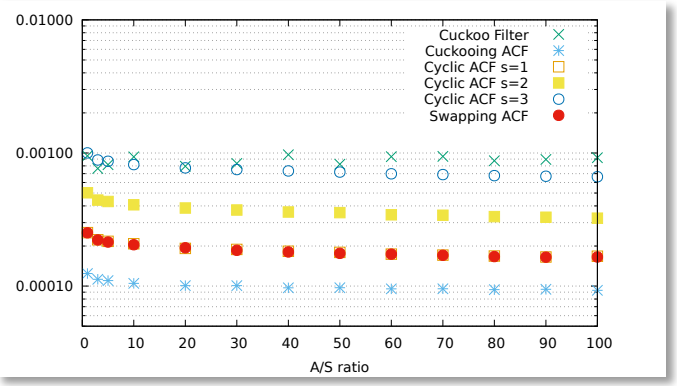


# Data Structures (Sam McCauley's research)

- Main principle: use randomness (and some math) to make these data structures simpler and more efficient
- Combines coding skills with worst-case analysis

$$\mathbb{E}\left[\sum_{\ell=1}^{k_i} |\{q \in Q_0 \mid h_\beta(q) = h_{\beta'}(x_{i_\ell}) \text{ for some } \beta, \beta' \in \{0,1\}\}| \;\middle|\; q_i \text{ is costly}\right] \leq 2 + 2k_i|Q|(\epsilon/n)$$

Using probability (like this) can lead to improved performance!



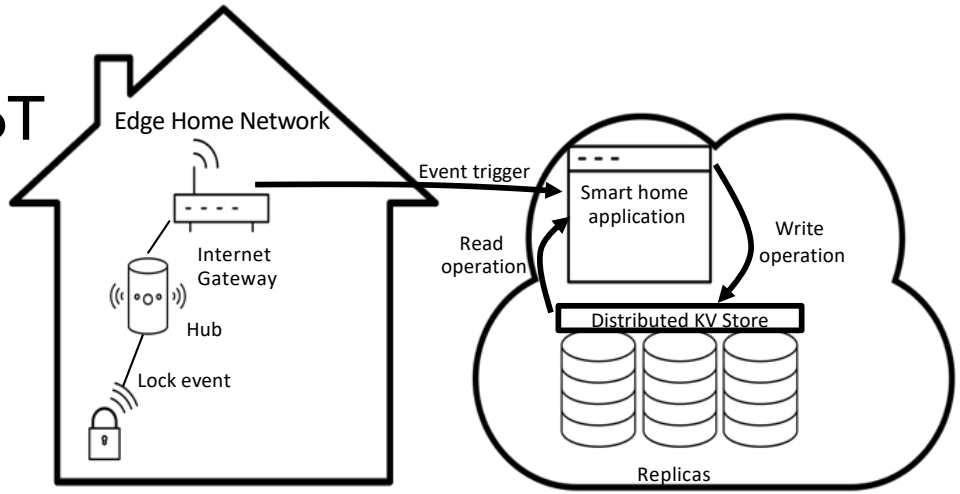
# What's needed?

- Flexible—both coding and math skills are useful
- But, you'll probably want to have taken Algorithms (CSCI 256)

- Projects may entail:
  - Designing new data structures
  - Implementing data structures that work in theory (how well do they do in practice?)
  - Looking at what use cases work best

# Kelly Shaw

- Kelly works on Parallel Architectures/IoT
- Interest in systems running concurrent applications
- Evaluation of existing systems
  - Analysis of application needs
  - Analysis of application/platform correctness
- System/tool creation
  - Static analysis tools examining correctness of applications
    - Amazon AWS IoT, Samsung SmartThings, Google IoT
  - Modifying existing systems to try different solutions related to concurrency
    - Mozilla WebThings – gateway for smart home devices
    - Cassandra – distributed database
- Prerequisite knowledge: CS237



Name. **Shikha Singh**

Research Area.

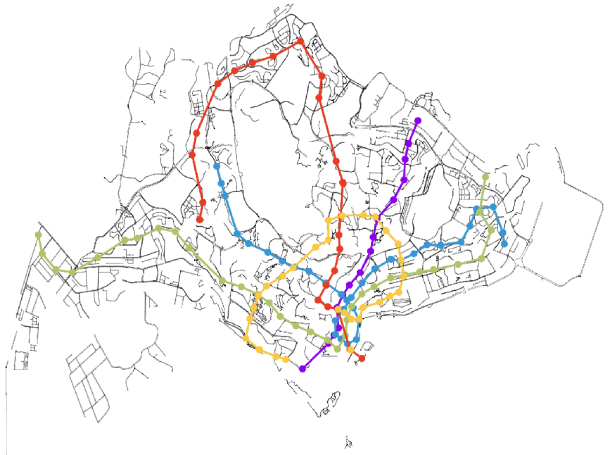
**Algorithmic Game Theory**

- Problems at the intersection of economics and computation
- Overarching goals of the field:
  - Design the rules of the game (a.k.a algorithms) so that strategic behavior by participants leads to a desirable outcome
  - Understand and analyze selfish behavior

*The next time we bemoan people exploiting loopholes to subvert the intent of the rule makers, instead of asking 'What's wrong with these people?" let's instead ask, `What's wrong with the rules?' and then adopt a scientifically principled approach to fixing them —Hartline and Kleinberg*

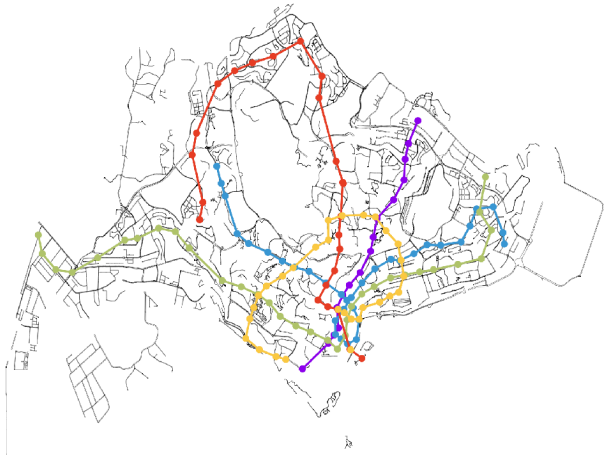
# Algorithmic Game Theory: Examples of Problems

- **Auctions.** Designing revenue maximizing auctions which incentivize bidders to bid their true value
- **Resource allocation.** Designing a fair allocation algorithm that incentivizes selfish participants to reveal true preferences
- **Selfish routing.** Analyzing global congestion in networks where each participant selfishly minimize their travel time

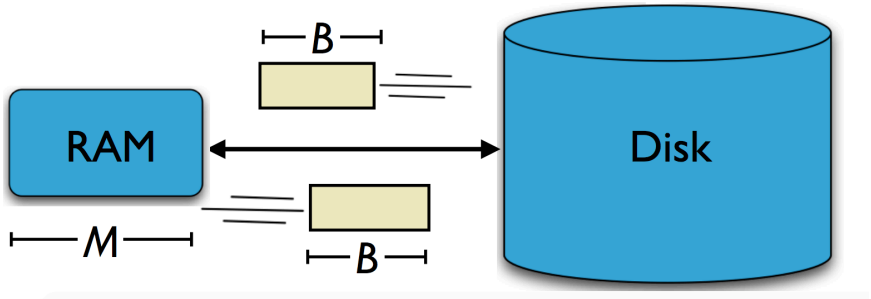
# Algorithmic Game Theory: Background

- What do you need to do work on work in this area?
  - Mathematical background, such as Discrete Mathematics
  - Some familiarity with designing and analyzing algorithms
  - Programming background useful but not necessary

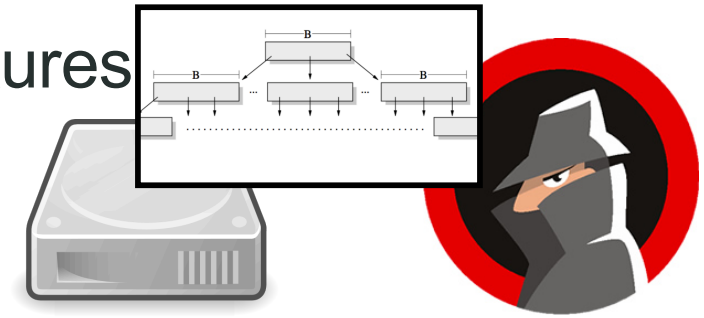



# Other Research: Algorithms and Data Structures

- I/O-efficient algorithms and data structures




- Randomized and adaptive data structures




- Optimization, Approximation, online and LP duality algorithms

# Aaron Williams

## Retrogame Archeology / History

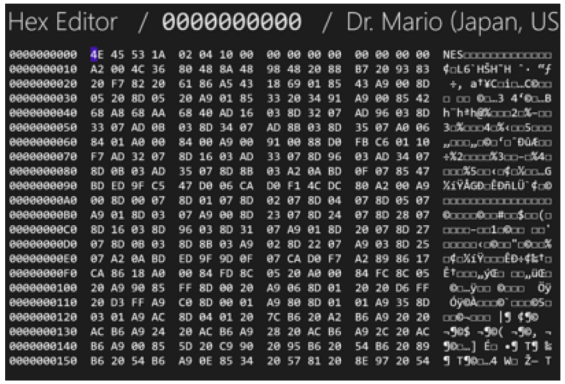
How were video games from the 1980's programmed?

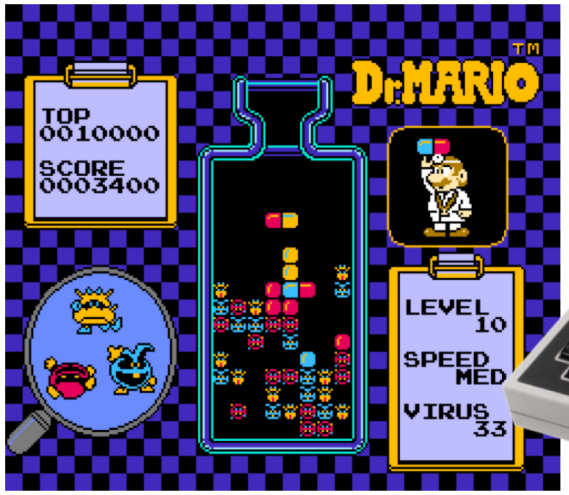
We examine the machine code and figure it out!

This is a new field and there are a lot of "important" problems to be solved.

- ~~Play~~ study video games all summer!
- Low-level system design and programming.
- Learn undocumented programming tricks and optimizations from masterful coders of yore.

**Requirements**: Programming experience.





Dr. Mario (NES) uses a 15-bit linear feedback shift register when generating random levels.

# Aaron Williams

## Computational Complexity of Puzzles and Games

Which type of puzzle is *harder*: Sudoku or Rush Hour?

| | | | | | | | | |
|---|---|---|---|---|---|---|---|---|
| 5 | 3 | | | 7 | | | | |
| 6 | | | 1 | 9 | 5 | | | |
| | 9 | 8 | | | | | 6 | |
| 8 | | | | 6 | | | | 3 |
| 4 | | | 8 | | 3 | | | 1 |
| 7 | | | | 2 | | | | 6 |
| | 6 | | | | | 2 | 8 | |
| | | | 4 | 1 | 9 | | | 5 |
| | | | | 8 | | | 7 | 9 |

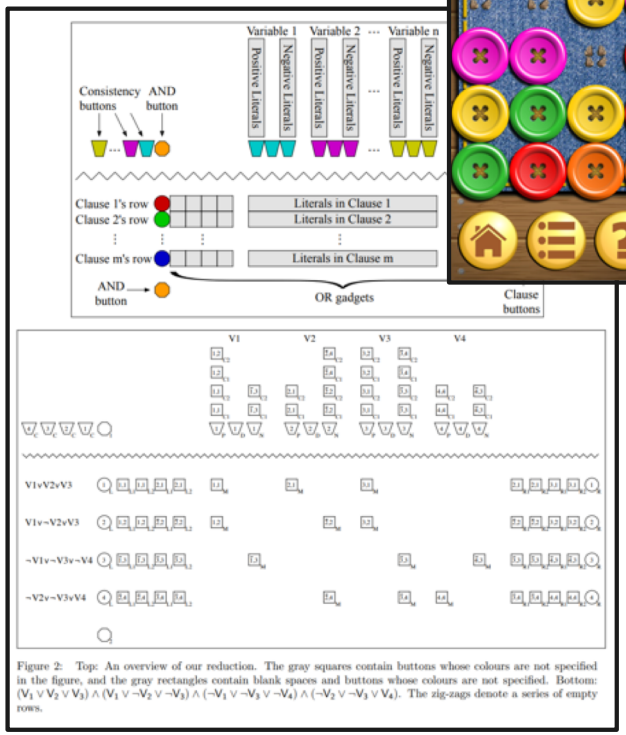




Figure 2: Top: An overview of our reduction. The gray squares contain buttons whose colours are not specified in the figure, and the gray rectangles contain blank spaces and buttons whose colours are not specified. Bottom: $(V_1 \vee V_2 \vee V_3) \wedge (V_1 \vee \neg V_2 \vee \neg V_3) \wedge (\neg V_1 \vee \neg V_3 \vee \neg V_4) \wedge (\neg V_2 \vee \neg V_3 \vee V_4)$. The zig-zags denote a series of empty rows.


Apply the area of the computational complexity to well-known puzzles and games.

- \+ Creative thinking.
- \+ Experience + supervision writing a research paper.
- \+ Turn computational complexity into an applicable skill for use later in your career.

The puzzle app *Buttons & Scissors* is NP-hard because it can model 3-SAT.

**Requirements**: CSCI 256 (required) / CSCI 361 (preferred).

# Aaron Williams

## Efficient Generation of Combinatorial Objects

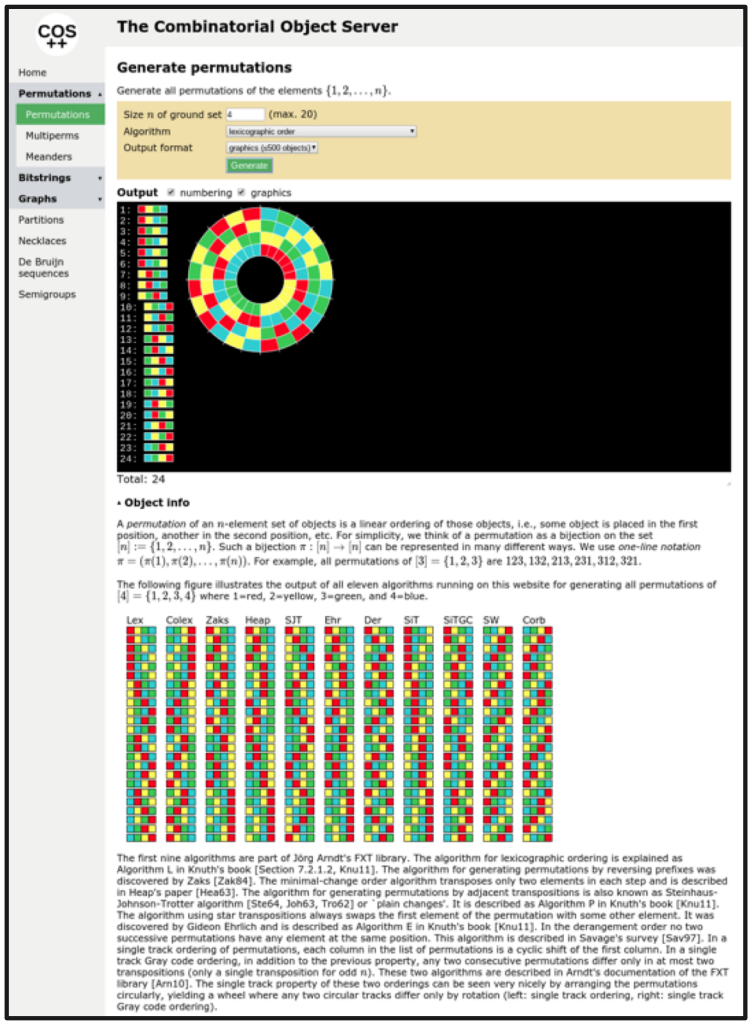
```
import itertools
for permutation in itertools.permutations([1,2,3,4,5]):
        # do something with the permutation
```

How is the above code implemented?
Why are the Python developers doing it wrong?

- Discrete Mathematics and Algorithms.
- Experience + supervision writing a research paper.
- Implement practical libraries that are used in high-efficiency applications.

Check out combos.org for more details.

**Requirements**: CSCI 256 / MATH 200




The Combinatorial Object Server page on permutations.

# Next Steps

- Come talk to us!
- Summer RA apps due Feb 17
- Thesis apps due Apr 10
- Forms can be found on dept website under research