APPlied Mathematics:
The Logic, Algebra, and Geometry of App Design

Amy Bigelow
Math Department Chair
East Haddam, CT
fa-ct.org
Students Served

- Top 12th graders taking AP & College courses
- Top 7th - 10th graders in Pre-Calculus
- 12th grade remedial math students
- 4th - 8th graders with no experience
Top Scholars

- AIME & USAMO Qualifiers
- Self-taught programmers
- Taking college courses through Wesleyan University & University of Connecticut
- Taken or preparing for AP exams in BC Calculus and Physics C
- Took this as easy, fun elective
- Still struggle with time management and long-term projects
- Now planning to take AP Computer Science Principles exam
  - Involves multiple choice exam, research project, and creating a program and writing a reflection and analysis
Weakest Students

- Struggle to understand the concept of variables
- Not yet comfortable finding percents of numbers
- Taking Life Skills and Common Sense math courses
- Enticed by future jobs in Cyber Security and Game Design
- Varied time management and self-advocacy skills
- Dropped the course after first semester
  - Now pursuing paths in Graphic Design, Video Editing, and Multimedia Applications
• One-day hackathons
• 4th - 8th graders with no prior experience
• Design an app that supports a non-profit's mission
APPlied Mathematics:

The Logic, Algebra, and Geometry of App Design

Franklin Academy
Belong | Grow | Succeed
Educating bright students with unique learning styles

Amy Bigelow
Math Department Chair
East Haddam, CT
fa-ct.org

References

Students Served

Mobile CSP Curriculum

Open-ended Projects

Teaching Strategies

Reflections

Resources
Mobile CSP Curriculum

- Approved for AP Computer Science Principles
- Uses MIT App Inventor to design Android apps
- Structures:
  - Step-by-step app design with enhancements
  - Website for portfolio write ups
  - Multiple choice with infinite guesses
  - Short programming drills
  - Practice Explore & Create tasks for AP
Common Core Practices

1.) Make sense of problems and persevere in solving them
2.) Reason abstractly and quantitatively
3.) Construct viable arguments and critique the reasoning of others
4.) Model with mathematics
5.) Use appropriate tools strategically
6.) Attend to precision
7.) Look for and make use of structure
8.) Look for an express regularity in repeated reasoning
Number Theory

- Using different bases (binary, hexadecimal)
  - With unit conversion (1 byte = 8 bits)
  - Basic combinatorics (How many values can this system encode?)
  - Exponents and very large numbers

- Modular arithmetic
  - mod 2 for error detection (parity arguments)
  - Pseudo-random number generators (recursion: \( x_{n+1} = (2x_n + 1) \mod 13 \))
  - RSA encryption: \( m^k \mod N \), factoring, and relatively prime numbers
Variables
Logic

- As physical gates and wiring diagrams
  - Simulated in logicly

- Abstrated to truth tables
  - With differences in English

- In flow charts as program outlines

- As parts of usable code
Geometry

- Determining necessary rotations to produce regular n-gons
Functions

- Moore's Law (exponential growth)
- Comparing run-time of sorting and searching algorithms (logarithmic, linear, quadratic, exponential, factorial)
- Simple substitution ciphers (f(x)=x+a)
• Free as PDF
• Examines issues of privacy and security in the 21st century
• Shows development of technology and how laws and practices connect
Internet Overview

• How was the internet built?

• How open-source protocols function

• Internet vs. world wide web

• Processing big data (MapReduce)

• Factors determining internet speed

• Understanding IP addresses and how data is transferred and processed online
How Computers Work

- Parts of a computer (RAM, CPU, integrated circuits)
  - Machine language, Assembly language, High level languages
  - How graphics & sounds are expressed using 0's and 1's
Practice for AP CSP
Explore Task

- Research a computing innovation
- Write ~1 paragraph explaining:
  - Its impact on society (benefits & drawbacks)
  - How the device deals with data
- Find relevant, recent sources and cite them
- Make a non-prose description of what the innovation does
Searching & Sorting Algorithms

• Searching
  • Linear
  • Binary
  • When to use each
  • Rough overview of Google's search algorithm

• Sorting
  • Bubble sort
  • Merge sort
  • Radix sort
  • Run-time/complexity
Data Visualization

- How to spot a misleading graph
- Using spreadsheets to add, average, find extremes
- Making charts & graphs with Google Sheets
- Drawing conclusions from visual representations
App Design

- Using Databases
- Soundboards
- Quiz Games
- Painting
- Logo Simulator
- Map Tours
- Whack-a-Mole
Painting

Provided template

Personalizations & Enhancements
Map Tours
"Lights Out"
Turn Off The Lights

Turn off the lights!
Logo Simulator
to drawPolygon [sidelength, sides]
do
  for each number from 1 to get sides by 1
do
    call forward [sidelength, pixels]
    call turn [360 / get sides, degrees]
Using Databases

- Clickers with App Inventor's database (TinyWebDB) and Google's Firebase
- E-mail or text Broadcasting Hub
APPlied Mathematics:
The Logic, Algebra, and Geometry of App Design

Amy Bigelow
Math Department Chair
East Haddam, CT
fa-ct.org
Student-led Projects

Open-ended Projects

Apps to Fill a Need
Context-Inspired Apps

Authentic project-based learning

Nautilus Game

Nautilus Tour

Battle of Groton Heights Book

Narratio Chat

Montage Initiative Education
APPlied Mathematics:
The Logic, Algebra, and Geometry of App Design

Amy Bigelow
Math Department Chair
East Haddam, CT
fa-ct.org
Teaching Strategies for Computer Science

Adapted from Neil C.C. Brown & Greg Wilson

Pair Programming  Live Coding  Debugging  Peer Instruction
Peer Instruction

1.) Give a mini-lesson or brief overview of concepts
2.) Pose a multiple choice question that 40-60% of students are likely to get right
3.) Students vote individually
4.) Students discuss in small groups
5.) Reconvene and vote again
6.) Respond as needed (move on or clarify)
Debugging

- Predicting outcome (solve on paper)
- Fixing provided mistakes
- Using students' natural mistakes as learning opportunities
  - Creating a climate that anticipates and encourages regular reflection and improvement
Live Coding

• I do/ we do method

• Stresses the problem solving process
  • Breaking into smaller parts
  • Testing code

• Have some parts already done!
Pair Programming

- One "driver"
- One "navigator"
  - This person can also research and interpret documentation
- Switch places frequently
APplied Mathematics:
The Logic, Algebra, and Geometry of App Design

Franklin Academy
Belong | Grow | Succeed
Educating bright students with unique learning styles

Amy Bigelow
Math Department Chair
East Haddam, CT
fa-ct.org

References
Reflections
Resources
Students Served
Mobile CSP Curriculum
Open-ended Projects
Teaching Strategies
Resources

- Infy.com/Pathfinders

*Take the course yourself over the summer!*

Free curriculum:

- Mobile CSP -- nice videos, should create your own questions

- Code.org -- works with iPhone, has [https://code.org/curriculum/algebra](https://code.org/curriculum/algebra)

- Beauty and Joy of Computing - all text-based, Snap! (not apps) [https://csforallteachers.org/group/bjcile](https://csforallteachers.org/group/bjcile)

- CS50 AP-- most rigorous
App Inventor Requirements

- A Google account (one account/file)
- Can run emulator on PC or OSX 10.12 or before (not High Sierra)
  - Otherwise use BlueStacks emulator
- Android device or tablet (~$100) that can use same wifi connection as computer
- iOS version "coming soon"

http://www.appinventor.org/book2
APPlied Mathematics:
The Logic, Algebra, and Geometry of App Design

Franklin Academy
Belong | Grow | Succeed
Educating bright students with unique learning styles

Amy Bigelow
Math Department Chair
East Haddam, CT
fa-ct.org

Mobile CSP Curriculum
Students Served
Open-ended Projects
Teaching Strategies
References
Reflections
Resources
Reflections

- AP?
- Best for younger students
- Lots of writing, little programming

- Don't use website/portfolio question format
- Multiple choice -> class discussion exercises
  - Pre-make some of the step-by-step apps
    OR live code as a class

- Prepare for tech setbacks!
APPlied Mathematics:
The Logic, Algebra, and Geometry of App Design

Franklin Academy
Educating bright students with unique learning styles

Amy Bigelow
Math Department Chair
East Haddam, CT
fa-ct.org
References


APPlied Mathematics:
The Logic, Algebra, and Geometry of App Design

Amy Bigelow
Math Department Chair
East Haddam, CT
fa-ct.org

Franklin Academy
BELONG | GROW | SUCCEED
Educating bright students with unique learning styles