ACM India at a Glance

- **ACM**: world's largest educational and scientific computing society
  - **Mission**: advancing computing as science and profession
  - **Members**: ~100,000 worldwide, ~13000 in India
  - Comprising students, faculty, professionals
- **ACM India Chapters**: ~200 student chapters, ~20 professional chapters
- **ACM-W India**: empowering women in computing
- **Research Initiatives**
  - Student research: ARCS Symposium, best doctoral dissertation, partial travel grant, PhD clinic and Anveshan Setu
  - SIG research conferences: CODS-COMAD, ISEC
- **ACM India Annual Event**
  - Discuss recent trends in technology and celebrate India's achievements in computing
- **Education Initiatives**
  - Summer and winter schools: ~2 week full-time course on technology area
  - Compute: Symposium on computing education
  - Teaching Partner Program: External experts offering a course
  - CSpathshala: inculcate computational thinking in schools
- **Learning and Professional Development**
  - Eminent Speaker Program
  - Industry Webinars, Education Webinars
  - Minigraphs: Comprehensive coverage of a tech area
  - ACM global resources: Digital Library, ACM Learning Center
- **New prestigious awards instituted**
  - Acknowledge and celebrate outstanding contributions
- **ACM Membership in India**
  - Student? student member form
  - Professional? professional member form

Increasing use of Active Learning in Computing

Barbara Ericson

barbarer@umich.edu

Assistant Professor

School of Information

University of Michigan

Problem

- Many courses use passive learning rather than active
  - Large failure rates
  - Students with less prior experience more at risk of failing
- Active Learning can improve student learning, motivation, and retention
  - Piaget – constructivism – students learn by constructing knowledge
  - Vigotsky – social learning – students learn best when interacting with others
  - Papert – constructionism – learning is most effective when creating a meaningul product

Research Projects

- Free and interactive ebooks on Runestone
- Mixed up code (Parsons) problems
- Support for Peer Instruction

Runestone Academy

- Founder
  - *Democratizing textbooks for the 21st century*
    - *Everyone should be able to learn computing*
  - Started with one ebook in 2011
    - Now 30+ for computing and math
  - Average of 2 million page views a week
    - Top 1% of websites in the world

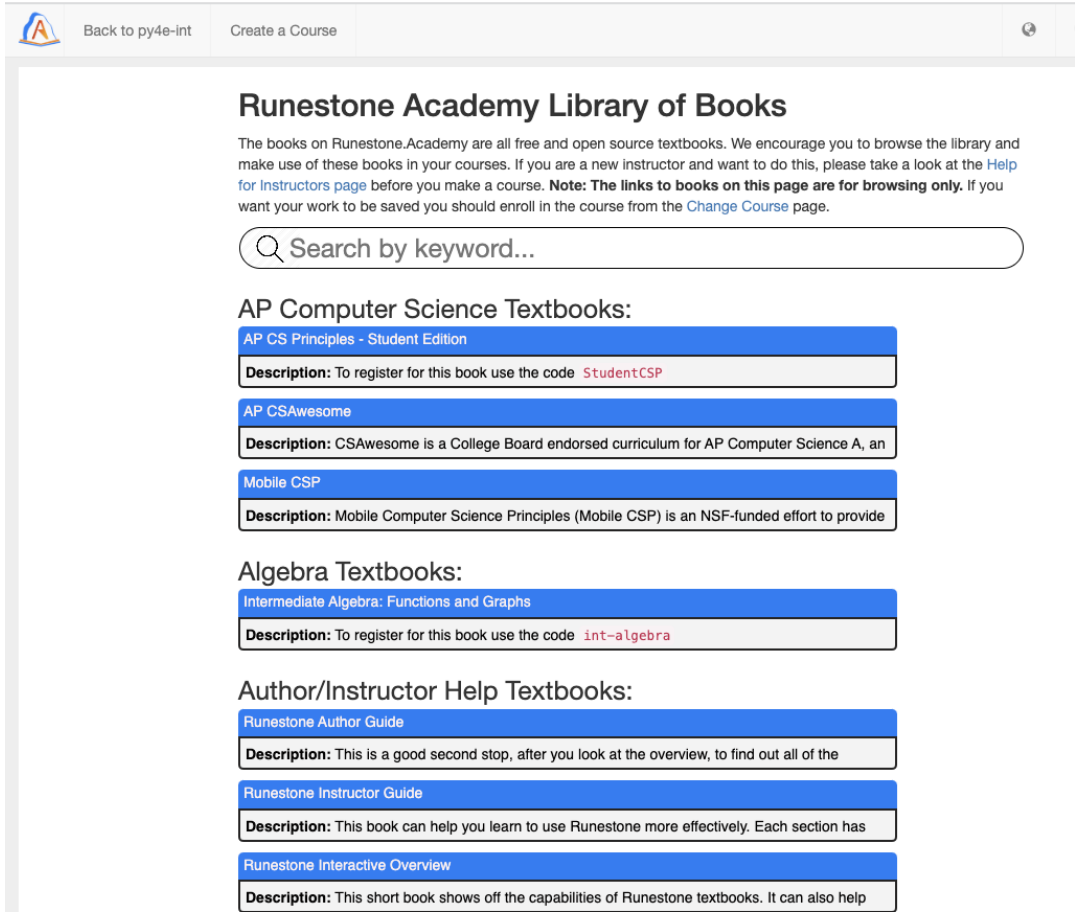
Brad Miller

Ericson and Miller, 2020, Runestone: A Platform for Free, On-line, and Interactive Ebooks, SIGCSE

List of Ebooks on Runestone

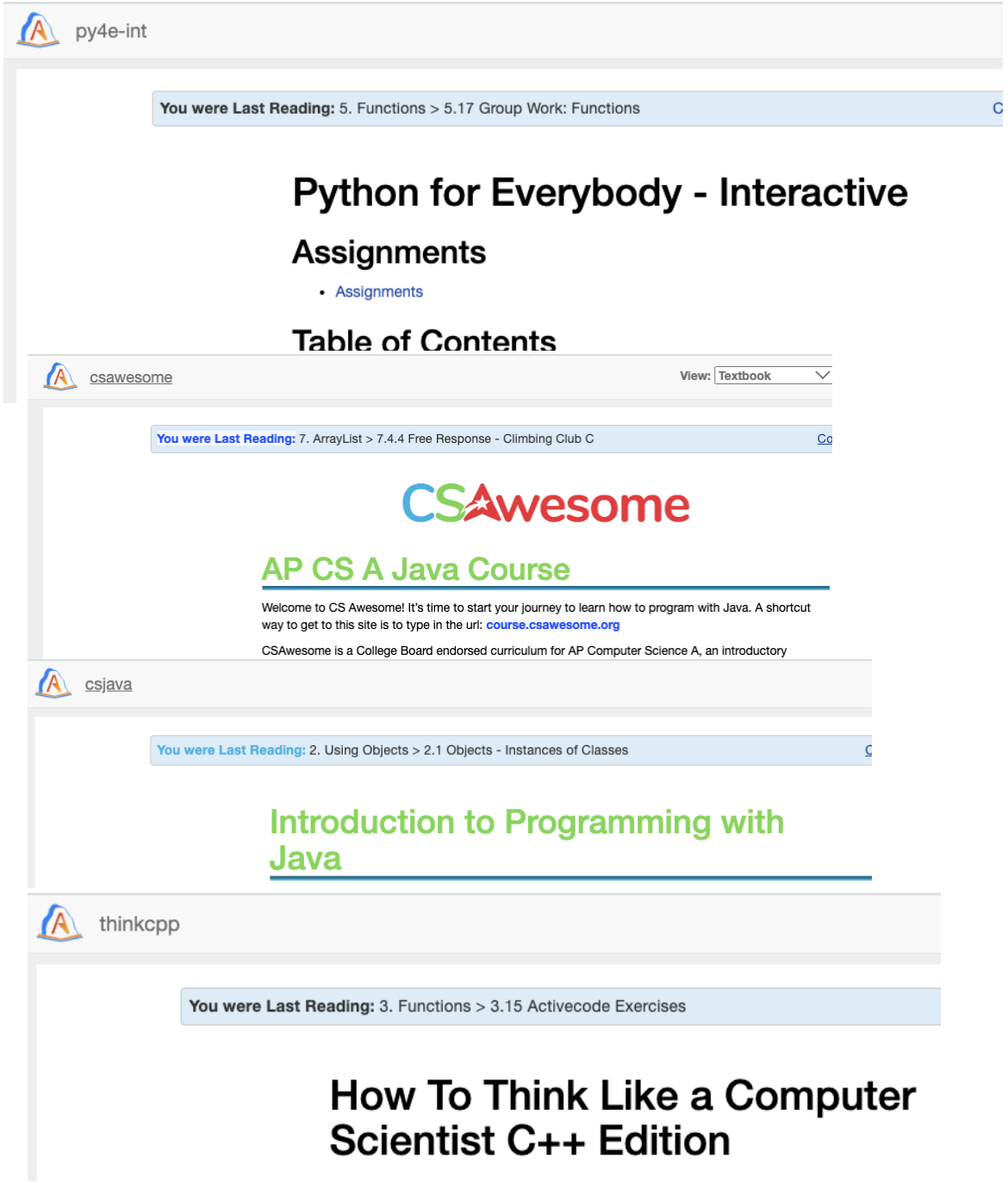
https://tinyurl.com/a48adr85

- 30+ free ebooks for computing and math



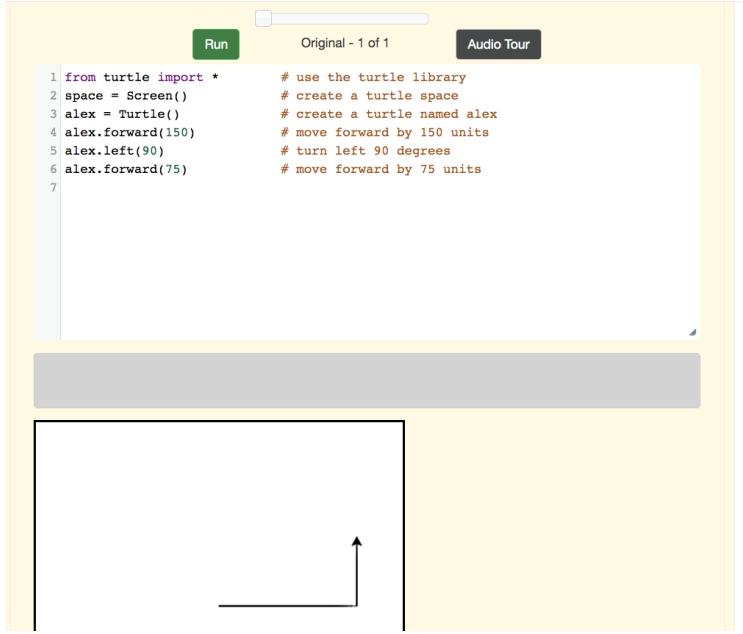
My Current Ebooks

- Python for Everybody – Interactive (py4e-int)
- CSAwesome (csawesome)/CSJava (csjava)
- How to Think Like a Computer Scientist – C++ (thinkcpp)
- Student CSP (StudentCSP)
- Discrete Math (dmoi)



Interactive Ebooks

- Educational Psychology
  - Interleaved worked examples plus practice problems - Sweller
  - Assessments with immediate feedback



csp-5-1-1: Which way does a turtle face when it is first created?

- A. North
- B. South
- C. East
- D. West

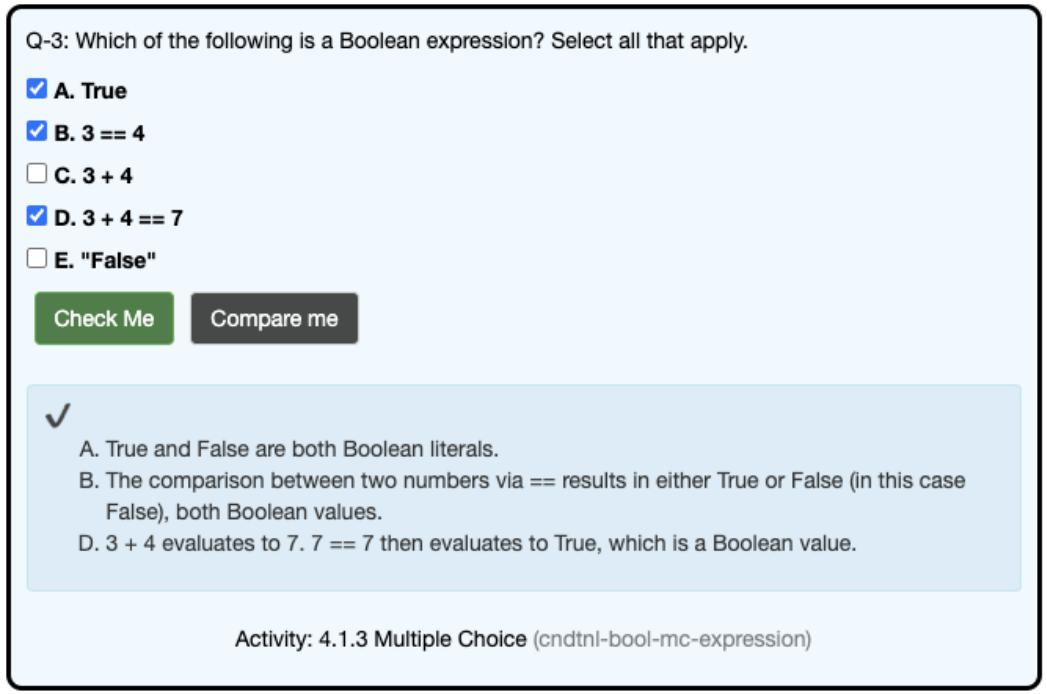
Check Me | Compare me

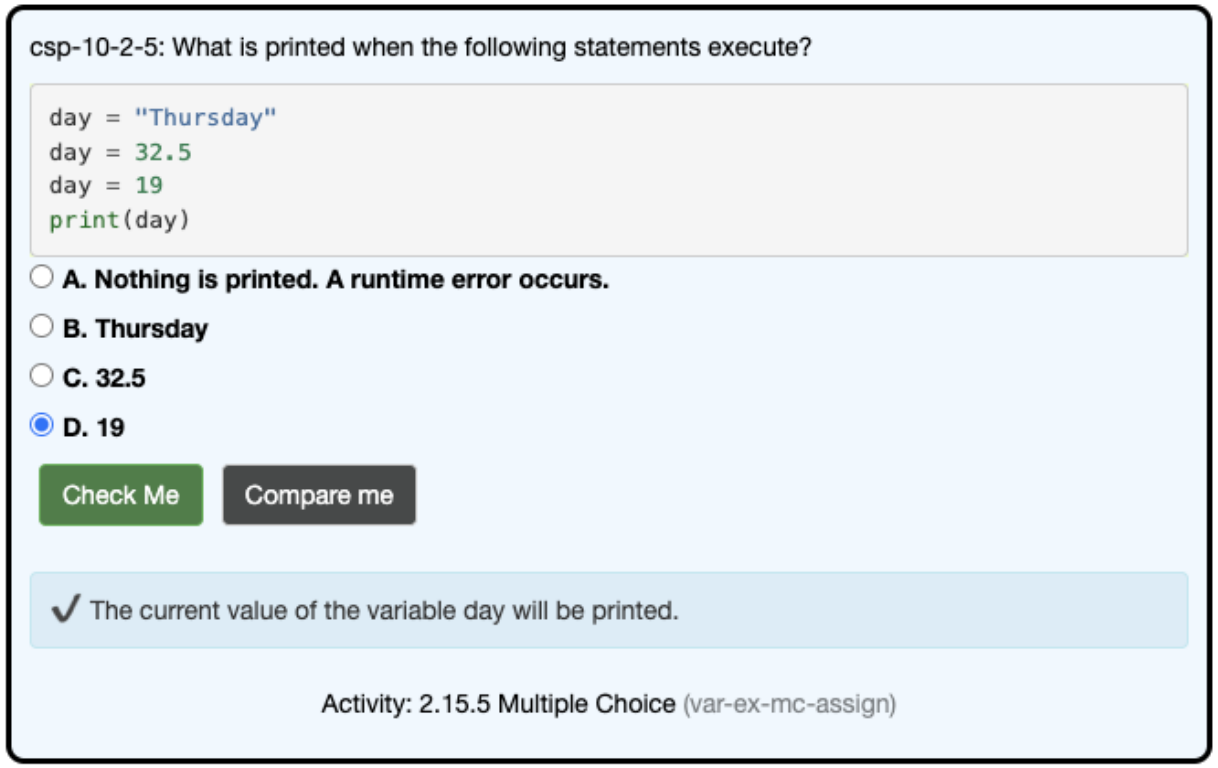
✖ The turtles in some of the examples faced north because of the `setheading(90)` instruction. Which way does chad move first?

Activity: 2 -- Multiple Choice (5_1_1_Turtle_Dir_Q1)

Multiple-Choice Questions

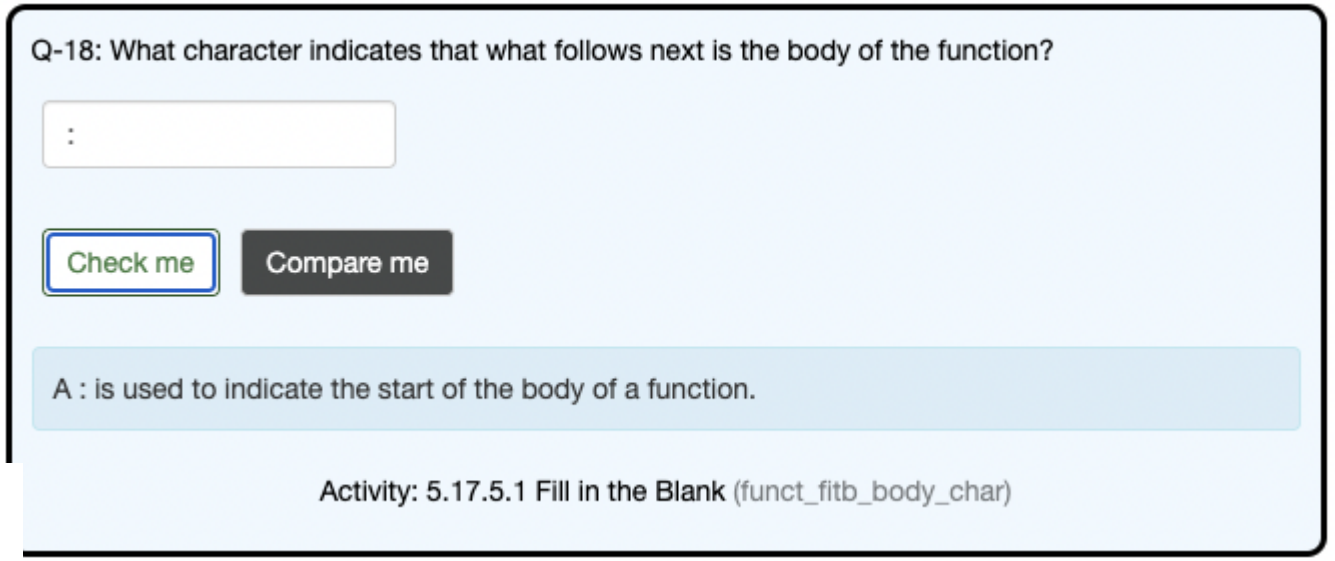
Single correct or multiple-correct

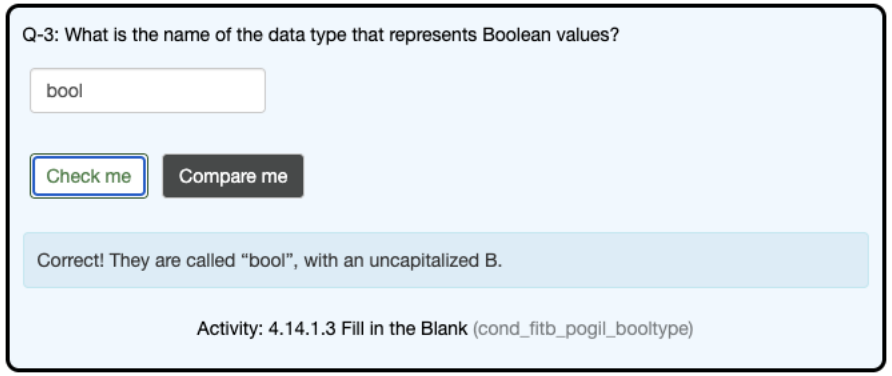




Fill in the Blank

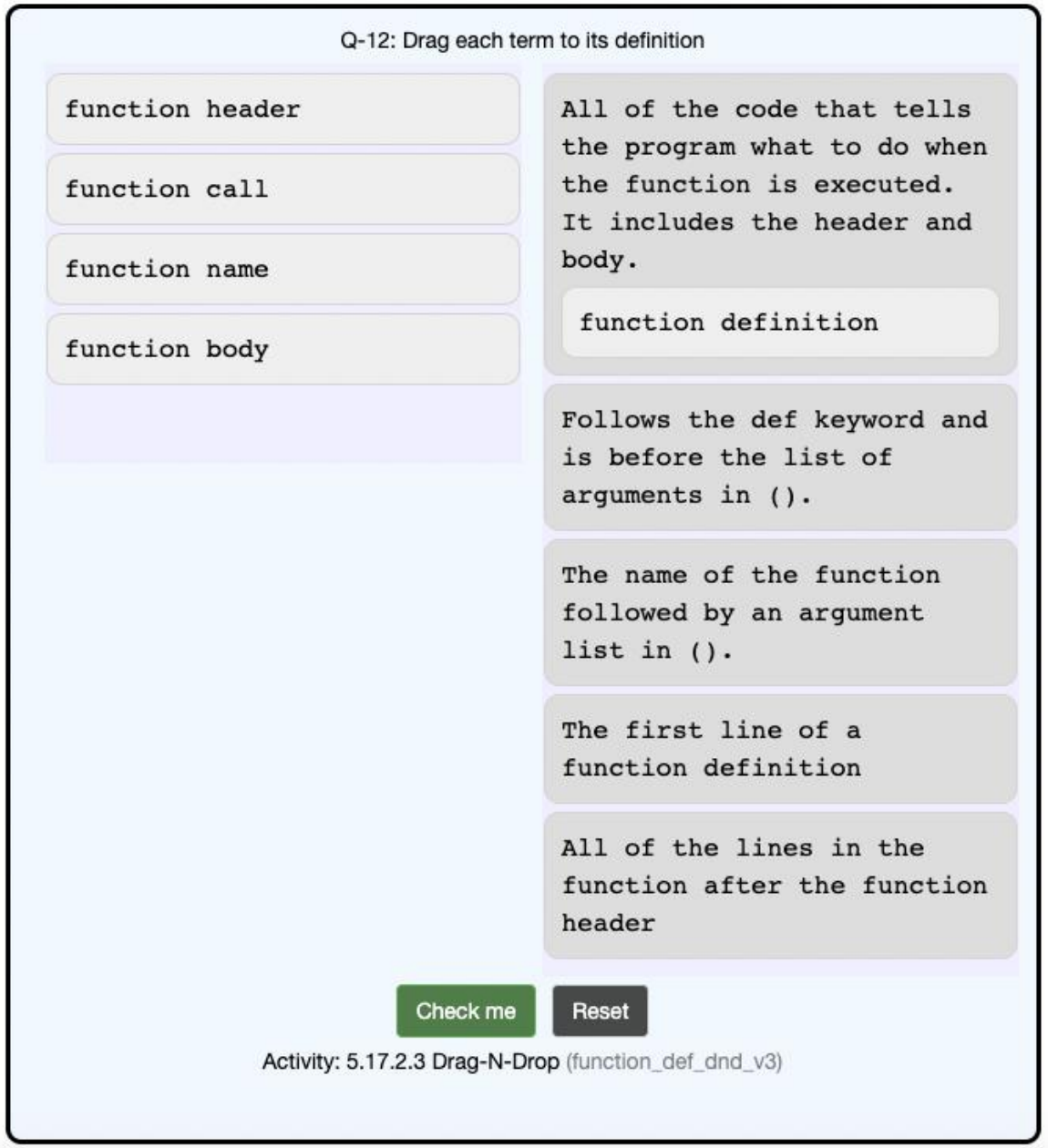
Uses a regular expression to check the answer





Matching

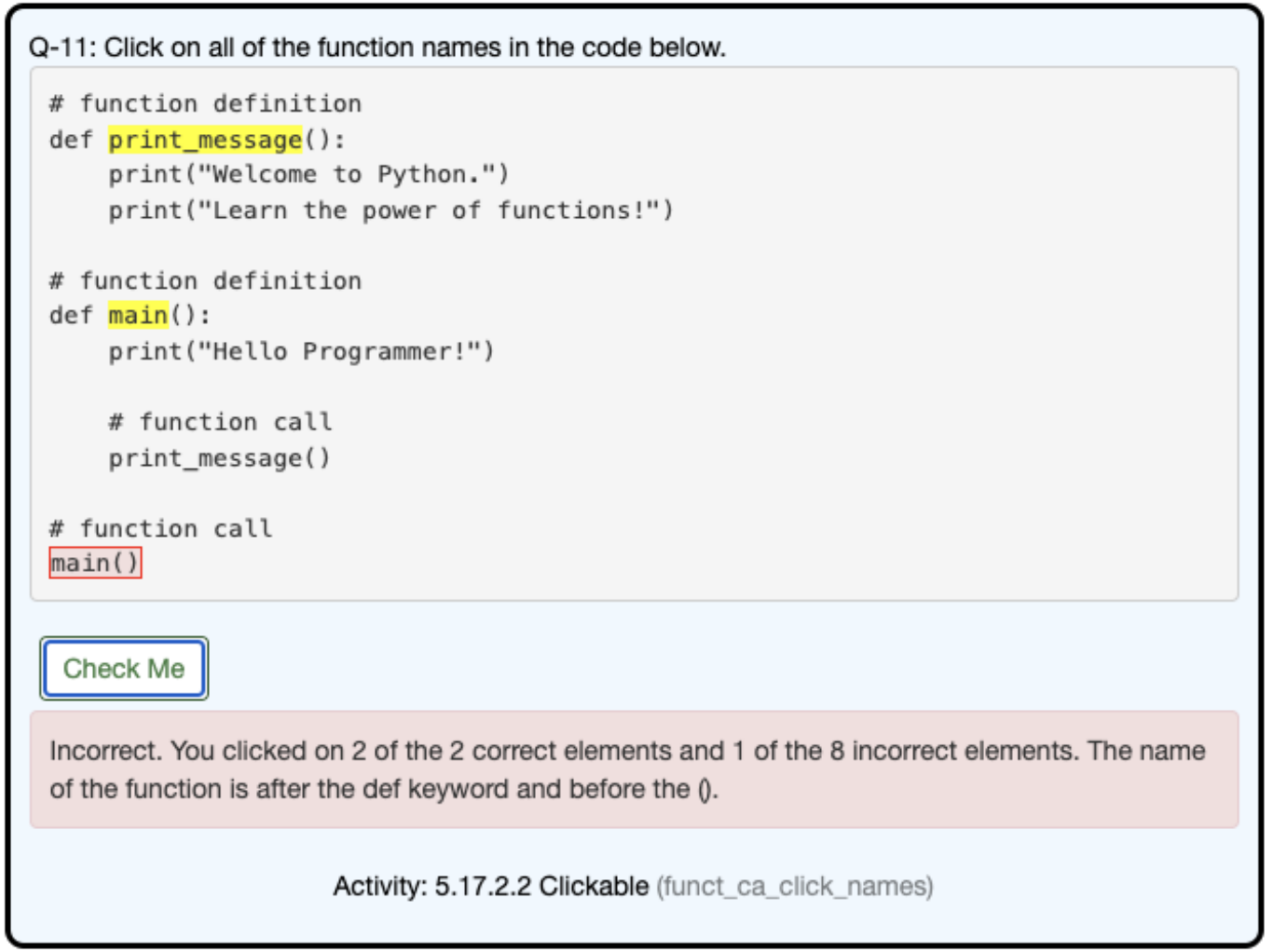
Drag terms to definitions



Clickable Code

Click on code to answer a question

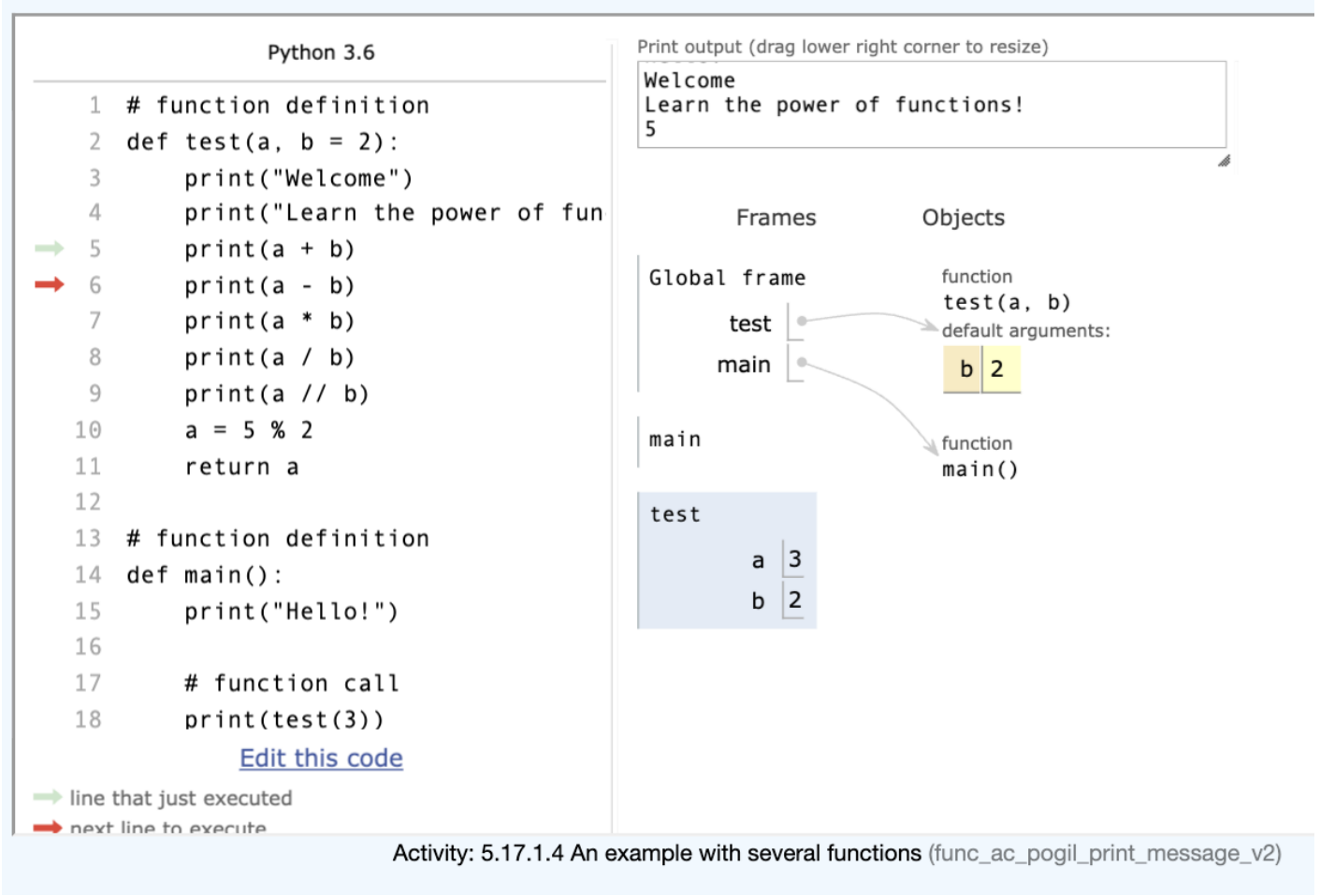
- Highlights correct and incorrect answers



Codelens with Optional Predictions

Step through code

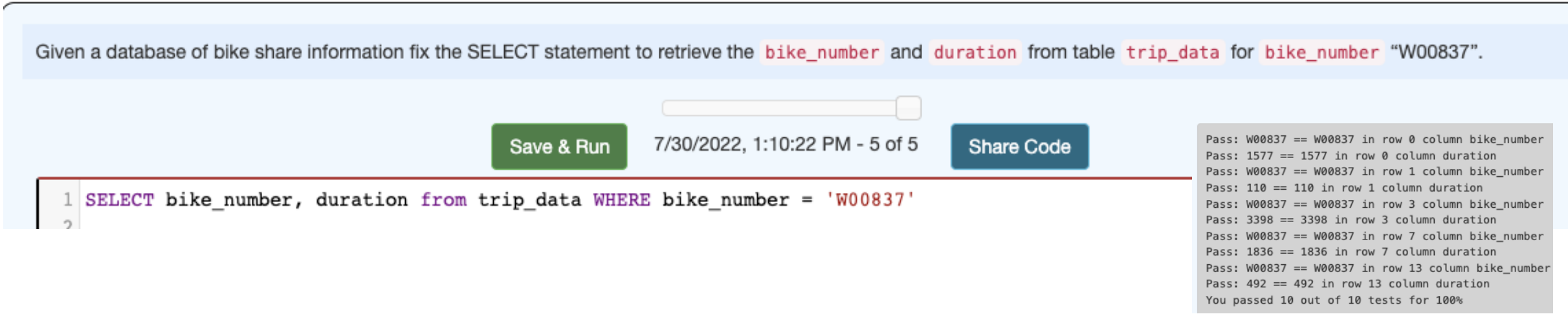
- Visualize variable values



Activity: 5.17.1.4 An example with several functions (func_ac_pogil_print_message_v2)

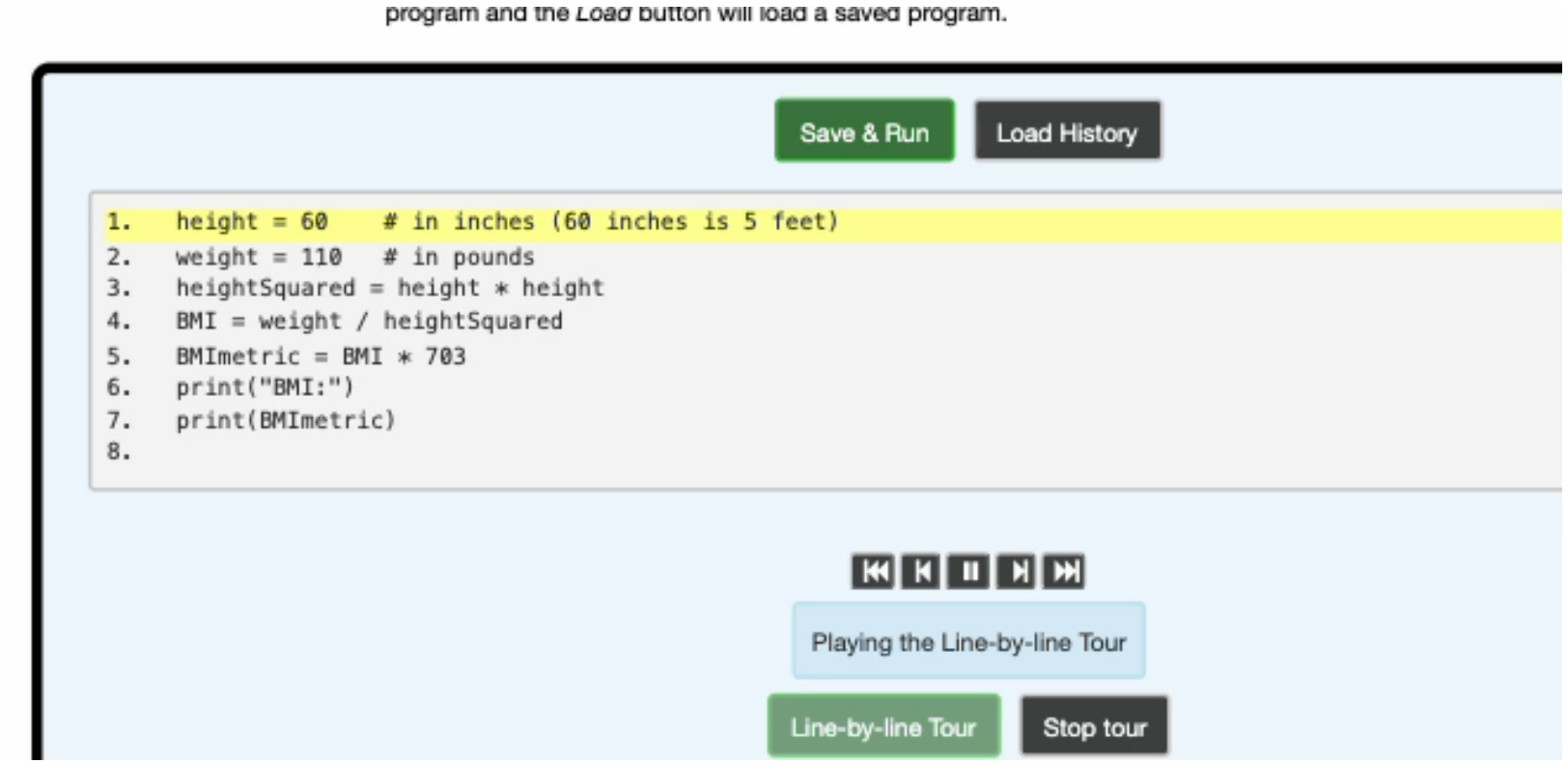
Write Code - ActiveCode

- Executable and editable
  - Python (JavaScript), C++, Java, SQL
- Can have unit tests



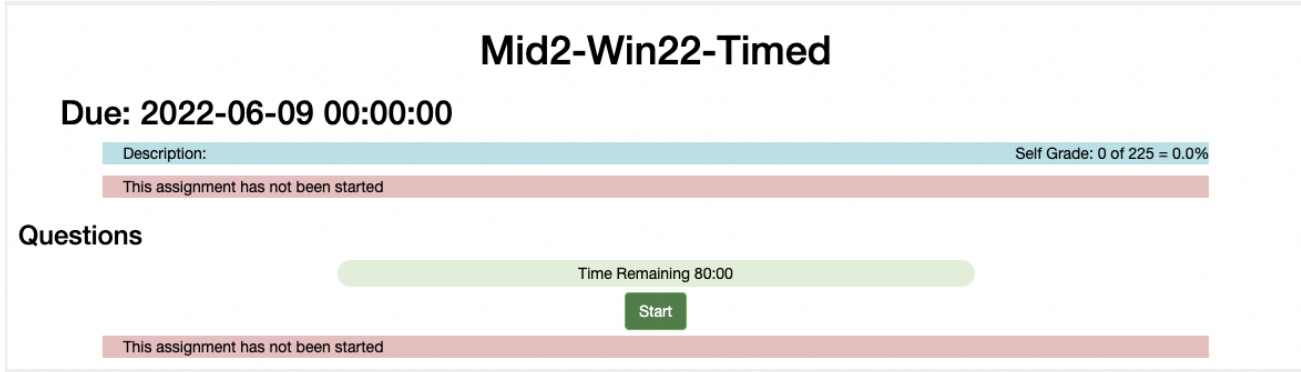
Audio Tours of Code

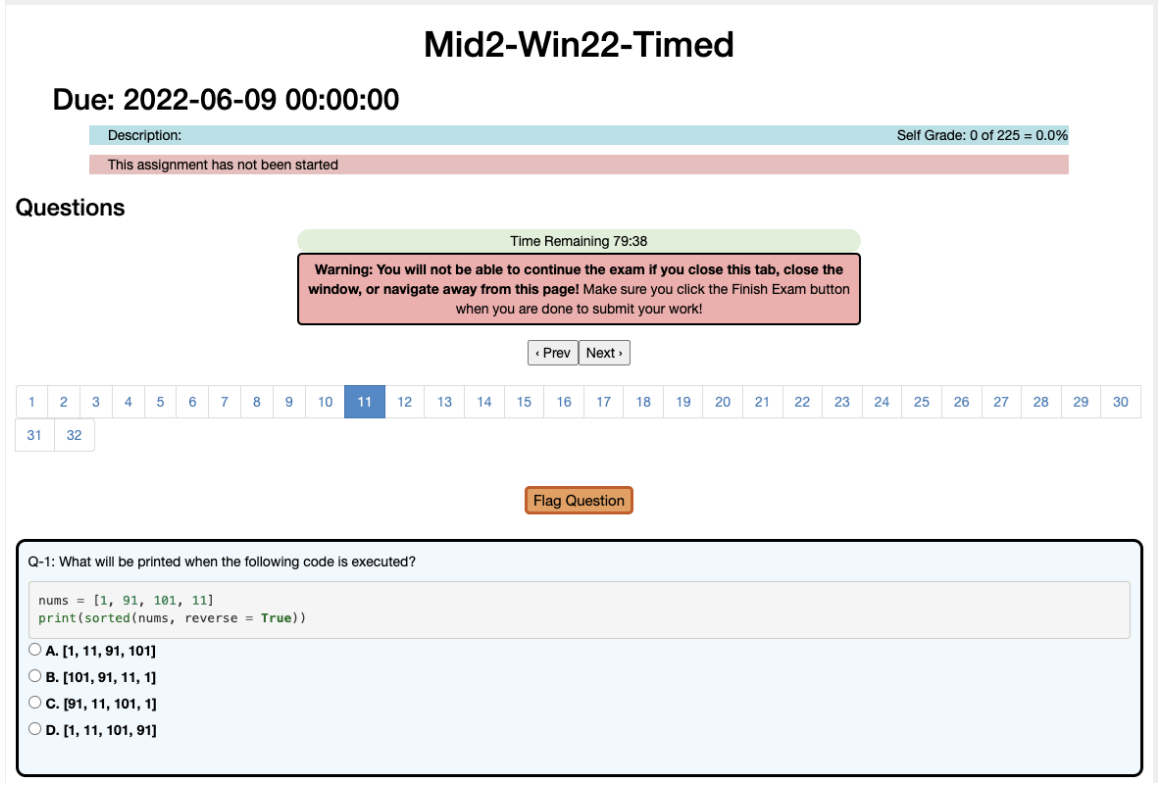
- Highlight a line or lines as audio plays explaining the code



Timed Exams

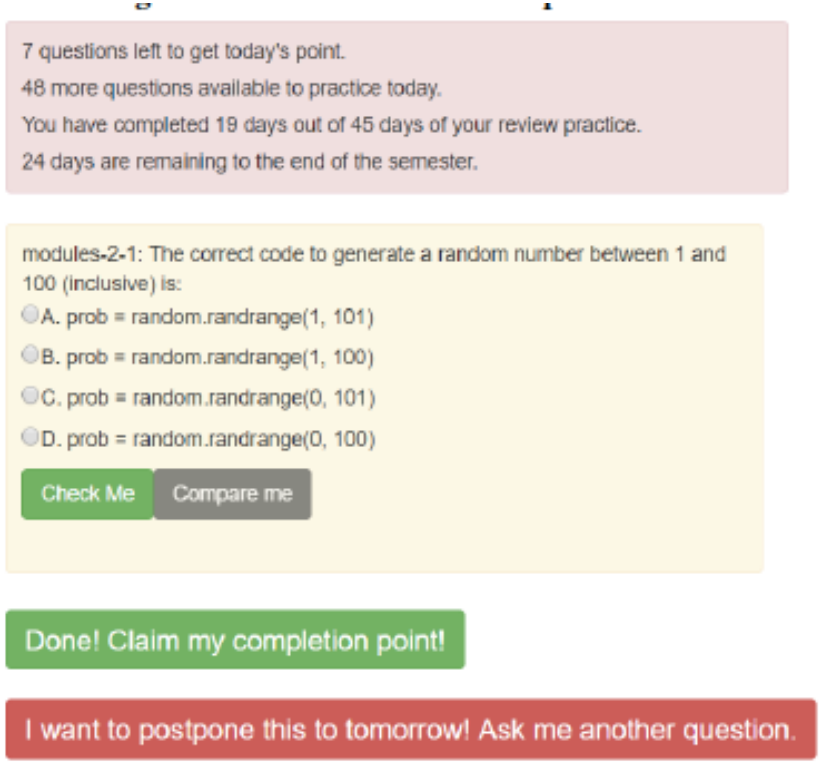
- Can set a time limit
- Can include
  - Multiple choice
  - Fill in the blank
  - Parsons
  - Write Code with Unit Tests
- Autograded
  - Can also manually grade
- Can have a list of problems for each question
  - Pick one at random





Spaced Practice Tool

- Give x points for solving y number of problems a day over z days
- Predicts when you are about to forget a topic
- Each hour of practice was associated with a 1% increase in the final grade

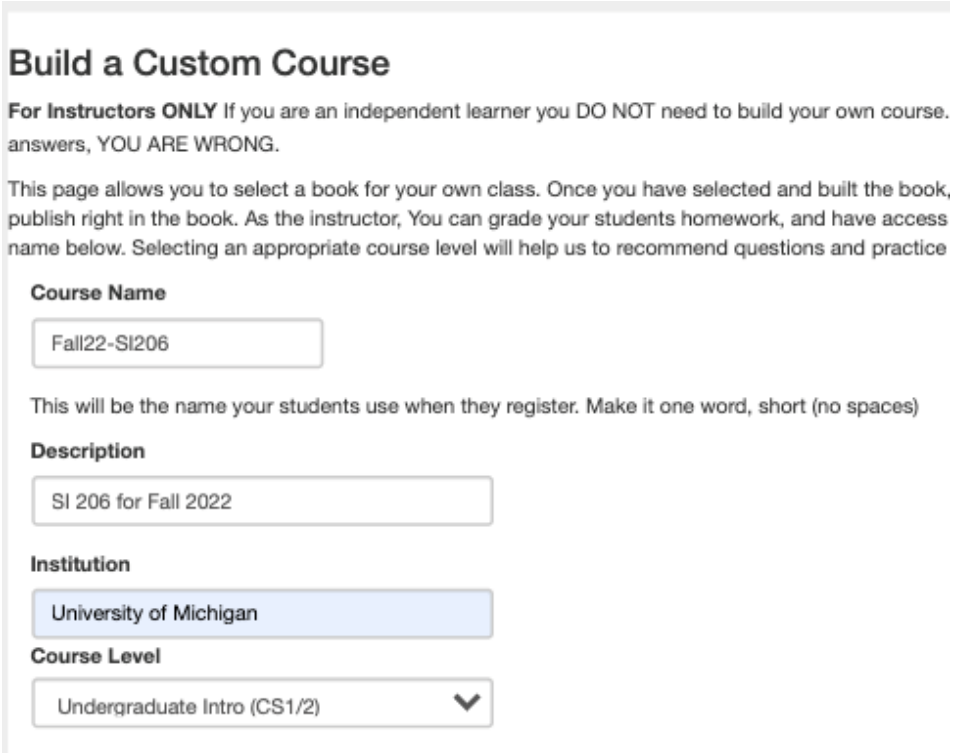


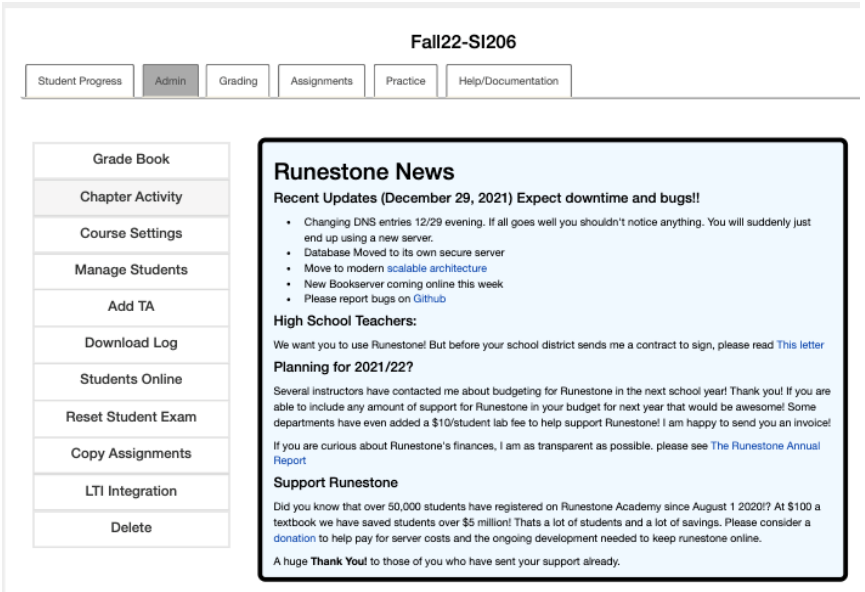
Iman Yeckehzaare

YeckehZaare, Resnick, and Ericson, 2019, A Spaced, Interleaved Retrieval Practice Tool that is Motivating and Effective, ICER

Features for Instructors

- Create a custom course from any existing ebook
  - Have students register for that course
  - https://tinyurl.com/y6uhbvnm
- Create assignments
  - From existing readings and problems
  - Or create new problems
- Automatically Grade Assignments
- Instructor guide (instructorguide) https://tinyurl.com/5n99fbtn





Research on Ebooks

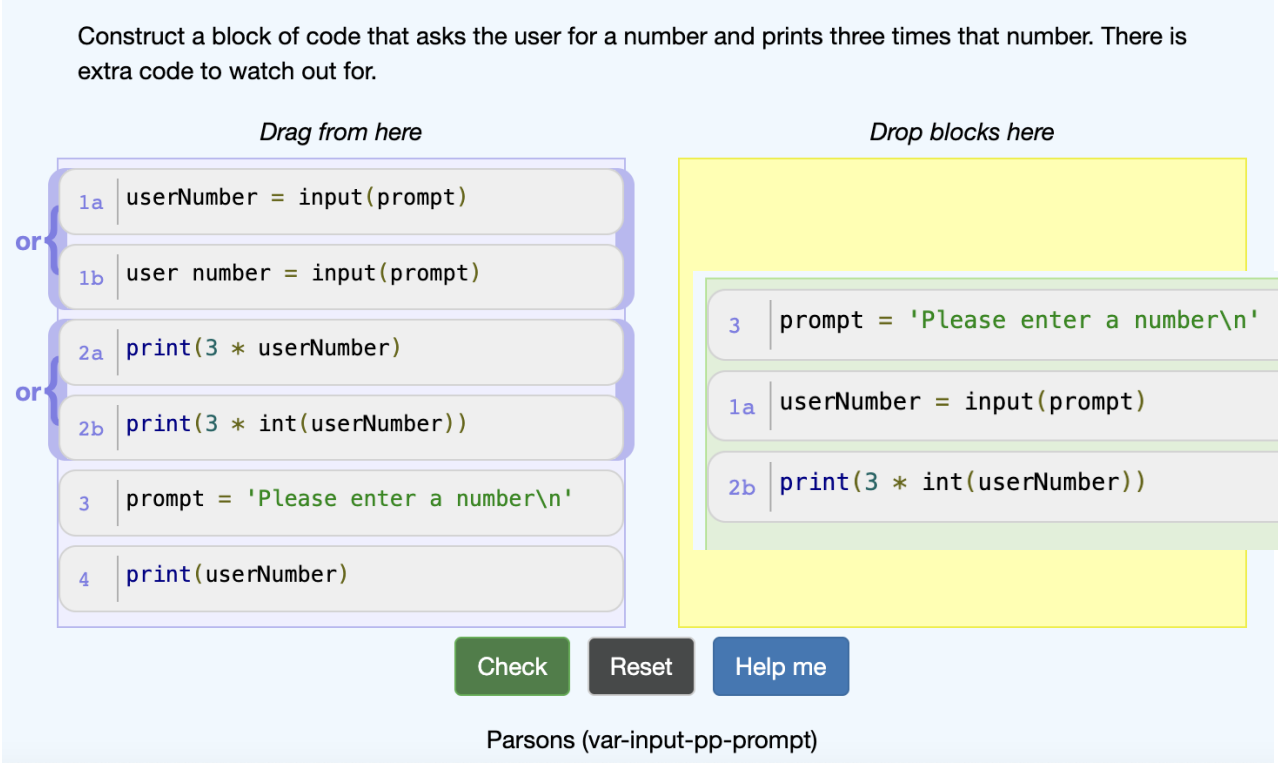
- Better learning gains than a static ebook - Pollari-Malmi et al 2017
- Increase in student performance and satisfaction - Croft and England 2019
- Reading time in an ebook does not always correlate with performance - interaction time is better - Smith et. al. 2021
- Can use ebook data to identify struggling students - Gökhan et al 2019
- Teachers who used more of the interactive features and spent more time in an ebook had higher confidence in their ability to teach the material - Ericson et al 2016

Free Online Courses – Python

- Expressing yourself in Python
  - https://tinyurl.com/2npzamhn
- The Power of Object-Oriented Programming
  - https://tinyurl.com/3ujekjv9
- Debugging: Hunting and Squashing Bugs
  - https://tinyurl.com/3tefuj92

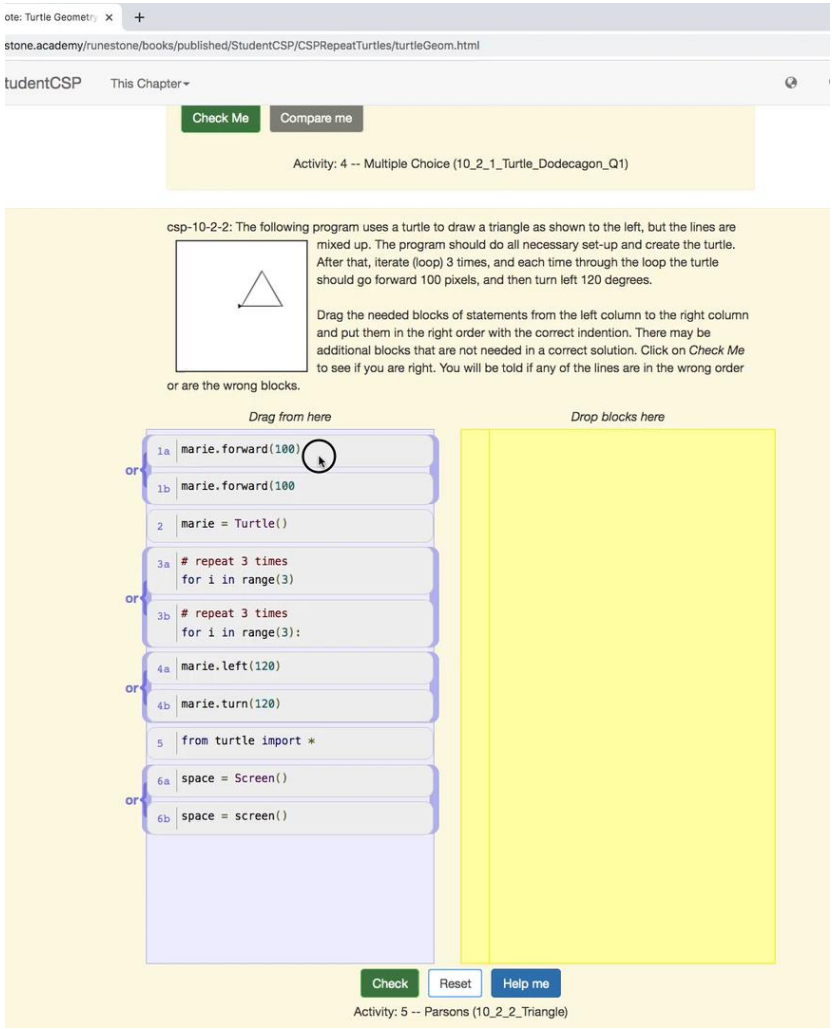
Parsons Problem

- Has a problem description
- Code fragments that must be placed in order
- Can require indentation as well
- Can have distractor blocks
- Should be less cognitive load than writing the code



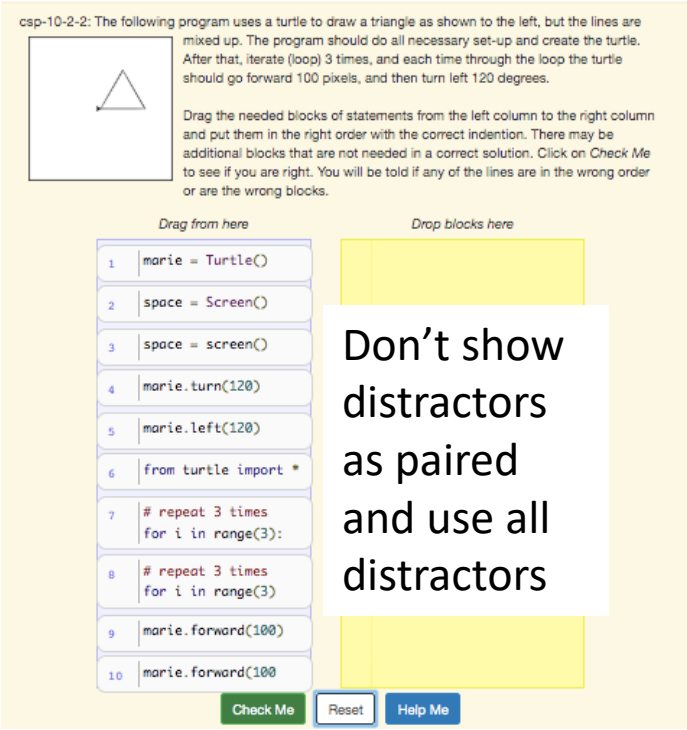
Adaptive Parsons Problems

- Intra-problem
  - If the learner is struggling to solve the current problem
    - Remove Distractors
    - Provide Indentation*
    - Combine Blocks

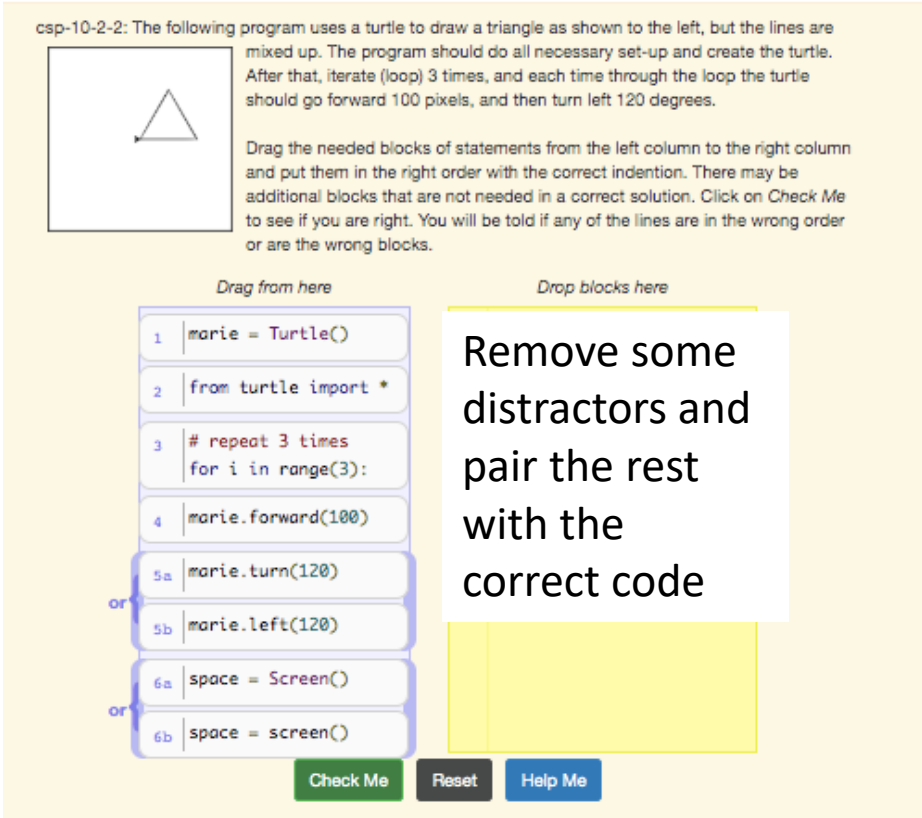


Adaptive Parsons Problems

- Inter-problem
  - If user solved the last one easily, make the next one harder

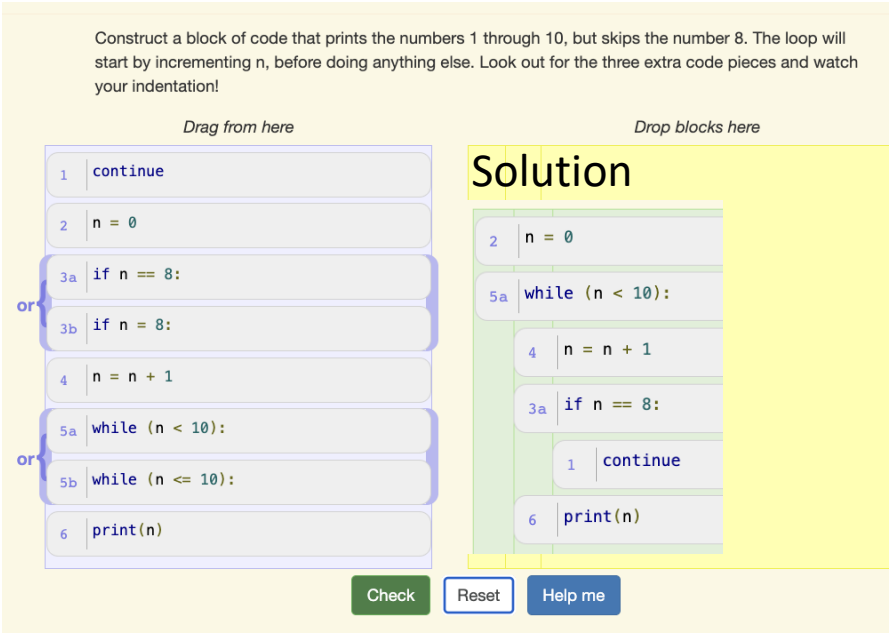


If many attempts, remove some distractors and pair them with the correct code



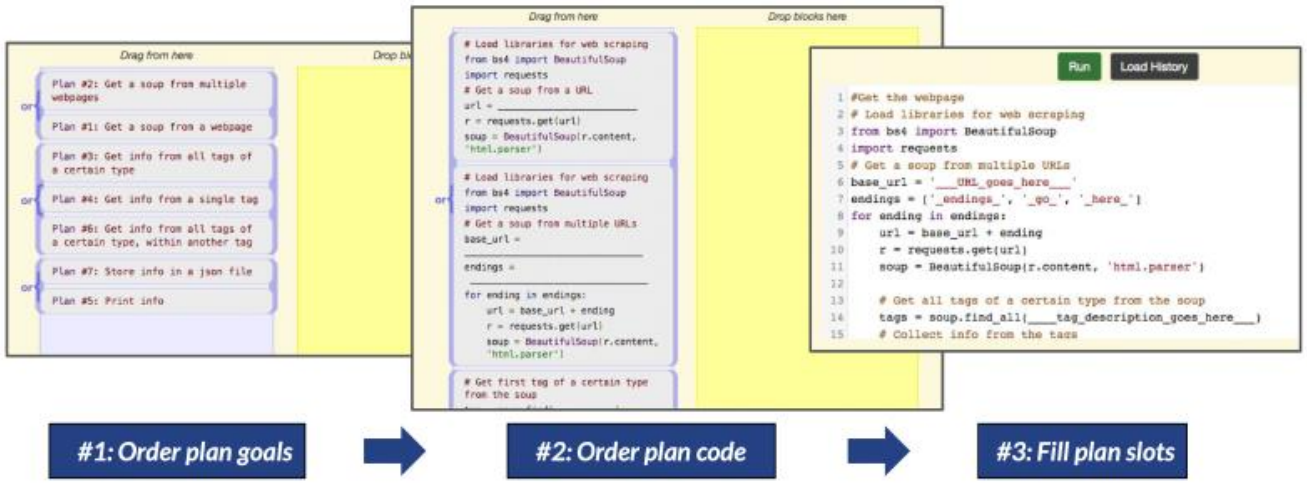
Parsons Problem Findings

- Signifcantly faster to solve Parsons than fix code or write code
  - similar learning gains
- Students ~2x as likely to solve adaptive Parsons problems
- Teachers felt they helped them learn to fix and write code
- Most students find Parsons useful for learning to program
  - Some prefer to write code from scratch



Purpose-First Programming

- Conversational Programmers
  - Don’t want to trace code
  - Don't want to be software developers
  - Liked learning at the plan level
  - With real-world code



Kathryn Cunningham

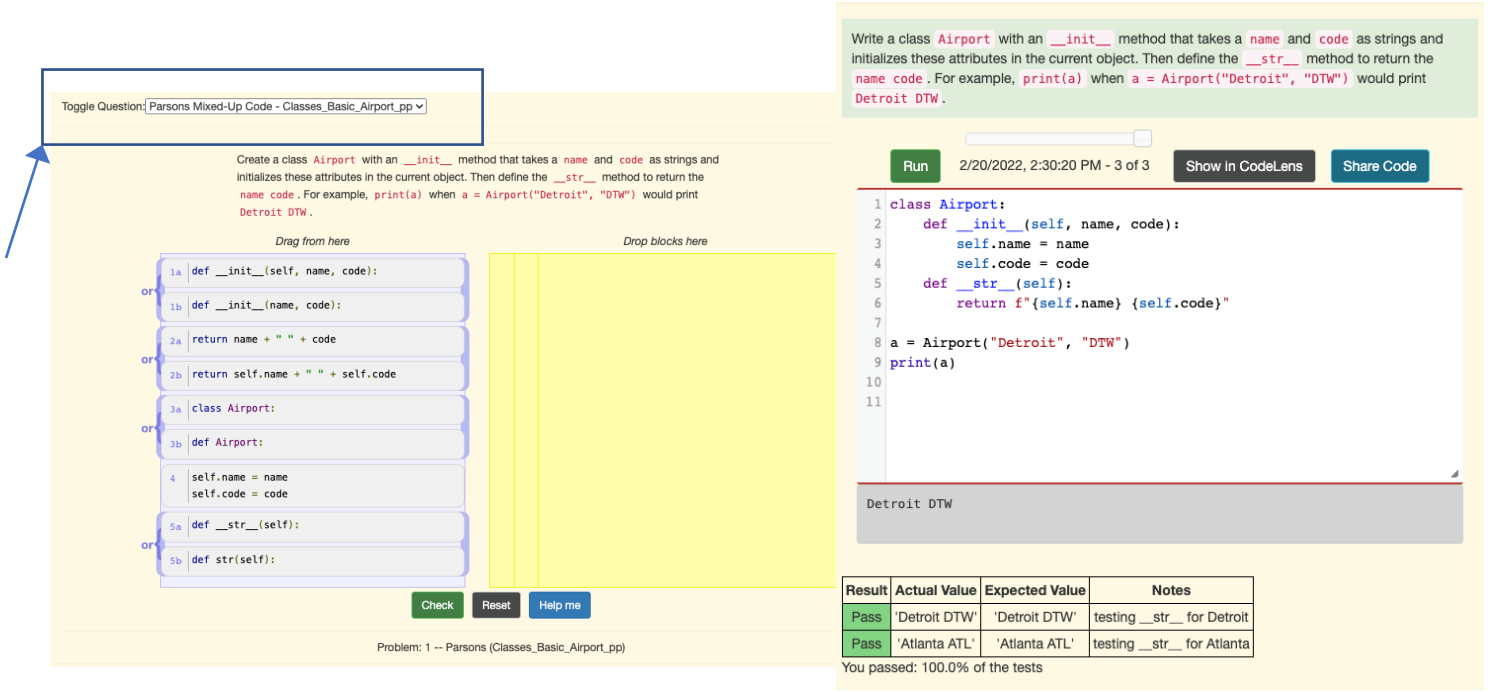
Cognitive Load and Efficiency of Parsons Problems versus Write Code

- Within-subjects study with undergraduate students
  - Lower mean cognitive load for Parsons
    - Not always significant
  - Lower median time to solve for Parsons
    - Not always significant – unusual solution
  - Ordering effect if solved unusual solution first
- Think-aloud observations of 11 students
  - Some hate Parsons problems
- Study with neurodiverse students

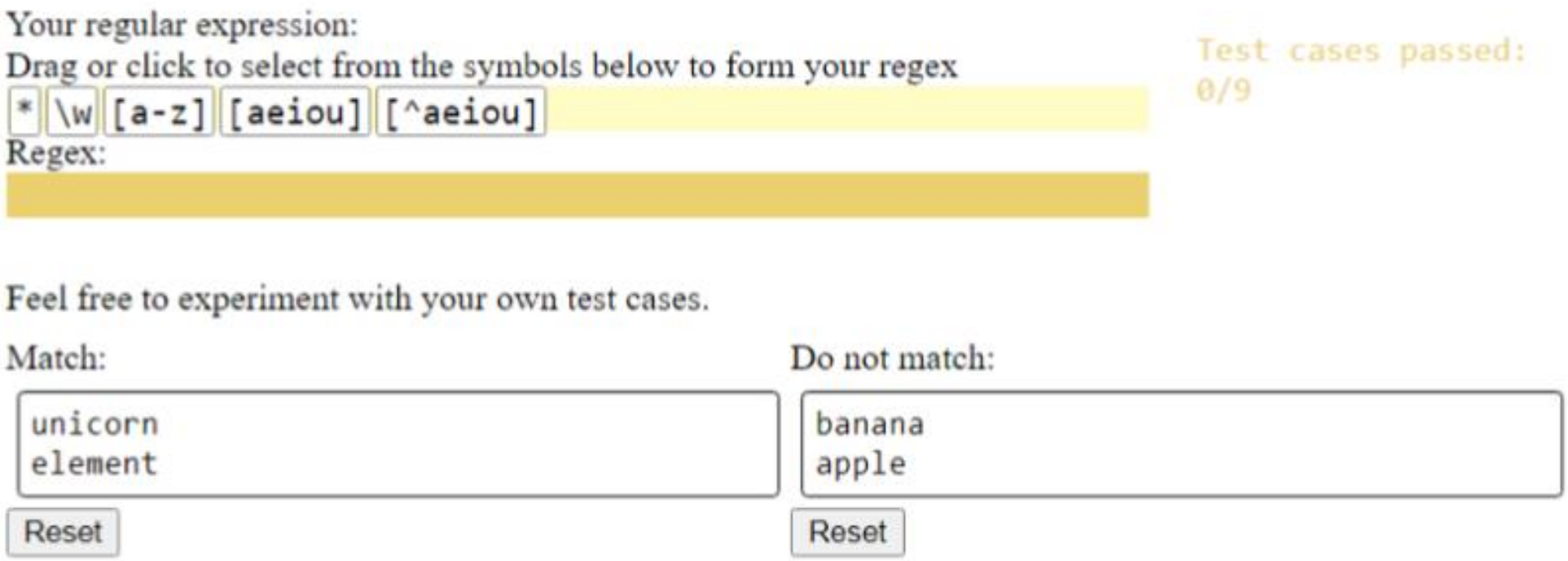
Carl Haynes-Magyar

Toggle Parsons Problems

- Users can solve either the Parsons problem
  - Or toggle to solve the equivalent write code problem with unit tests
  - Grades whichever one the user leaves selected



Horizontal Parsons Problem for Regex



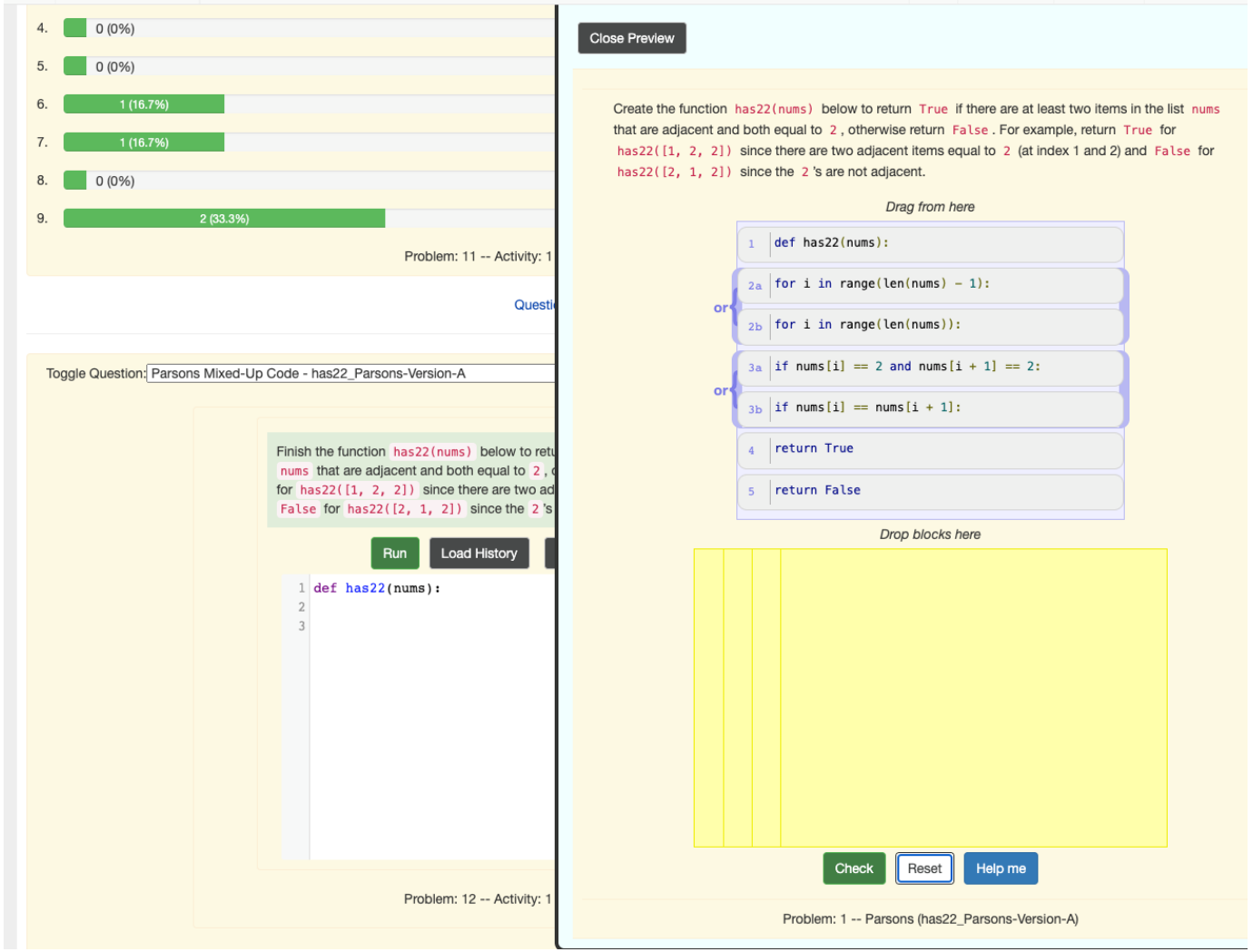
Horizontal Parsons Problems vs Text Entry

- Within-subjects think-aloud study
  - Horizontal Parsons are easier than text entry
  - Parsons problems are more limiting
- Between subjects-study optional assignment in a MOOC
  - Text entry condition had a significantly higher drop-out rate
  - Parsons condition had higher learning gains on the meaning of symbols and took less time to complete

Zihan Wu

Parsons as Hint While Writing Code

- When writing code
  - Can pop-up a Parsons problem
- Must still solve the write code problem

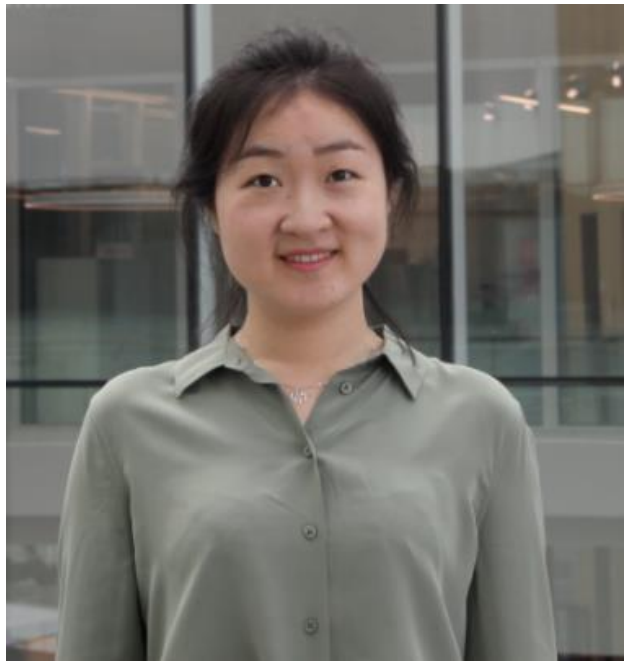


Parsons as Hint While Writing Code

- Think aloud study – 11 students
  - Targeted help vs search
  - Right direction
  - Syntax reminder
  - Makes me think
- Between-subjects study
  - Group in hint condition had better learning gains on write code problems

Xinying Hou

Xu Wang



ITiCSE Working Group on Parsons Problems

- Extensive Literature Review
- Studies in a Box
  - Python 3 + Lists and Dictionaries – Parsons vs Write Code
    - https://tinyurl.com/2p8t6yup
  - Python 3 + Lists and Dictionaries – Write Code vs Write Code with Parsons as Scaffolding
    - https://tinyurl.com/2af62pad
  - Python 3 + Classes – Parsons with Distractors vs without
    - https://tinyurl.com/yedw2xxv

Peer Instruction

- Eric Mazur from Harvard
  - Realized that students could solve textbook problems in Physics
  - But couldn’t apply what they knew in real-world situations
  - Students do better if they discuss their answers with each other

Peer Instruction Process

- Students read material before lecture (optional)
- Assessment before or at the beginning of lecture (optional)
- Instructor displays a hard multiple-choice question every 15 - 20 minutes
  - Students answer individually
  - Students discuss answers with peers
  - Students answer individually again
  - Instructor shows the results of both answers and leads a discussion

Catherine H Crouch and Eric Mazur. “Peer instruction: Ten years of experience and results”. In: *American journal of physics* 69.9 (2001), pp. 970–977.

Research on Peer Instruction

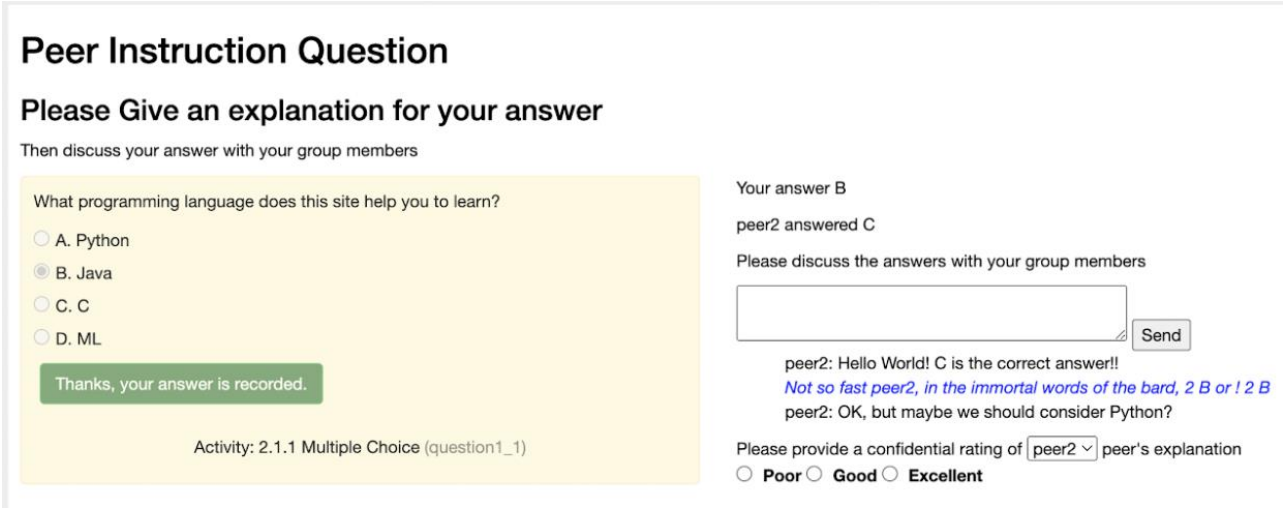
- Twice the learning gains over traditional lecture in Physics
- Increased student understanding and engagement
  - Physics, Biology, Math, and Computer Science
- In CS (both introductory and advanced) courses
  - Reduced failure rates
  - Improved retention
  - Increased final exam performance

Leo Porter and Beth Simon. "Retaining nearly one-third more majors with a trio of instructional best practices in CS1". In: *Proceeding of the 44th ACM technical symposium on Computer science education*. 2013, pp. 165–170.

Peer+ Tool in Runestone

Support for Peer Instruction questions

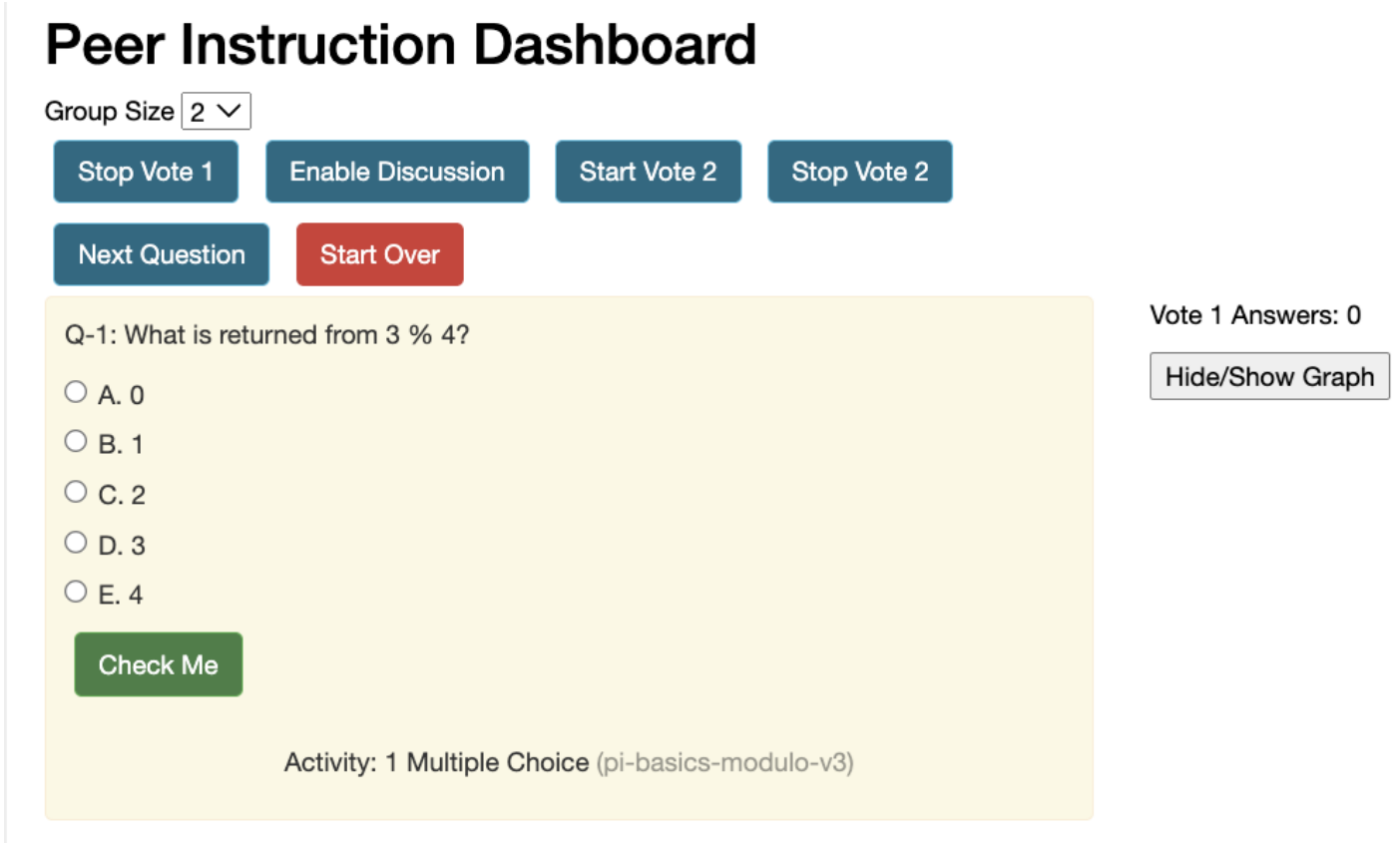
- Students answer in an ebook
- Can discuss question via chat



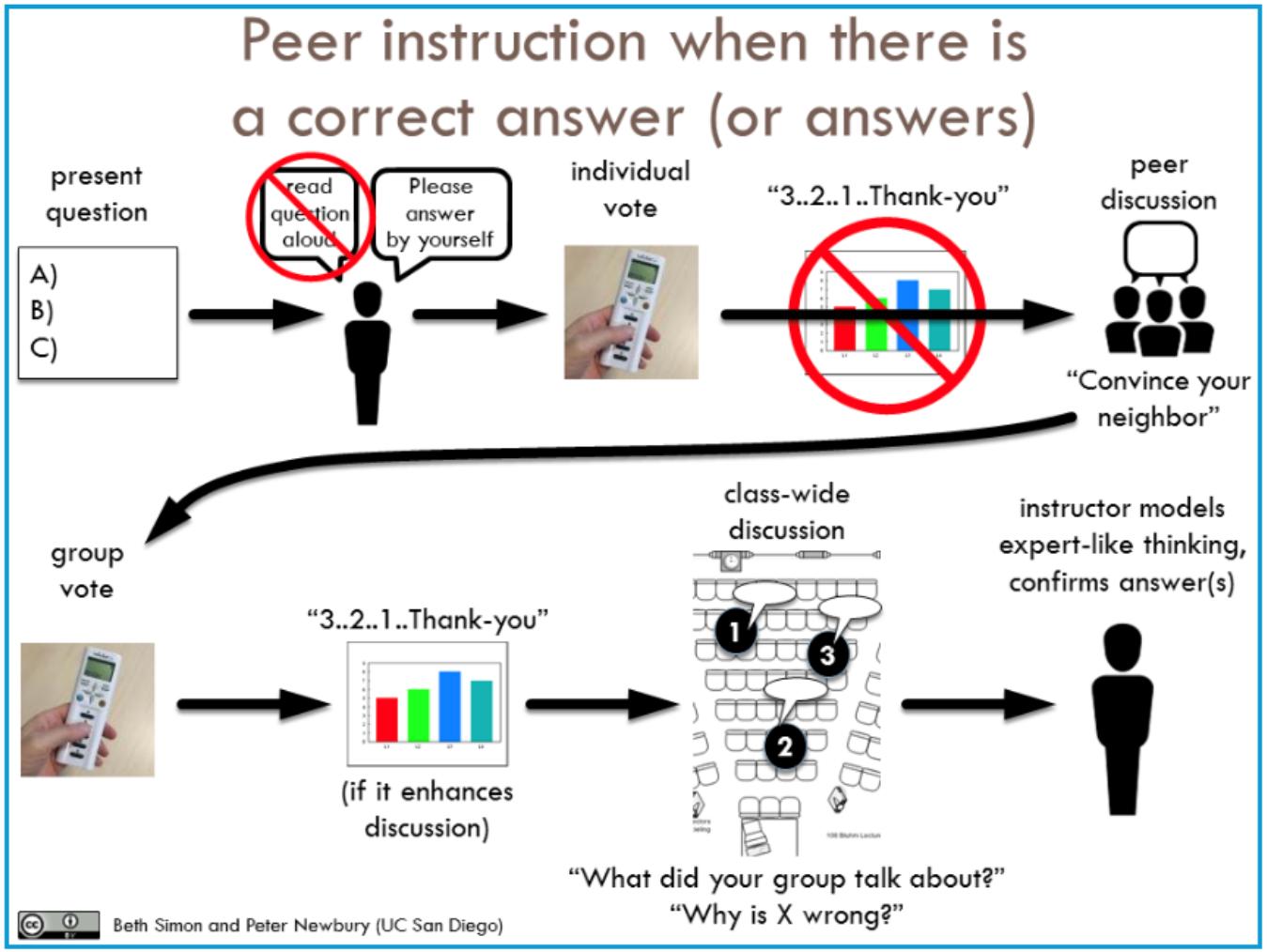
Peer+ Tool in Runestone

Support for Instructor

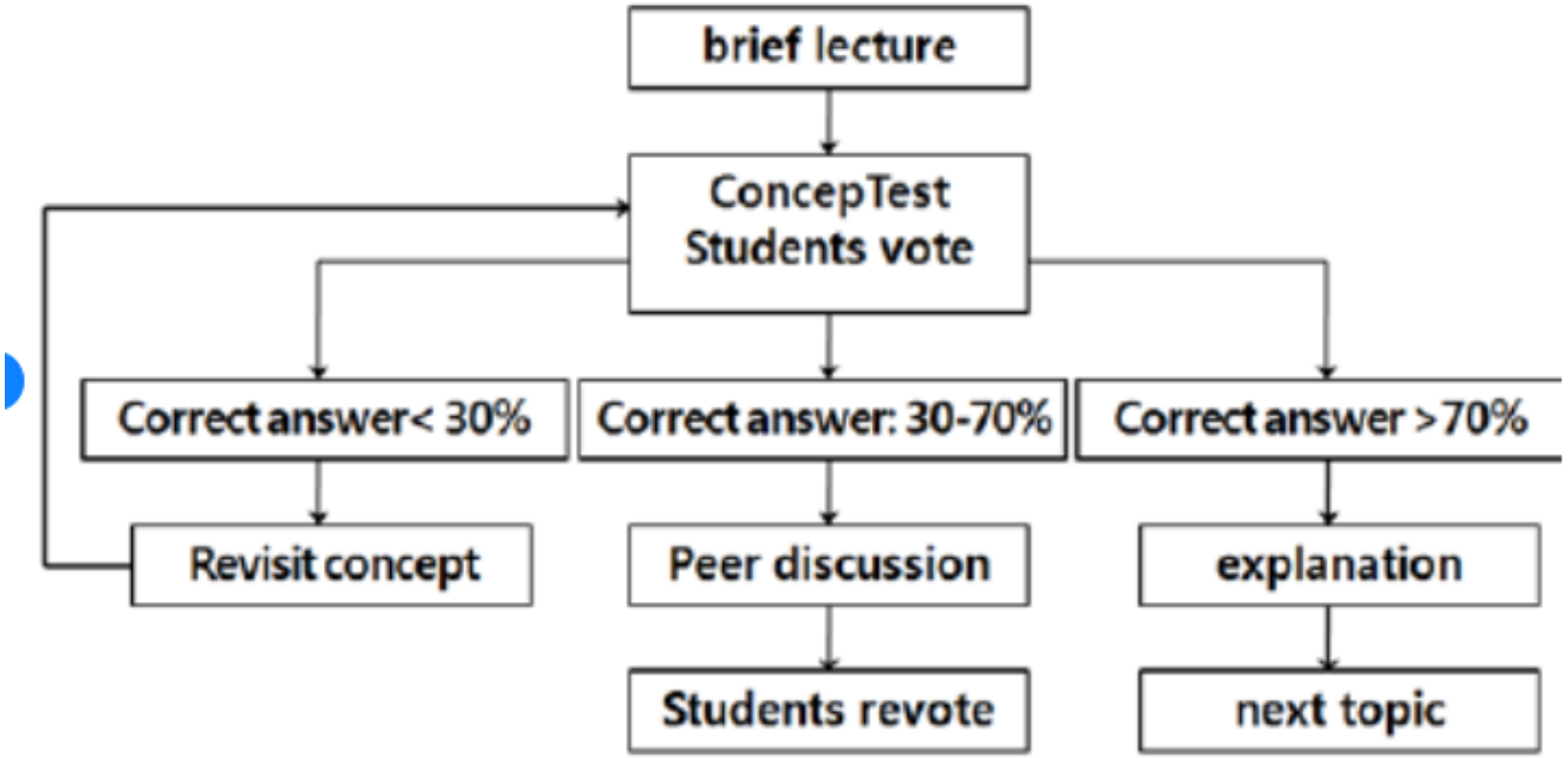
- Buttons for process
- The number of answers updates live
- Show the graph of the two votes



Peer Instruction Process



Hard Questions are Best



Finding Peer Instruction Questions

We added questions from several public question banks to Runestone

- CS Peer Instruction questions: http://peerinstruction4cs.com/
  - CS1 in Python
    - Dan Zingaro (dz-)
    - Cynthia Taylor (ctp-)
- PeerWise Question Bank
  - peerwiseqb
- Canterbury Question Bank
  - Canterburyqb

Will be adding questions in Java and C++ this summer

Summary

- Active Learning can improve student learning and engagement
  - Interactive ebooks
  - Parsons problems
  - Peer Instruction
- Ebooks can reduce instructor's workload
  - Automatic grading
  - Lots of practice problems