

Rethinking Engagement: Teaching and Learning with AI in CS Education

June 11, 2026

Margaret Ellis
Professor of Practice
Computer Science
Virginia Tech

Introduction

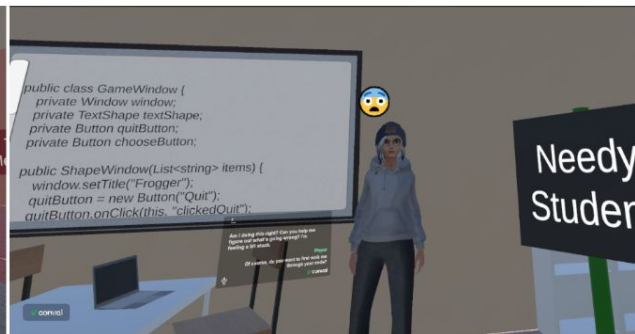
Margaret Ellis, Professor of Practice, Virginia Tech

- **VISTA** (Virtual Interactive Simulation for Teaching Assistants)
- **Peer+** (Peer Instruction using Runestone e-books)
- **DURA** (Lightweight Framework for LLM Course Integration)

VISTA

Virtual Interactive Simulation for Teaching Assistants

Arezoo Sarshartehrani, Margaret Ellis, Denis Gracanin, Brett Jones, Christine Julien



Background: MUSIC Model of Motivation

The MUSIC® Model —of Motivation—

eMpowerment

Usefulness

Success

Interest

Caring

Instructors need to ensure that:

- Students feel empowered by having autonomy over some aspects of their learning
- Students understand why the instruction is useful for their goals in life
- Students believe that they can succeed if they put forth the effort
- Students are interested in and enjoy the content and instructional activities
- Students believe that others in the learning environment care about their learning and them as a person

CS TA Training Project



Applying the MUSIC Model of Motivation to Support One-on-One Student Interactions

Interest & Usefulness

Create an enjoyable motivational climate that highlights usefulness

eMpowerment

Teach students how to problem-solve by themselves

M Strategies:

- scaffolding
- rubber ducking
- self-regulation
- active listening
- mirroring

Big Ideas

Success beliefs

Help students believe they can succeed

S Strategies:

- self-efficacy
- growth mindset
- success feedback

Caring

Show students you care about them

C Strategies:

- caring feedback
- foster belonging

© Brett D. Jones, 2025,
www.theMUSICmodel.com

Jones, B. D., Ellis, M., Kim, I. . (August 1, 2023 to July 31, 2025; NSF Award #IUSE-2315574). *Training Computer Science Teaching Assistants to Motivate Students*. National Science Foundation

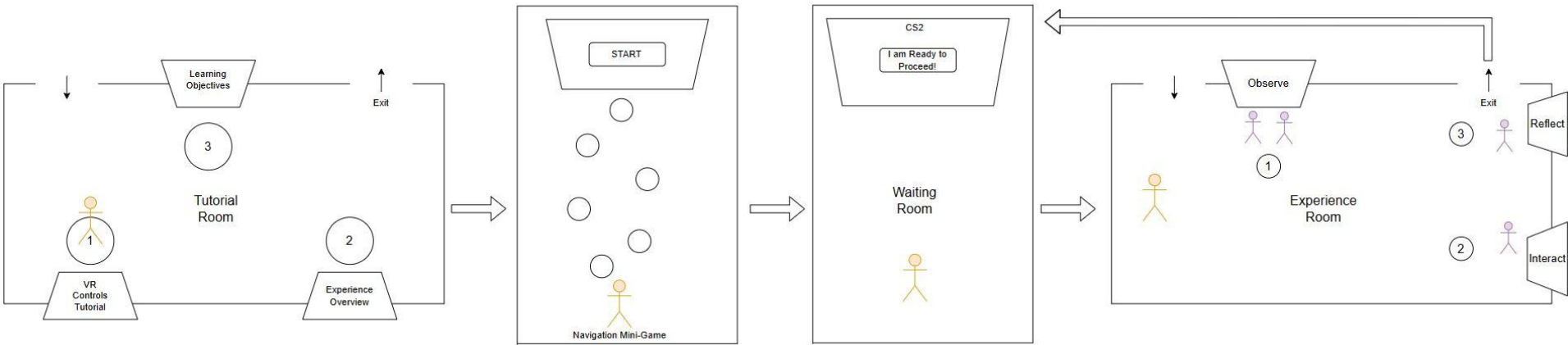


Observe → Interact → Reflect

Supports deliberate practice

- Observe: watch expert interaction
- Interact: talk to AI student(created with TAs)
 - Cooperative and Successful
 - Unconfident
 - Impatient
 - Needy
 - Demanding
- Reflect: AI mentor feedback

System Flow



Oh, I don't know — there are so many nested objects there.

Observe: Ideal
TA-Student
Interaction

```
public class GameWindow {  
    private Window window;  
    private TextShape textShape;  
    private Button quitButton;  
    private Button chooseButton;
```

```
public ShapeWindow(List<string> items) {  
    window.setTitle("Frogger");  
    quitButton = new Button("Quit");  
    quitButton.onClick(this. "clickedQuit");  
    ...  
}
```

Am I doing this right? Can you help me figure out what's going wrong? I'm feeling a bit stuck.

Player
Of course, do you want to first walk me through your code?

convai

convai



Needy Student

TA Mentor

by using more mirroring techniques, such as restating her concerns to ensure she feels heard and understood. While you guided her to explain her code, you could have provided partial hints to help her progress further, such as suggesting areas she might want to double-check, like the initialization of the 'window' object.



 convai



VT 2025 Pilot Study



- [Iterative development with faculty and TA feedback](#)
- Fall 2025: CS Department
 - 24 CS TAs
 - 2 sessions (4 weeks apart)
 - 5 persona interactions per session
- Key Results
 - Performance improved across sessions
 - Strong gains for difficult personas
 - Better confidence calibration
 - AI scoring aligns with human raters

Future Work

- Fall 2026: UIUC Graduate Academy
 - Comparison of learning gains and user experience among modalities
 - Immersive (VR)
 - Desktop
 - Chat
 - CS GTAs in coordination with Center for Innovation in Teaching & Learning
- Extend scenarios and course topic materials
- Enhance rubrics

Peer+

Peer Instruction Tool in Runestone

Testing the Effects of Peer Discussion, NSF iUSE Level II

Barbara Ericson, Associate Professor, University of Michigan

Jan Pearce, Professor and Chair, Berea College

Susan Rodger, Professor of Practice, Duke University

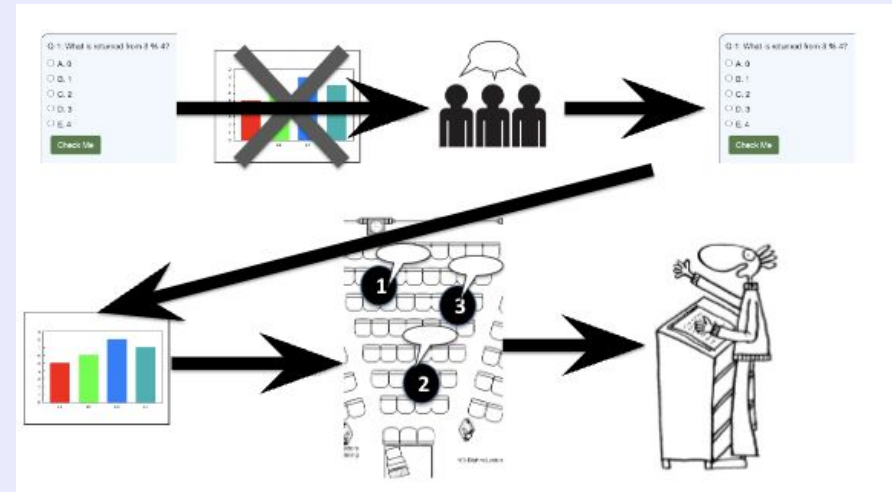
Yesenia Velasco, Lecturer, Duke University

Margaret Ellis, Professor of Practice, Virginia Tech

Brad Miller, Runestone

What is Peer Instruction?

- May have a pre-reading/quiz
- In lecture show a hard multiple-choice question
 - Students answer individually
 - Discuss their answer with peers
 - Students answer individually again
 - Instructor leads discussion of the question and answers



Runestone Peer+ Interface

The screenshot displays the Runestone Peer+ interface. At the top, a navigation bar includes tabs for 'Students', 'Progress', 'Monitor', 'Settings', 'Manage Materials', 'Feedback', and 'Help/Documentation'. Below this, the page title is 'Peer Instruction: PI-Fun-Str-Cond'. The main section is titled 'Question 1 of 2' with a 'Group Size' dropdown set to '3'. A progress bar at the top shows 'Vote 1 in Progress...' and buttons for 'Next Question' and 'Start Over'. The progress bar consists of four numbered steps: 1 (active, with a 'Stop Vote 1' button), 2 (with 'Enable in-person Chat' and 'Enable Text Chat' buttons), 3 (with a 'Start Vote 2' button), and 4 (with a 'Stop Vote 2' button). Below the progress bar, the question text is: 'Q-16: Which of the following sets x to 5 if x is set to the result from calling the function?'. A code block shows the following Python code:

```
def f1():  
    return 5  
def f2():  
    print(5)  
def f3():  
    return print(5)
```

 Below the code, there are five radio button options: 'A. $x = f1()$ ', 'B. $x = f2()$ ', 'C. $x = f3()$ ', 'D. Two of the above', and 'E. All of the above'. A 'Check Me' button is at the bottom. On the right side, it says 'Vote 1 Answers: 0 (0% Correct)' and a 'Hide/Show Graph' link. At the bottom left, it says 'Activity: 16 Multiple Choice (dz-lecture_5_9)'.

1. Starts first vote automatically (planning to add a button to start)
 - a. Constantly updates the number of votes cast
 - b. "Stop Vote 1" to end the first vote
2. Enable in-person chat or text chat with assigned groups
 - a. Can set the size of the group for the chat (near the top) before starting a vote
 - b. In text-chat see message count
3. Start vote 2
4. End vote 2
 - a. Show the histogram of the two votes
 - b. Discuss as a class

Peer Instruction Study

- Text-Chat in Peer+
 - Assign groups based on different answers
 - Maximize disagreement in groups
 - Supports verbal + text-chat discussion
 - Keeps same PI cycle
- Evaluating Text-Chat vs Verbal Discussion
- Multi-institutional study
- Focus: How Peer Instruction works

Synchronous Text Chat for Discussion

If you click the *"Enable text chat"* button students will get a text chat window.

- Discuss the answers with peers
- Tries to maximize the number of groups with members with different answers

The screenshot displays a web-based Peer Instruction interface. At the top, a navigation bar includes links for 'Introduction', 'VISTA', 'Peer+', 'DURA', and 'Questions'. The main content area is titled 'Peer Instruction: PI-FunctionsWithLists'. It features a question 'Q-1: What will the following code print?' with a code block containing a Python function 'mystery' and a list 'list1'. Below the code, four multiple-choice options (A, B, C, D) are listed, with option B selected. A 'Submit' button is present. To the right of the question, a chat window titled 'Your answer' shows 'seura answered D' and a prompt to 'Please discuss the answers with your group members'. The chat area contains the text 'The pop method will pop the element at a certain index.' and a 'Send' button. Below the chat, a rating section asks for a 'confidential rating of seura's peer's explanation' with options 'Poor', 'Good', and 'Excellent', where 'Excellent' is selected. At the bottom, a status bar indicates 'Your Answer has been recorded' and 'Activity: 1 Multiple Choice (listEx_MC11_v2)'. A footer note states: 'Please do not leave or refresh the page during a peer instruction session. Exception: If you should have seen a new question at this point then you may continue.'

Back to Win23-SI206

Peer Instruction: PI-FunctionsWithLists

Q-1: What will the following code print?

```
def mystery(num_list):  
    index = 0  
    while index < len(num_list):  
        num = num_list[index]  
        if num == 0:  
            num_list.pop(index)  
            index += 1  
  
list1 = [3, 0, 2, 0, 0]  
mystery(list1)  
print(list1)
```

☐ A. [3, 0, 2, 0, 0]
☒ B. [3, 0, 2]
☐ C. [3, 2]
☐ D. [3, 2, 0]

Your Answer has been recorded

Activity: 1 Multiple Choice (listEx_MC11_v2)

Your answer
seura answered D

Please discuss the answers with your group members

The pop method will pop the element at a certain index.

Please provide a confidential rating of seura's peer's explanation
☐ Poor ☐ Good ☒ Excellent

Please do not leave or refresh the page during a peer instruction session. **Exception:** If you should have seen a new question at this point then you may continue.

Key Findings

- Text-chat groups showed higher learning gains
- Students still prefer verbal discussion
- Preference $\neq$ effectiveness
- Design of interaction matters most

Ongoing and Future Work

- Large collections of searchable Peer+ questions available
- Updates to user interface for class and question building
- LLM based after lecture text-chat feature
- Incorporation of Study Clues in Runestone books
 - Creation of pools of Peer+ questions
 - LLM tutor

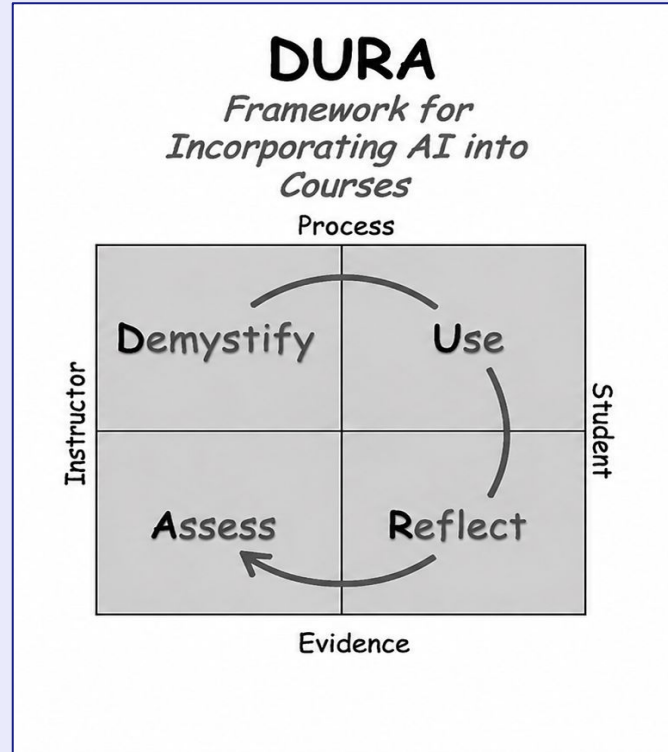
Runestone and Peer+ Workshops

Workshop Sign Up (led by Barbara Ericson):

<https://forms.gle/gtMJbnZPvi2gw2on6>.



1. Monday July 6th, 1:00pm - **Getting started with Runestone**
2. Tuesday July 7th, 1:00pm - **Instructor interface: creating a custom book for your students**
3. Wednesday July 8th, 1:00pm - **Instructor interface continued**
4. Thursday July 9th, 1:00pm - **Peer Instruction** *We will also show the new LLM based after lecture text-chat feature!*



*Margaret Ellis, Naren Ramakrishnan, Sehrish Basir Nizamani
Virginia Tech Computer Science*

DURA Framework Overview

- Demystify – Use – Reflect – Assess
- Integrating AI while maintaining critical thinking
- Lightweight framework for course redesign
- Supports ethical and effective LLM use

CS2104 Problem Solving Course Schedule

Spring 2025



LLM Topics Across Weeks(W) to Classwork (CW)
Homework (HW) and Lectures (L)

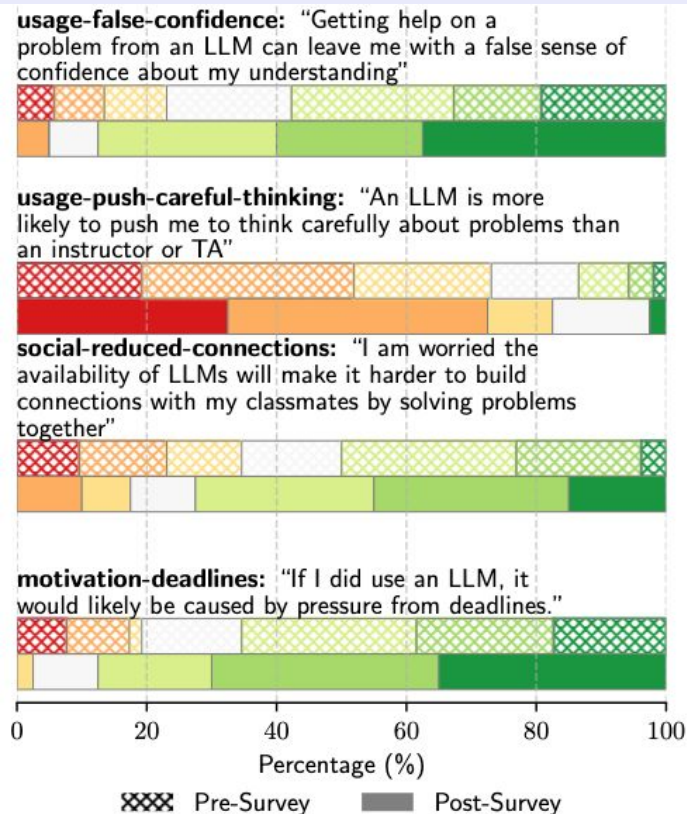
		<i>LLMs</i>		<i>Algorithms</i>		<i>Soft. Eng.</i>		<i>Networking</i>		<i>HCI</i>		<i>Data Science</i>	
Goals	Approaches	W1	W2	W3	W4	W5	W6	W8	W9	W10	W11	W12	W13
Demystify LLMs	How LLMs Work	CW	L					L					
	Current Resources	HW	CW	HW		HW							HW
LLM Usage Skills	Demos			L	L	L		L		CW	L		
	Semi-LLM-Resistant				HW	HW	HW		HW				HW
Reflective and Responsible LLM Use	Risks and Benefits		L			CW, L	L		L		L		
	Reflection on Use			L		HW, L	L						HW

CS2114 Data Structures and Software Design

<i>Component</i>	<i>Our CS2 Implementation</i>
Demystify	Demystify how LLMs work, how they can affect learning and performance.
Use	Use LLMs with guidance and attribution.
Reflect	Reflect on LLM use and problem-solving process.
Assess	Assess programming assignment skills through proctored assessments and allow retakes.

DURA in Physics

Collaborators: Jake O'Brien, Alma Robinson, John Simonetti



Pre-Planning

1. Define learning outcomes (with/without AI)
2. Establish AI usage policies
3. Consider student access to tools
4. Align with prerequisite & downstream courses

Demystify

- Teach how LLMs work
- Explain impact on learning
- Introduce real-world AI use
- Address ethics

Use

- Student roles from user to developer
- Tasks: coding, debugging, research
- Guided instruction and scaffolding
- Tool policies and constraints

Assess

- Assess with and without AI
- Assess frequently, allow retakes
- Consider: *explanation-based tasks, open-ended projects, process, design*

Reflect

- Supports metacognition
- Reflect on process and AI use
- Evaluate effectiveness and tradeoffs
- Consider ethical implications

CS2114 MUSIC Motivation Components

Term	Weeks	<i>n</i>	<i>N</i>	%	M	U	S	I	C
2025 Fall	15	235	442	53.2	4.6	5.0	4.9	4.4	5.3*
2024 Fall	15	329	597	55.1	4.6	5.0	5.0	4.2	4.7

Note. *n* = number of respondents; *N* = total enrollment; % = response rate. * $p < .001$

Thank you!

Questions?

Contact: Margaret Ellis,
maellis1@vt.edu

Reviewing Peer Instruction Results

Description:

Question 1

[Previous](#)[Next](#)

Question Effectiveness

Percent Correct Votes in Vote 1: 55%

Normalized Change

69%

Q-1: What does the following code evaluate to?

```
(1!=1) or (0 == 1/0)
```

Choose one

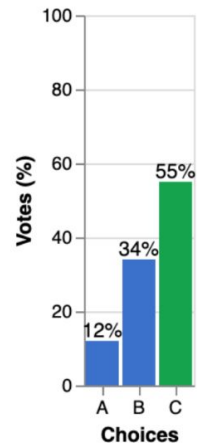
- ☐ A. True
- ☐ B. False
- ☒ C. Error

✓ Correct! The first expression (1!=1) evaluates to False, making it so we must evaluate the second expression. The second expression (0 == 1/0) causes a divide by zero error.

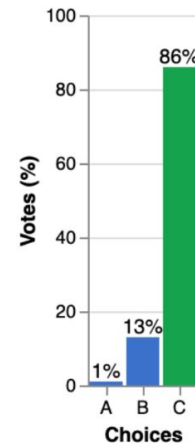
Activity: 1 Multiple Choice (duke101_sc_not_equal_divide_using_or)

[Show Question in the Book](#)

Vote 1 Results (n=161)



Vote 2 Results (n=156)



Takeaways

- - XR + AI supports interpersonal skill training
- - Enables scalable feedback
- - Improves metacognitive awareness

- Future: more personas, adaptive systems

Memorable Student Quote

“I don’t trust AI as much at all anymore, I didn’t know they were probability machines”

System Overview

- - XR virtual office hours (Meta Quest 2)
- - AI student avatars (LLM-driven)
- - Real-time dialogue + feedback
- 16-item rubric used for evaluation
 - - Empowerment (scaffolding)
 - - Success (growth mindset)
 - - Caring (empathy)
- Goal: Develop interpersonal + pedagogical skills