



Microteaching: Binary Heaps, Side-Channel Attacks, Equitable Grading, Java Classes, Loops, and 3D Java

Colleen M. Lewis

Computer Science Department
Univ. of Illinois Urbana-Champaign
Urbana, IL, USA
ColleenL@illinois.edu

Maria Camarena

Maywood Center for Enriched Studies
Los Angeles Unified School District
Maywood, CA, USA
MariaCamarena@mfala.org

Cynthia Bailey Lee

Computer Science Department
Stanford University
Stanford, CA, USA
cbl@stanford.edu

Manuel Hernandez

Mathematics Department
Bell Gardens High School, MUSD
Math for America - Los Angeles
Bell Gardens, CA, USA

Hernandez_Manuel@montebello.k12.ca.us

Adam Blank

Computing & Mathematical Sciences
California Institute of Technology
Pasadena, CA, USA
blank@caltech.edu

Frank Vahid

Computer Science & Engineering
University of California, Riverside
Riverside, CA, USA
vahid@cs.ucr.edu

ABSTRACT

This microteaching session is like Nifty Assignments for instruction. Instead of having the presenters just talk about their teaching, they will simulate how they would actually teach something. Covering a range of topics and grade levels, six educators will demo how they would teach a specific topic. To help identify the pedagogical practices that cut across grade bands and topics, the moderator, Colleen Lewis, will describe how their pedagogical practices connect with education research. The goal of the session is to inspire SIGCSE attendees by highlighting innovative instruction by exceptional educators. Attendees can adopt the content and/or pedagogical practices from each microteaching example.

KEYWORDS

pedagogy; pedagogical content knowledge; innovating teaching

ACM Reference Format:

Colleen M. Lewis, Cynthia Bailey Lee, Adam Blank, Maria Camarena, Manuel Hernandez, and Frank Vahid. 2024. Microteaching: Binary Heaps, Side-Channel Attacks, Equitable Grading, Java Classes, Loops, and 3D Java. In *Proceedings of the 55th ACM Technical Symposium on Computer Science Education V. 2 (SIGCSE 2024), March 20–23, 2024, Portland, OR, USA*. ACM, New York, NY, USA, 2 pages. <https://doi.org/10.1145/3626253.3633406>

1 INTRODUCTION AND STRUCTURE

SIGCSE is packed with teaching insights and inspiration. However, we get these insights and inspiration from hearing our colleagues talk about their teaching. Why not *watch* them teach? This session does exactly that!

The goal of the session is to model general pedagogical techniques and pedagogical techniques applicable to teaching specific content. Each of six exceptional educators will teach the audience

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

SIGCSE 2024, March 20–23, 2024, Portland, OR, USA

© 2024 Copyright held by the owner/author(s).

ACM ISBN 979-8-4007-0424-6/24/03.

<https://doi.org/10.1145/3626253.3633406>

something (7 minutes each). After each teaching demonstration, Colleen Lewis will draw the attention of the audience to particular pedagogical moves that the instruction included (1 minute). The remaining time will be used for attendees to ask follow-up questions to the presenters of their choice. The structure of a special session provides the flexibility to have six, fast-passed demos, brief comments highlighting pedagogical practices, and ample time for small-group discussion with individual presenters.

These educators and educators in the past [4–7] were selected because they are inspiring, engaging, and effective teachers. Our intended audience is CS educators who want practical and inspiring ideas to take back into their classroom. We intend for attendees to benefit by taking content and/or pedagogical moves from this session directly back to their teaching.

2 EXCEPTIONAL EDUCATORS

2.1 Binary Heap Insert - Cynthia Bailey Lee

This teaching demo will show how I use Peer Instruction (PI) and live coding together in class. Peer Instruction questions both build conceptual knowledge of data structures, and directly support live coding by engaging all students in authoring key pieces of code. The first activity uses a classic “how many of these are correct?” PI question structure, which intentionally leaves unstated *which* are correct, as a means of provoking students to initiate discussion of exactly that when they turn to each other. To support Peer Instruction questions, we will use the Poll Everywhere website/app (no install or account is needed).

Cynthia Bailey Lee is a Senior Lecturer in the Computer Science Department at Stanford. She specializes in team-based and question-based course design, and founded peerinstruction4cs.org to support educators in flipping their computer science classrooms using peer instruction.

2.2 Side-Channel Attacks - Adam Blank

In this teaching demo, I will demonstrate how a topic traditionally taught in a security course can help unify and demystify typical topics in an “introduction to systems” course. I will use an active learning approach to engage and involve the audience as well as

a small programming exercise to demonstrate the applicability of the topic.

Adam Blank is a Teaching Professor of Computing and Mathematical Sciences at Caltech. Their teaching approach involves developing new technologies and techniques to enhance the learning of computer science students. They experiment with new techniques and technologies involving technology, human computation and collaboration to improve the classroom experience. They are especially interested in broadening participation in computing via their teaching methods.

2.3 Equitable Grading - Maria Camarena

To foster equity, I implement standards-based grading [1, 8]. In this teaching demo, I will show how I introduce and explain my grading system to my students. Most students are unfamiliar with standards-based grading, and helping them understand how the system helps them understand my commitment to their learning and my expectations for them. This demo aims to provide valuable insights into practical strategies and perspectives for promoting a fair and transparent grading process.

Maria Camarena is a CS teacher at Maywood Center for Enriched Studies in Los Angeles Unified School District. Over the last five years, Maria developed a CS pathway in her school that teaches students in grades 6 through 12. Maria serves as a Computer Science Teachers Association (CSTA) Equity Fellow. For her efforts to include students who identify as women, Hispanic, and/or Latinx, Maria received the 2020 Teaching Excellence Award from CSTA and the Infosys Foundation, and the 2022 National Aspirations in Computing Educator Award from the National Center for Women and Information Technology (NCWIT).

2.4 Java Classes: Class Structure and Writing Constructors - Manuel Hernandez

Object-oriented programming depends on the ability to take common and complex ideas and turn them into objects. Halfway through the AP Computer Science A (a Java course), students are formally introduced to the structure of a class. In the lesson, we will look at a basic structure, apply the basic structure to common objects, and get participants to pair code to create classes and constructors.

Manuel Hernandez has been a high school mathematics teacher for 17 years, and a member of the Master Teacher Cohort in Math for America Los Angeles for 8 years. He was involved in starting a Computer Science program at Bell Gardens High School, in Bell Gardens, California in 2019 and has taught AP Computer Science A for 3 years. He is currently teaching Computer Science Discoveries.

2.5 Loops - Frank Vahid

With loops, we can read a list of values and do things like print only the negative values, but how can we do something like printing the list in reverse? We need to store the list, and that's why we need vectors. We'll show how to append values onto a vector, and how to traverse and print the vector's elements, first forward, then in reverse. We'll point out common mistakes, and maybe even make some of those mistakes along the way! With vectors, programs can be much more powerful.

Frank Vahid has been a Professor of Computer Science at the University of California, Riverside, for 30 years, is co-founder and chief learning officer at zyBooks, and is founder and president of CollegeStudentAdvocates.org. He shifted his research focus to CS education just over a decade ago, passionate about helping more wonderful CS majors achieve their degree goals.

2.6 3D Java - Colleen M. Lewis

In this teaching demo, I will illustrate how I use physical objects to help students understand aliasing in Java using 3D representations [3]. In particular, I use a strategy known as Concreteness Fading [2] wherein I begin with a concrete, physical representation and systematically transition to a more abstract representation.

Colleen Lewis is an Associate Professor of CS at the University of Illinois at Urbana-Champaign. Lewis was previously the McGregor-Girard Associate Professor of CS at Harvey Mudd College. She researches equitable and efficient teaching practices and curates CSTeachingTips.org, a NSF-sponsored project for disseminating effective CS teaching practices.

Colleen also serves as the moderator for the session. This role involves connecting the pedagogical practices in all of the teaching demonstrations with education research. This builds on her training; at the University of California, Berkeley, Colleen completed a PhD in science and mathematics education.

ACKNOWLEDGMENTS

CSTeachingTips.org is funded by the NSF (1821136 and 2144249).

REFERENCES

- [1] Joe Feldman. 2024. *Grading for equity: what it is, why it matters, and how it can transform schools and classrooms* (second edition ed.). Corwin, Thousand Oaks, California.
- [2] Emily R. Fyfe, Nicole M. McNeil, Ji Y. Son, and Robert L. Goldstone. 2014. Concreteness Fading in Mathematics and Science Instruction: a Systematic Review. *Educational Psychology Review* 26, 1 (March 2014), 9–25. <https://doi.org/10.1007/s10648-014-9249-3>
- [3] Colleen M. Lewis. 2021. Physical Java Memory Models: A Notional Machine. In *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education (SIGCSE '21)*. Association for Computing Machinery, New York, NY, USA, 383–389. <https://doi.org/10.1145/3408877.3432477>
- [4] Colleen M. Lewis, Leslie Aaronson, Eric Allatta, Zachary Dodds, Jeffrey Forbes, Kyla McMullen, and Mehran Sahami. 2018. Five Slides About: Abstraction, Arrays, Uncomputability, Networks, Digital Portfolios, and the CS Principles Explore Performance Task. In *SIGCSE Proceedings*. ACM, 269–270.
- [5] Colleen M. Lewis, Christine Bassem, Jason M. Grant, Felienne Hermans, Angel Kuo, Art Lopez, and Beth Trushkowsky. 2023. Microteaching: Ad-Hoc Networks, Binary Heaps, Variables in Hedy, Loops, Lists, and Data Storage. In *Proceedings of the 54th ACM Technical Symposium on Computer Science Education V. 2 (SIGCSE 2023)*. Association for Computing Machinery, New York, NY, USA, 1218–1219. <https://doi.org/10.1145/3545947.3569593>
- [6] Colleen M. Lewis, Kathi Fisler, Jenny Hinz, David J. Malan, Joshua E. Paley, Manuel A. Pérez-Quinones, and Shikha Singh. 2021. Microteaching: Semantics, Definition of a Computer, Running Times, Fractal Trees, Classes as Encapsulation, and P vs NP. In *SIGCSE Proceedings*. ACM, 766–767. <https://doi.org/10.1145/3408877.3432582>
- [7] Colleen M. Lewis, Daniel D. Garcia, Helen H. Hu, Saber Khan, Nigamanth Sridhar, Bryan Twarek, and Chinma Uche. 2019. Microteaching: Recursion, Coding Style, Creative Coding, Inheritance and Polymorphism, Loops, and the Internet. In *SIGCSE Proceedings*. ACM, 962–963.
- [8] Peter Liljedahl. 2021. *Building thinking classrooms in mathematics, grades K-12: 14 teaching practices for enhancing learning*. Corwin, Thousand Oaks, California.