

EDPS 49100: Methods of Teaching Computer Science

Fall 2012, Taught Online, 3 credits

- *What are effective techniques for teaching computational thinking and programming?*
- *How can you increase your skills as a computer science (CS) teaching assistant?*
- *How will the ability to teach CS make you more competitive in the job market?*

This course will answer these and other questions by presenting effective methods and tools for teaching computer science at the secondary and post-secondary levels. The primary target audience for the course is secondary teaching majors, particularly those in the STEM disciplines preparing for student teaching. However anyone interested in teaching computer science, including CS majors, CS teaching assistants, and in-service computing teachers, will also benefit from this course.

Prerequisites

- Knowledge of Java (CS 18000 or equivalent) and experience in a second programming language (e.g., CS 15900, CS 17700, CS 24000), or consent of the instructor.

What you will learn

- You will acquire pedagogical content knowledge (i.e., how to teach CS concepts), as well as practical experience, to be an effective CS teacher at the secondary school level.
- You will be exposed to the latest thoughts in how to teach computing skills. Even if teaching computer science or programming is not your long-term objective, the material you will learn will make you a better and more effective instructor overall.
- Assignments, projects, and exams are quite different in computer science because of the lab environment and technology requirements. You will learn how to design and grade effective programming assignments, including group projects.
- The course will also give you skills that can help you in the work place, for instance leading a team of programmers and developers.

Course Topics

- **Introduction & Pedagogy**
 - the state of CS education in high schools
 - pre-AP curricula (ISTE/CSTA vs state standards)
 - review of general pedagogy skills (i.e., block II)
 - apprentice-based and problem-based learning
- **General Methods for CS**
 - lesson planning and reflection
 - challenges of teaching programming
 - managing a computer lab environment
 - teaching unit testing and debugging
- **Methods for Tough Topics**
 - kinesthetic activities and role playing

- variables and flow of execution
- arrays, sorting and memory management
- dynamic data structures and recursion
- effective use of visualization
- **Assignments & Grading**
 - designing projects and grading rubrics
 - guidelines for successful team projects
 - designing and administering exams
- **AP CS & Resources**
 - how to run the AP CS course
 - objects first and other OOP approaches
 - overview of the GridWorld case study
 - upcoming AP CS Principles course
 - professional organizations (CSTA, ACM)
- **Microteaching & Peer Feedback**
 - prepare a lesson plan and teach it to peers
 - instruct and tutor students in CS 17700 labs
 - observe CS classrooms at local high schools

Organization

This 3-credit course runs with a 2-hour lecture and a 2-hour lab each week. The lectures survey the state of computer science education and the pedagogical skills to teach computational concepts and programming. The labs will (1) give students the opportunity to practice teaching methods in small groups, and (2) provide hands-on training of various software tools for learning CS and programming (e.g., Alice, BlueJ, Greenfoot, GridWorld, Jeroo, Scratch).

Course requirements include written assignments (e.g., lesson plans and reflection journals), lab activities, field experience (classroom observations both at Purdue and local high schools), microteaching and peer feedback, and active participation in class.

The course will be co-taught by Dr. Aman Yadav from Educational Studies and Dr. Tim Korb from Computer Science. For more information visit <http://cs4edu.cs.purdue.edu/courses>.