

# Code in Place 2023: Understanding learning and teaching at scale through a massive global classroom

Ali Malik  
malikali@stanford.edu  
Stanford University  
California, USA

Thomas Jefferson  
tjj@stanford.edu  
Stanford University  
California, USA

Patricia Wei  
patwei@stanford.edu  
Stanford University  
California, USA

Julie Zelenski  
zelenski@cs.stanford.edu  
Stanford University  
California, USA

Juliette Woodrow  
jwoodrow@cs.stanford.edu  
Stanford University  
California, USA

Miranda Li  
mirandal@stanford.edu  
Stanford University  
California, USA

Dora Demszky  
ddemszky@stanford.edu  
Stanford University  
California, USA

Mehran Sahami  
sahami@cs.stanford.edu  
Stanford University  
California, USA

Brahm Capoor  
brahmcapoor@gmail.com  
Stanford University  
California, USA

Sierra Wang  
sierraw@stanford.edu  
Stanford University  
California, USA

Jennifer Langer-Osuna  
jmlo@stanford.edu  
Stanford University  
California, USA

Chris Piech  
piech@cs.stanford.edu  
Stanford University  
California, USA

## ABSTRACT

Code in Place 2023, was a large-scale, free, online, introductory programming course based on Stanford University’s CS106A curriculum. This paper presents an overview of some useful data collected from the course that could help answer important questions about learning and teaching at scale. It also attempts to document and credit a list of core contributors responsible for creating Code in Place.

This iteration of the course had 8,762 students across 146 countries and 594 volunteer teachers from 66 countries engaging in a six-week educational journey starting in April 2023. Our dataset offers a thorough examination of the student learning experience, encompassing student code trajectories, course progression analytics, assignment submissions, lecture engagement, and detailed feedback from the diagnostic exams. Additionally, it provides insights into the volunteer teacher experience through metrics of their interactions with the course and students, as well as transcripts from weekly discussion sections.

While direct access to the data is restricted due to privacy, this paper outlines the dataset’s structure and potential utility for researchers and educators who are allowed access.

## 1 CODE IN PLACE

Code in Place is a free introductory programming course using the Python language, based on material from the first half of Stanford University’s established intro course, CS106A. Code in Place 2023 was the third iteration of this course and lasted for six weeks starting in April of 2023. In Code in Place 2023, there were 8,762 students from 146 countries and 594 volunteer teachers from 66 countries.

## 2 OVERVIEW OF THE DATA

Our dataset provides a snapshot of the student learning experience throughout this course. This includes student code trajectories for each assignment, detailed insight into student course progress such as lesson completions, assignment submissions,

and lecture viewings. Data also encompass unit test results for all code submissions and style scores for fully functional code submissions. Additionally, we collected extensive, detailed, and individualized feedback from the diagnostic exam that students completed.

We also gathered data on the volunteer teacher experience, including metrics on how teachers interact with the course and its students. Additionally, we have transcripts of their weekly discussion sections.

Access to this data is limited to protect the privacy of our students and staff. Please email [codeinplace@gmail.com](mailto:codeinplace@gmail.com) with any questions.

## 3 CORE CONTRIBUTORS

Code in place was the product of hard work by a massive team. Here, we present a non-exhaustive list of core contributors.

### 3.1 Instructors

**Chris Piech:** founded Code in Place and lead the creation of Code in Place 2023. He also served as an instructor, led the software development team to create the entire course platform, and advised on all of research projects.

**Mehran Sahami:** founded Code in Place and served as one of the instructors for Code in Place 2023.

**Julie Zelenski:** served as one of the instructors for Code in Place 2023 and advised on all aspects of volunteer TA hiring and training.

**Ali Malik:** founded Code in Place and served as an instructor for Code in Place 2023. He helped manage all aspects of the course and was tech lead for the course platform.

**Juliette Woodrow:** served as one of the instructors for Code in Place 2023, leading the creation of weekly section material and managing all aspects of the course. She also delivered guest lectures and served as a tech lead for the course platform.

**Brahm Capoor:** founded Code in Place and served as an instructor for Code in Place 2023. He taught weekly sections for the

course and section leaders, guest lectured, and help steer course pedagogy and policy.

### 3.2 Course Infrastructure

**Brahm Capoor:** also served as lead developer for Code in Place 2023. He built and maintained the course's discussion forums and architected parts of its foundational back-end infrastructure.

**Thomas Jefferson:** was a lead developer for Code in Place 2023. Most significantly, he developed the system that allowed users to run, debug, and unit test their code in the browser. He also worked on aspects of the browser based IDE including the terminal and graphics library.

**Joseph Tey:** worked on developing the IDE user interface, including a responsive mobile editor mode. He also built the virtual, cloud based, file system, and ability to publish student programs.

**Jason Ford:** set up back-end infrastructure such as the Yjs webrtc signaling server and Jitsi video storage.

**Aaron Broder and Matt Peng:** helped with many software tasks across the course website.

### 3.3 Teacher Training

An essential feature of Code in Place is its dual nature as a learning environment for both students and teachers. While students acquire new skills, volunteer teachers simultaneously gain teaching experience and receive robust support to enhance their pedagogic abilities. In this iteration of Code in Place, we had a multifaceted approach to training teachers.

**Jennifer Langer-Osuna:** led a project to develop asynchronous training modules for volunteer instructors, with significant contributions from Jim Malamut and Miroslav Suzara, among others on her team.

**Dora Demszky** and her team implemented a tool that used AI to analyze section transcripts and provide high quality feedback to the volunteer teachers each week.

**Julia Markel** and her team created and implemented a tool for volunteer teachers to practice teaching interactions with simulated students.

**Julie Zelenski** directed the creation of synchronous training modules for each iteration of Code in Place, working with a team to produce comprehensive materials and strategies that bolstered the skills of novice volunteer teachers within the program.

### 3.4 Data for Research

**Juliette Woodrow:** developed and deployed resources to record essential course metrics for assessing student achievement. Notably, she co-developed an automated grading tool that graded according to a rubric and delivered detailed personalized feedback on the diagnostic exam for thousands of students.

**Ali Malik:** Created much of the backend to monitor and track student outcomes, course progress trajectories, and timeseries of student interactions such as engagement with parts of the course.

**Sierra Wang:** set up infrastructure to log program runs within the course Integrated Development Environment (IDE), allowing for a comprehensive timeseries of user code each time it is executed.

### 3.5 Course Staff

The entire course was only possible due to an incredible course staff that was in charge of organising the volunteer teachers, offering necessary assistance, as well as actively contributing to the development, refinement, and refreshment of the course materials.

**Miranda Li:** was the lead TA for the course and managed the entire team of head TAs and volunteer teachers. She also developed the innovative story feature to help foster community.

**Patricia Wei:** was a co-lead TA for the course along with Miranda, helping manage groups of TAs and assist with course logistics.

**The Head TA Team:** was essential for helping the entire course run smoothly. Head TAs: Joseph Lee, Akinbowale Akin-Taylor, Ryan Lian, Nadin Tamer, Logan Marquis, Haven Whitney, Catherine (Cat) Huang, Kayla Patterson, Sejal Jain, Sasha Ronaghi, Inessa Royt, Je-Mé Kruger-Baartjes, Colton Botta, Cameron Mohne, Julia Rose Chin, Nancy Hoang, Sanjana Chakravarty, Caroline Tran, Areeba Khan, Gonzalo Comas, Linda Tong, Ty Baker.

**Administrative team:** The Course Administrative Team managed the logistics of the course, ensuring that everything ran smoothly. The administrative team: Amelie Byun, Elizabeth Cao, Matthew Harvey, John Cho, Swati Daub.

### ACKNOWLEDGEMENTS

Our sincere gratitude is extended to the Carina Foundation for providing the support that enabled this research. Additionally, we would like to thank the numerous individuals whose dedication and hard work laid the groundwork for the initial two iterations of Code in Place. The advancements presented in Code in Place 3 are intrinsically a testament to these earlier efforts. We would particularly like to acknowledge Michael Chang, Kylie Jue, Kate Rydberg, Moussa Doumbouya, Lisa Einstein, Cathy Zhang, and Yosefa Gilion for their exceptional contributions.