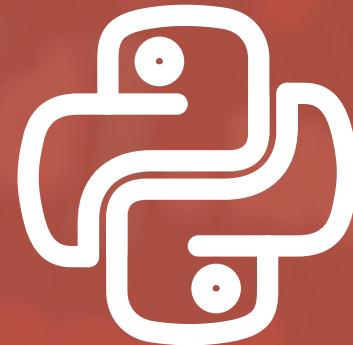


Code in Place



CS49

MATTHEW RASCOFF
VICE PROVOST FOR DIGITAL EDUCATION
STANFORD UNIVERSITY

SARAH KHAN
CODE IN PLACE SECTION LEAD
FOOTHILL COLLEGE

AGENDA



1

**What is Code
in Place?**

2

**Code in
Place Comes
to Foothill
College**

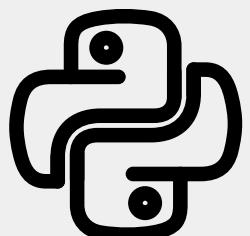
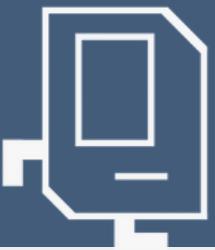
3

**Section
Leaders**

4

**Section is in
Session!**

CiP OVERVIEW



Remote Python Programming

Stanford's Code in Place is a free online course offering an introduction to Python programming, covering the first half of Stanford's CS106A.



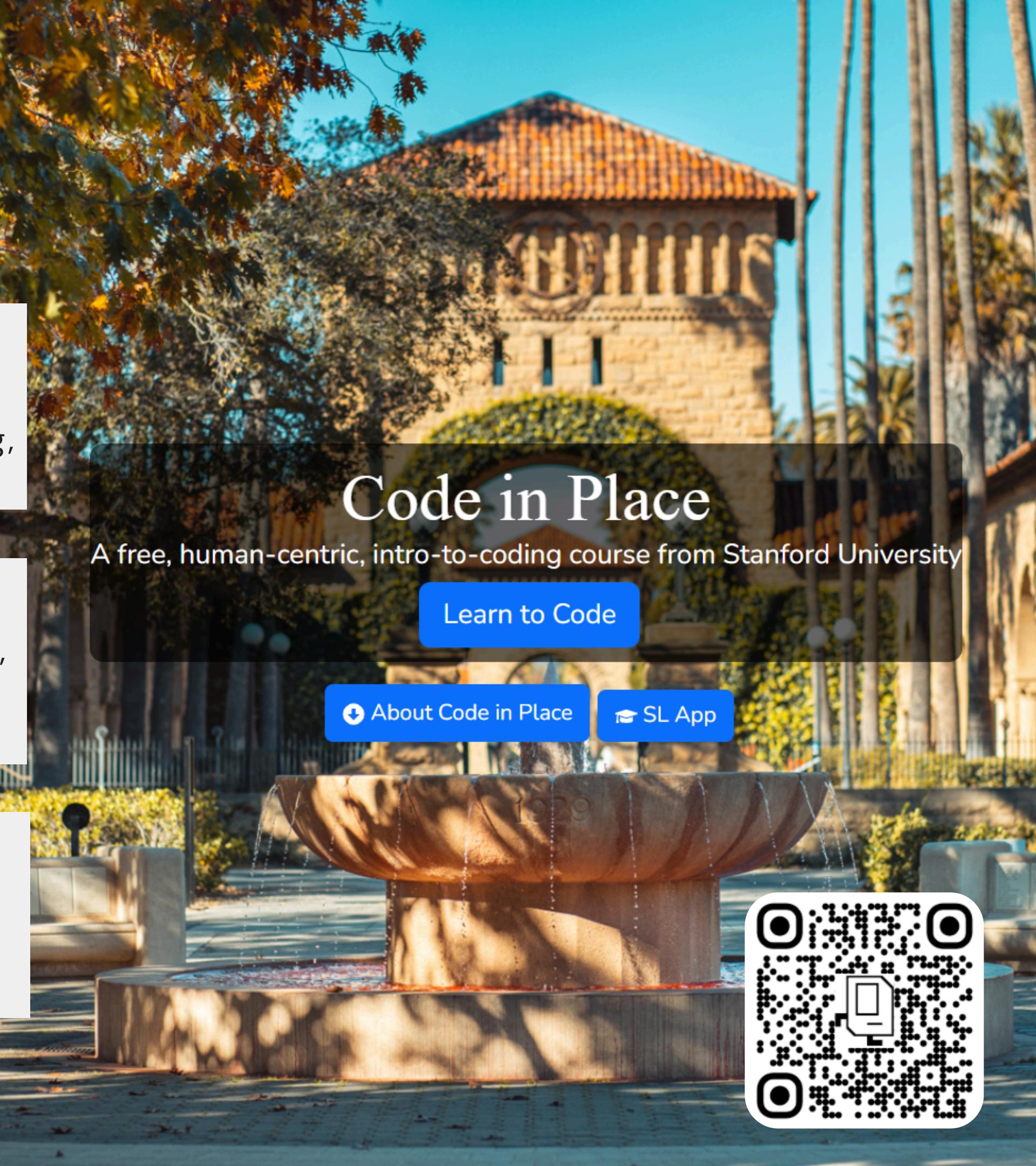
Scalable Education Model

With over 30,000 students globally since 2020, the program combines recorded lectures and guided learning sections.



Human-Centered Approach

Sections led by volunteer instructors ensure personalized guidance and enhanced engagement.



Code in Place

A free, human-centric, intro-to-coding course from Stanford University

[Learn to Code](#)

[About Code in Place](#)

[SL App](#)

PAST OFFERINGS

3,000

section leaders teach

30,000

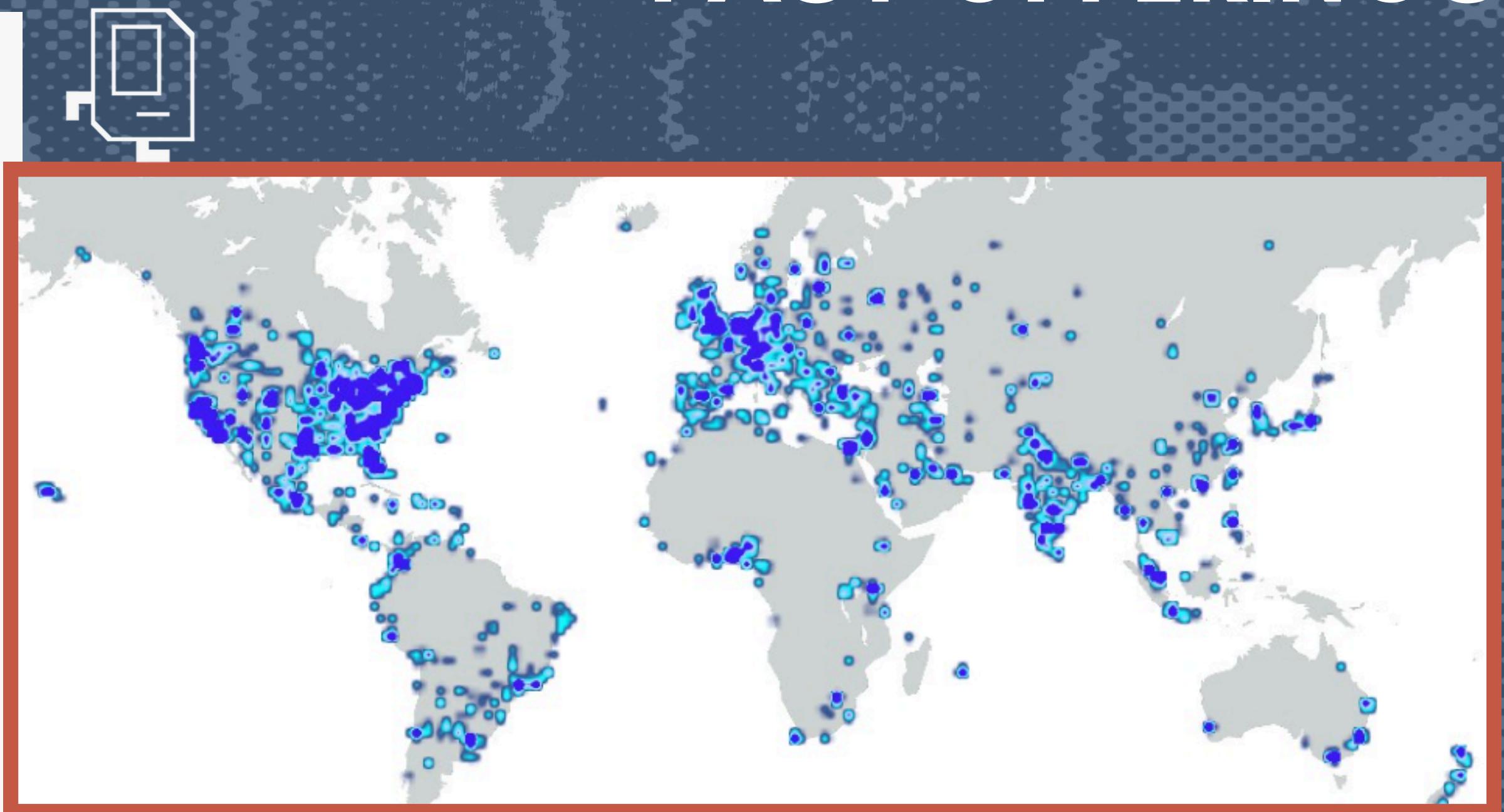
students

1:10

student to section leader ratio

1/2

of Stanford's CS106A



Map of students from Code in Place 2020, the first two offerings. The course had 10,000 students in the first offering and 12,000 in the second

The screenshot shows the course home page for Stanford & Foothill CS 49. The top navigation bar includes 'Stanford & Foothill CS 49' and 'Course Home'. The left sidebar has links for 'Home', 'Lessons', 'Forum', 'Code', 'Section', and 'Stories'. The main content area features two weeks of curriculum:

- Wk. 01 - Welcome**: Includes 'Course Commitment', 'Share Your Story', 'Lesson: Welcome to Karel', 'Reading: Karel ch. 1-3', 'Warmup', 'Karel's Home', 'Jigsaw Karel', and 'Open Playground (optional)'.
- Wk. 02 - Karel Control Flow**: Includes 'Lesson: Control Flow', 'Reading: Karel ch. 5-7', 'Piles', 'Beeper path', and 'Conditional Turn'.

The right sidebar displays 'YOUR PROFILE' (Sarah Khan), 'INSTRUCTORS' (Lane, Chris and Mehran), and a world map. A question at the bottom asks, "Are we the class in the history of the world with the most teachers?"

IDE | Hello Learning Lab! window: Shows the 'main.py' file with code for Karel, a 'Docs' panel with Karel language documentation, and a 'World' panel showing a 10x10 grid for Karel's world.

Feature Rich

The platform provides ample modularity with fast tailored feature releases.



Easy to Navigate

User Interface provides clear curriculum sequence and easy access to core course components.

Embedded Coding Environment

Students quickly begin coding without the pain point of environment and language installation.

PLATFORM

COMMUNITY

SL DISCUSSION VIEW

STUDENT DISCUSSION VIEW

Stanford & Foothill CS 49

Section Forum

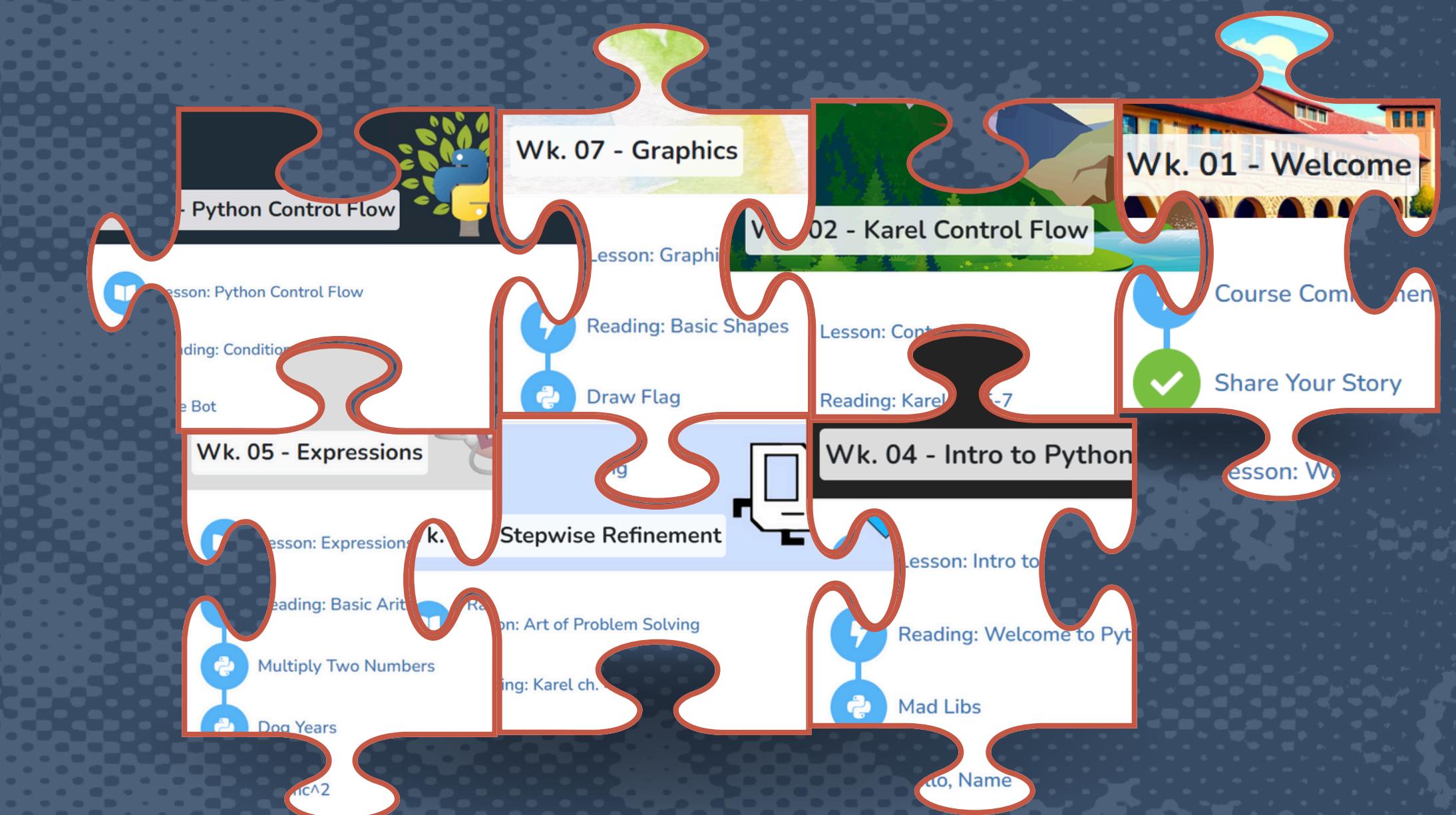
Good morning!

Select a post on the left to get started, or **New Post**

Discussion Rich

Discussion forums facilitate teaching team engagement among each other and with students. Section leads share resources and ongoing collaboration.

MODULARITY



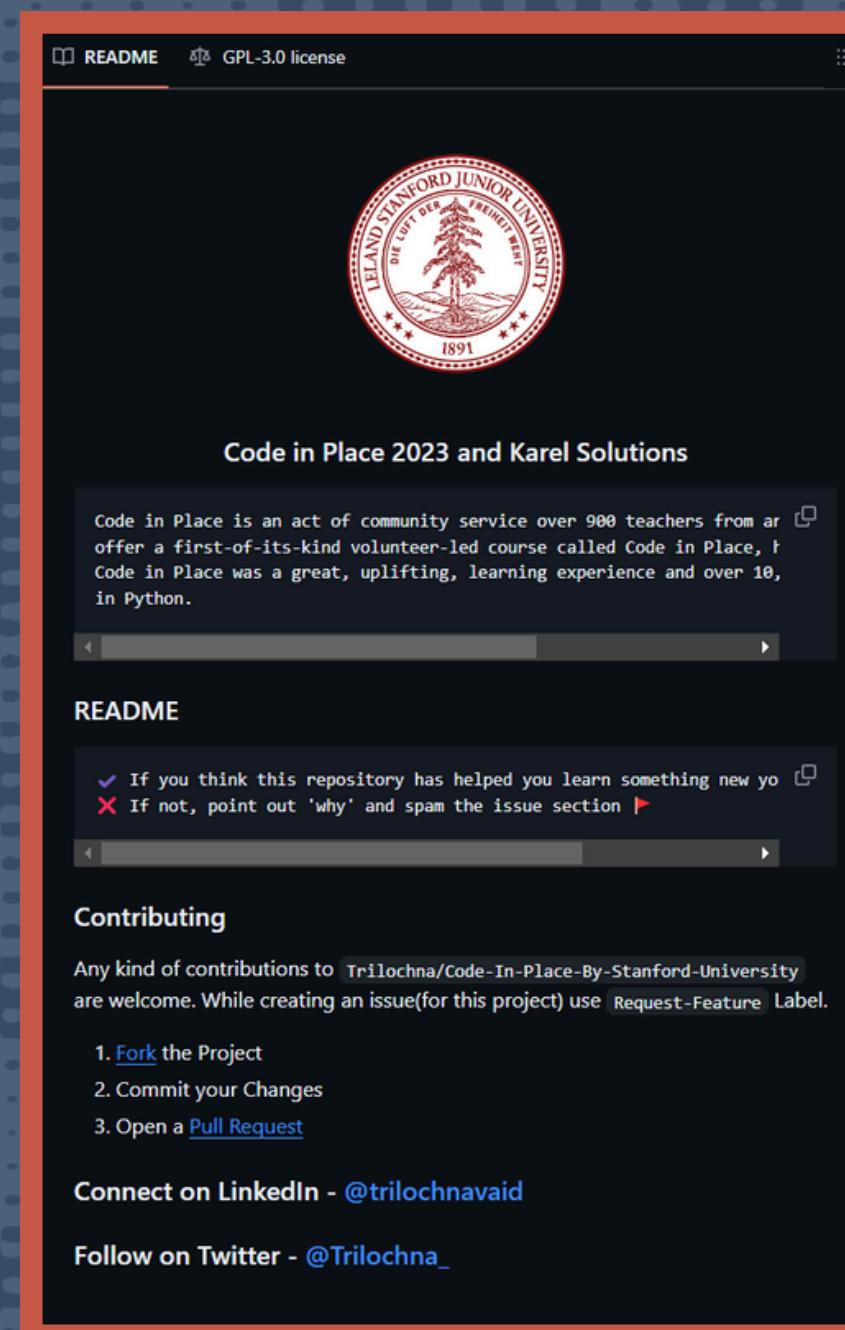
Flexibility

Curriculum modules are customizable and easily integrate existing curriculum and course materials.

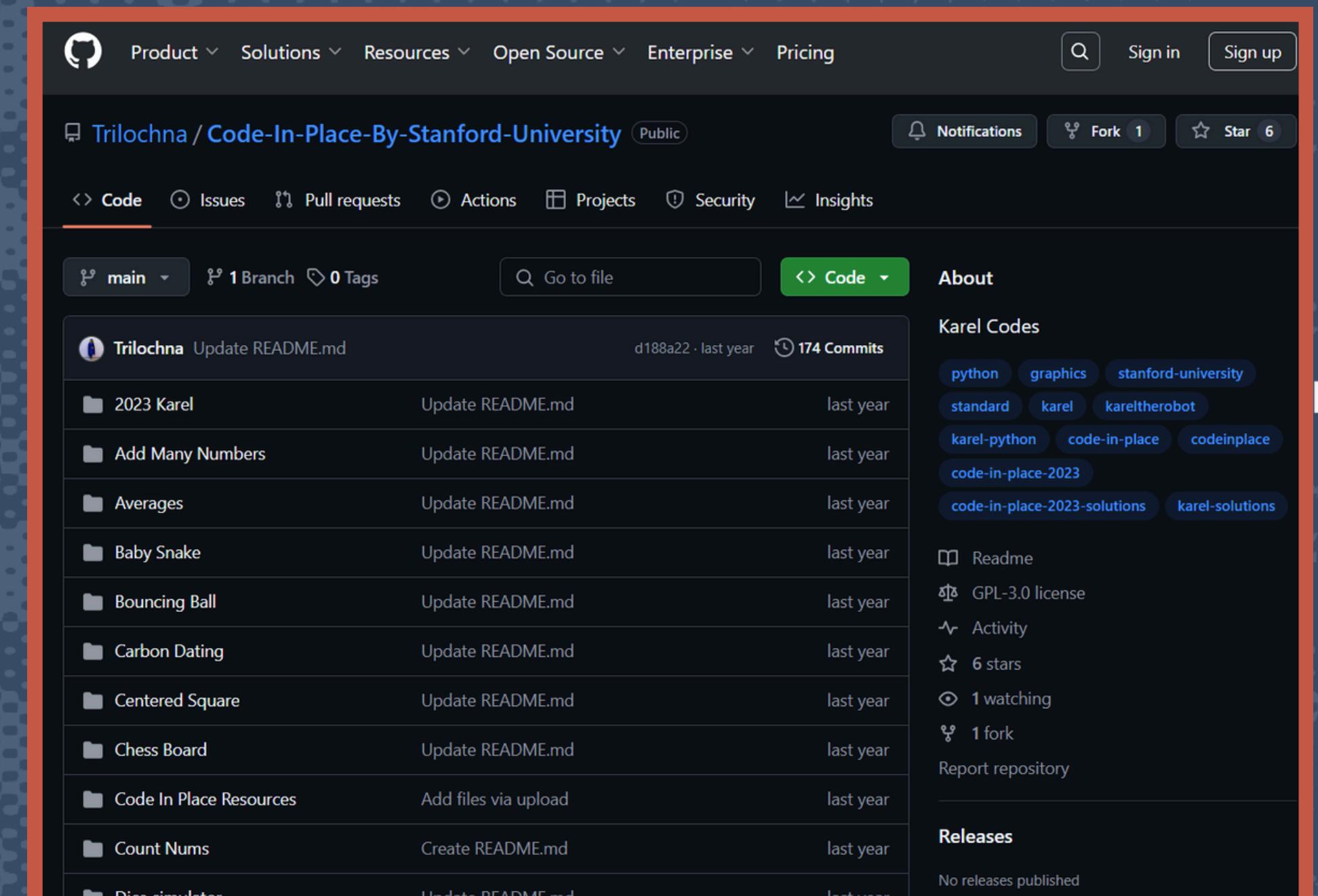
Out of the Box
The standard Code in Place is also ready to use right out of the box, covering the first half of Stanford's CS106A.



OPEN SOURCE



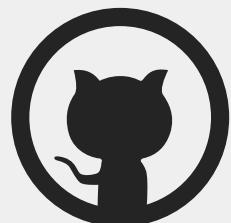
The screenshot shows the GitHub README page for the 'Code in Place 2023 and Karel Solutions' repository. It features the Stanford University seal at the top, followed by a section titled 'Code in Place 2023 and Karel Solutions'. Below this, there's a detailed description of the project, mentioning it's an act of community service over 900 teachers from around the world. The 'README' section includes instructions for contributing, with a list of steps: 1. Fork the Project, 2. Commit your Changes, 3. Open a Pull Request. At the bottom, there are links to connect on LinkedIn (@trilochnavaid) and Twitter (@Trilochna_).



The screenshot shows the GitHub repository page for 'Trilochna / Code-In-Place-By-Stanford-University'. The repository is public and has 174 commits. The 'Code' tab is selected, showing a list of commits from 'Trilochna' updating the README.md file for various projects like '2023 Karel', 'Add Many Numbers', 'Averages', etc. The repository has 1 fork and 6 stars. The 'About' section includes tags such as python, graphics, stanford-university, standard, karel, kareltherobot, karel-python, code-in-place, codeinplace, code-in-place-2023, code-in-place-2023-solutions, and karel-solutions. There are also links for Readme, GPL-3.0 license, Activity, 6 stars, 1 watching, 1 fork, and Report repository. The 'Releases' section indicates no releases have been published.



GitHub Repo
Out of the Box Code in Place is available for free via GitHub. Clone it and play around!



REMOTE + COLLABORATION



Content

CiP provides complete lessons with video lectures paired with course textbook all in one place.

The screenshot shows the CiP platform interface. On the left, there's a sidebar with a navigation menu for 'Karel' topics: Import Packages, main function, helper functions, and start program. Below this is a 'Course Home' section with a 'Stanford & Foothill CS 49' banner, a 'Wk. 01 - Welcome' slide featuring a Stanford University building, and a 'Course Commitment' section with a lightning bolt icon. The main content area displays 'Chapter 4: Decomposition' with a text description and two diagrams labeled 'Before' and 'After' showing a Karel robot navigating a grid to fill potholes.



Exercise Rich

Course materials are self contained within the platform serving as a self-contained learning experience.

This screenshot shows the CiP IDE interface. It features a code editor with Python code for 'main.py' and a 'World' simulation area where a Karel robot is placed on a grid. The code editor includes a 'Docs' section for Karel commands and structures.

A screenshot of the CiP course home page for 'Stanford & Foothill CS 49'. It shows a 'Wk. 01 - Welcome' slide with a Stanford building background, followed by a series of cards for 'Course Commitment', 'Share Your Story', 'Lesson: Welcome to Karel', and 'Reading: Karel ch. 1-3'.

A screenshot of the CiP 'Demo Section' page for 'Stanford & Foothill CS 49'. It features a large image of a Stanford building, a red circular 'D' icon, and sections for 'Demo Section' (with leader 'Demo SL'), 'Section Leader: Demo SL', 'Tuesdays, 3am', and 'Next section: All done!'. There are also buttons for 'Join Section Zoom', 'Section Forum', and 'Email Your Section'.

Delivery Mechanism

Course materials are self contained within the platform serving as a self-contained learning experience.



Online Learning + Human Contact

The flexibility of a MOOC, with a human touch. Weekly sections give students learning support in small collaborative groups.



AGENDA



1

**What is Code
in Place?**

2

**Code in
Place Comes
to Foothill
College**

3

**Section
Leaders**

4

**Section is in
Session!**

F24 CS F049 Foundations Of Computer Progra 01W, 01W Johnson 22336, 22558

FOOTHILL + Stanford

CS49 online!

Start Here

- read the [syllabus](#)
- consume [modules 0 & 1](#)

Code in Place

To visit the URL, click the button below.

[Open in New Tab](#)

COURSE HOMEPAGE

CiP Navigation

CANVAS INTEGRATION

Stanford & Foothill CS49 Fall 2024 Home Page for Students

Student

Home

Admin

Teacher

Student

Section

Code

Forums

Events

SL Training

Lessons

Stories

Grades

Wk. 01 - Welcome

Overview

Course Commitment

Share Your Story

Lesson: Welcome to Karel

Reading: Karel ch. 1-3

Warmup

Karel's Home

Jigsaw Karel

Open Playground (optional)

YOUR PROFILE Sarah Khan

S

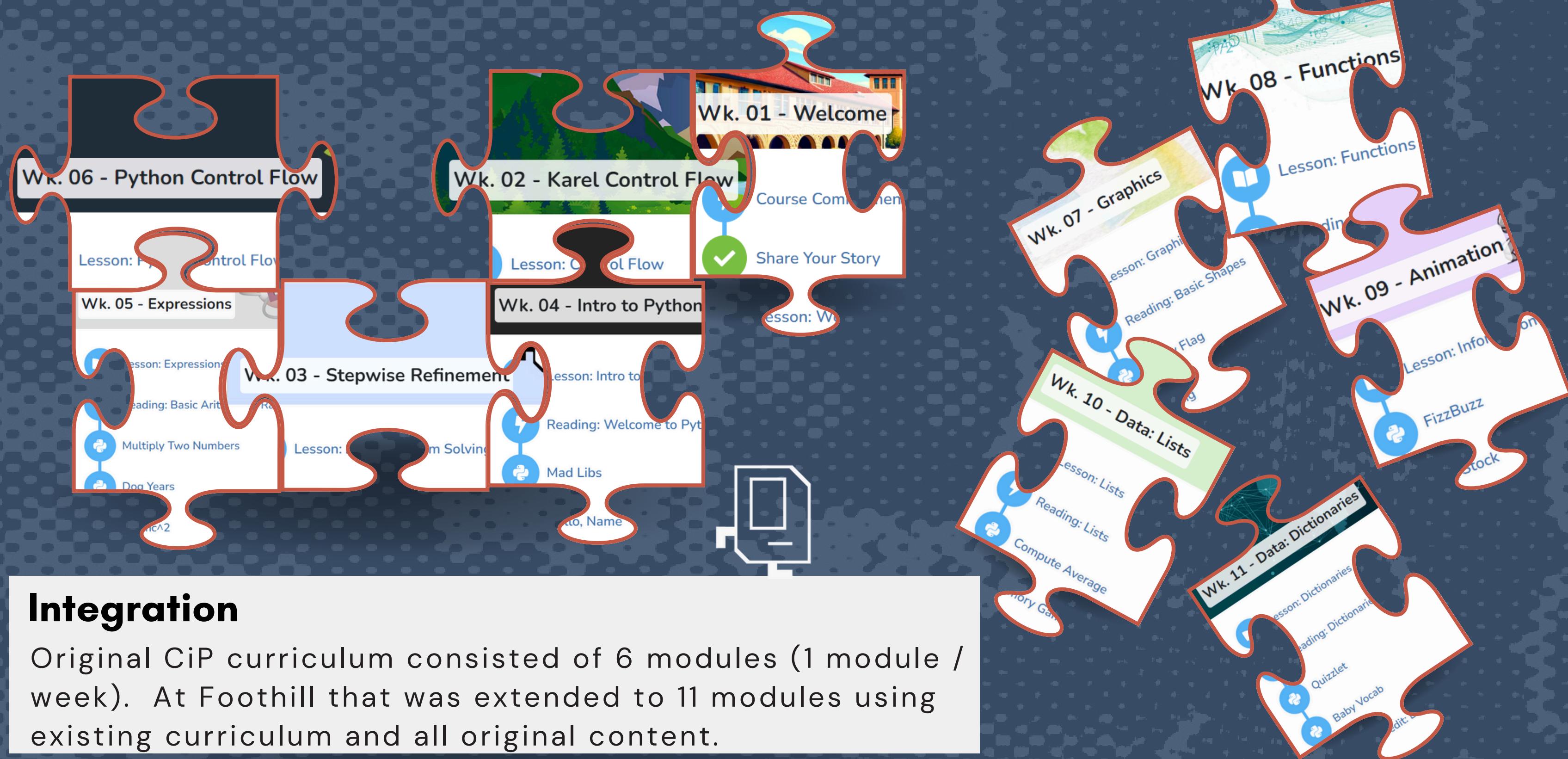
INSTRUCTORS Lane, Chris and Mehran

A top Core Team and amazing section leaders

Are we the class in the history of the world with the most teachers?

CiP Landing Page

CUSTOMIZED CURRICULUM



Integration

Original CiP curriculum consisted of 6 modules (1 module / week). At Foothill that was extended to 11 modules using existing curriculum and all original content.

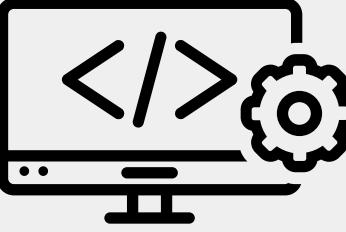
CURRICULUM



FOOTHILL COLLEGE



Innovation

 "It's really fun to see into the development side of things and watch visions and aspirations come to life, and especially with such a shared vision of serving our students in mind."



CiP for Credit

For-credit CiP is still in its early days, but there has been ample support and ongoing collaboration making it an ever growing course.



TRAILBLAZING



Content Integration

CiP's existing content provides a solid, well-rounded foundation that's ready to expand seamlessly to meet any curriculum requirements.

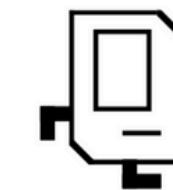


Karel

- 1 - Meet Karel
- 2 - Programming
- 3 - New Functions
- 4 - Decomposition
- 5 - For Loops
- 6 - While Loops
- 7 - Conditionals
- 8 - Refinement
- 9 - Extra Features
- 10 - Reference
- 11 - Code

Chapter 1: Introducing Karel the Robot

In the 1970s, a Stanford graduate student named Rich Parris decided that it would be easier to teach the fundamentals of programming if students could somehow learn the basic ideas in a simple environment free from the complexities that characterize most programming languages. Rich designed an introductory programming environment in which students teach a robot to solve simple problems. That robot was named Karel, after the Czech playwright Karel Čapek, whose 1923 play R.U.R. (Rossum's Universal Robots) gave the word robot to the English language.



Karel

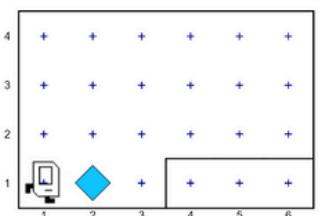
- 1 - Meet Karel
- 2 - Programming
- 3 - New Functions
- 4 - Decomposition
- 5 - For Loops
- 6 - While Loops
- 7 - Conditionals
- 8 - Refinement

Chapter 2: Programming Karel

The simplest style of Karel program uses text to specify a sequence of built-in commands that should be executed when the program is run. Consider the simple Karel program below. The text on the left is the program. The state of Karel's world is shown on the right:

```
# File: FirstKarel.py
# -----
# The FirstKarel program defines a "main"
# function with three commands. These commands cause
# Karel to move forward one block, pick up a beeper
# and then move ahead to the next corner.
from karel.stanfordkarel import *

def main():
    move()
    pick_beeper()
    move()
```



Karel

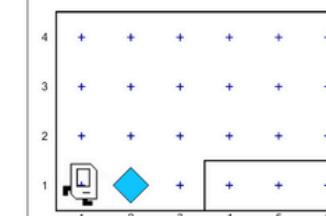
- 1 - Meet Karel
- 2 - Programming
- 3 - New Functions
- 4 - Decomposition
- 5 - For Loops
- 6 - While Loops
- 7 - Conditionals
- 8 - Refinement
- 9 - Extra Features
- 10 - Reference
- 11 - Code

Chapter 3: Defining New Functions

In the last chapter we wrote a program to help Karel climb a simple ledge:

Example: FirstKarel

```
# File: FirstKarel.py
# -----
# Karel picks up a beeper and places it on a ledge.
from karel.stanfordkarel import *
def main() :
    move()
    pick_beeper()
    move()
    turn_left()
    move()
    turn_left()
    turn_left()
    turn_left()
    move()
    move()
    put_beeper()
    move()
```



Even though the FirstKarel program above demonstrates that it is possible to perform the `turn_right()` operation using only Karel's built-in commands, the resulting program is not particularly clear conceptually. In your mental design of the program, Karel turns right when it reaches the top of the ledge. The fact that you have to use three `turn_left()` commands to do so is annoying. It would be much simpler if you could simply say `turn_right()` and have Karel understand this command. The resulting program would not only be shorter and easier to write, but also significantly easier to read.

CONTENT

AGENDA



1

**What is Code
in Place?**

2

**Code in
Place Comes
to Foothill
College**

3

**Section
Leaders**

4

**Section is in
Session!**

WHAT ARE SECTION LEADERS?



Stanford & Foothill CS 49 Your Section

Home Section Forums Code SL Training Lessons Stories

Sarah's Section

Section Leader: Sarah K

Thursdays, 8pm
Next section: All done!

Join Section Zoom
Section Forum

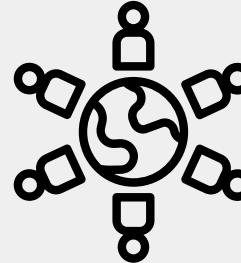
Manish Calli
Sarah Samigano

Ever Expanding

CiP built from the ground up with 5 section leaders at the start, to 3000 section leaders now.

Diversity

Section leaders come from diverse backgrounds with a shared mentality centered on helping guide students into the world of programming.



Stanford & Foothill CS 49 Your Section

Home Teacher Student

Section Forums Code SL Training Lessons Stories

SECTION SCHEDULE

| # | Date | Topic | Docs | Atten. |
|----|---------------------|---------------------|------|--------|
| 1 | April 16th, 3:00 AM | Karel Control Flow | | |
| 2 | April 23rd, 3:00 AM | Stepwise Refinement | | |
| 3 | April 30th, 3:00 AM | Intro to Python | | |
| 4 | May 7th, 3:00 AM | Expressions | | |
| 5 | May 14th, 3:00 AM | Python Control Flow | | |
| 6 | May 21st, 3:00 AM | Graphics | | |
| 7 | May 28th, 3:00 AM | Functions | | |
| 8 | June 4th, 3:00 AM | Animation | | |
| 9 | June 11th, 3:00 AM | Lists | | |
| 10 | June 18th, 3:00 AM | Dictionaries | | |

COLLABORATION

Stanford & Foothill CS 49 Teacher's Lounge

New Post

Your Drafts Announcements

Section Attendance Lane Instructor

Week 1 to-dos! Miranda Instructor April 8 at 4:20 PM

Welcome, SLs! How exciting -- we're getting started at last :)

Your TODO list for Week 1:

- Go to your section page (you can find it on the left sidebar of this page -- green video icon)
- Test your Zoom link and make sure it works -- click the "Section Video Call" button on your section page to try it out. If it doesn't work email Miranda mirandal@stanford.edu ASAP!
- Set a banner image for your section, if you like!



Stanford & Foothill CS 49 Teacher's Lounge

New Post

Your Drafts Announcements

Section Attendance Lane Instructor

Week 1 to-dos! Miranda Instructor

Important SL info Miranda Instructor

Posts

Triple-quoted strings in IDE... Surajit B SL

Close file in headsup Surajit B SL

Week 10 Slides Sarah K SL June 14 at 11:54 AM

Hi everyone!

Here are my slides for this week:
[https://www.canva.com/design/DAGIDeOLkfA/8i8WiMD2utFCjlF51O-pla/edit?](https://www.canva.com/design/DAGIDeOLkfA/8i8WiMD2utFCjlF51O-pla/edit?utm_content=DAGIDeOLkfA&utm_campaign=designshare&utm_medium=link2&utm_source=sharebutton)

Let me know if you have any questions!
Sarah

Training + Support + Learning Model

Section leaders meet weekly to discuss that week's modules with teaching team. CiP platform has a dedicated discussion area for section leaders and teachers as well.

SL CONTENT

Stanford & Foothill CS49 Fall 2024

Your Section

Sarah's Section

Section Leader: Sarah Khan

Thursdays, 8pm

Next section: October 17th (5 days and in 22 hours)

ANNOUNCEMENTS

SL SECTION RESOURCES

SECTION SCHEDULE

| # | Date | Topic |
|---|-----------------------|---------------------|
| 1 | October 3rd, 8:00 PM | Karel Control Flow |
| 2 | October 10th, 8:00 PM | Stepwise Refinement |
| 3 | October 17th, 8:00 PM | Intro to Python |
| 4 | October 24th, 8:00 PM | Expressions |

Handout Code Lesson Plan Solutions Attendance

Stanford & Foothill CS49 Fall 2024

Week 2 section: Karel control flow

This week in section, your first priority is to meet your section leader and discover what sections in Code in Place are all about. Your section leader will therefore spend the first part of section on introducing logistics. Afterwards, you'll solve a Karel problem together using decomposition and stepwise refinement.

1. Hospital Karel

Countries around the world are dispatching hospital-building robots to make sure anyone who needs treatment. They have decided to enlist Karel robots, and your job is to program those robots.

Karel begins at the left (west) end of a row that might look like this:

HANDOUT

IDE | Hospital Karel

Instructions

Your country is prototyping hospital-building robots. They have decided to enlist Karel robots. Your job is to program those robots.

Karel begins at the left end of a row that might look like this:

main.py

```
from karel.stanfordkarel import *
# Here is a place to program your Section problem
def main():
    """
    You should write your code to make Karel do its task in this function. Make sure to delete the 'pass' line before starting to write your own code. You should also delete this comment and replace it with a better, more descriptive one.
    """
    pass
if __name__ == '__main__':
    run()
```

World

CODE

Stanford & Foothill CS49 Fall 2024

Week 2 section: Karel control flow (Solution)

Hospital Karel

```
"""
Program: Hospital Karel
Karel traverses 1st street from west to east, building hospitals
wherever it encounters a beeper.
"""

from karel.stanfordkarel import *

"""

Program: Hospital Karel
Karel traverses 1st street from west to east, building hospitals
wherever it encounters a beeper.

"""

def main():
    while front_is_clear():
        if beepers_present():
            build_hospital()
            safe_move()
```

Sample Section Teaching Videos

- One of the Head TAs giving a demo teach

Helpful Tools

- <https://karelhelper.com/>
- Ice Breaker Ideas
- Slide Deck from a Past SL
- Slide Deck from the Head TAs

Hospital building Karel

Here is the [section handout](#) the students will see! Here are the [section solutions](#) you can use as a reference. Students will see these one hour after their section starts.

LESSON PLAN

SOLUTIONS

FLEXIBILITY

CODE IN PLACE WEEK 2

Welcome to Section!
I'm Sarah, your Section Leader

DECOMPOSITION

Purpose: To break complex sequences into digestible "batches" of steps

- Helper Functions:
- Functions that support other functions
 - Good for reuse



CONTROL FLOW

FOR-LOOP

Syntax:
`for i in range(#):
 (Do this)
 step 1
 step 2`

What are they?
Run through a set of steps a defined number of times.

WHILE-LOOP

Syntax:
`while (active condition()):
 (Do this)
 step 1
 step 2`

What are they?
Continuously cycles through defined step while the condition is active.
Inactive condition means it's invalid and loop stops.
!! CAUTION !!

IF STATEMENTS

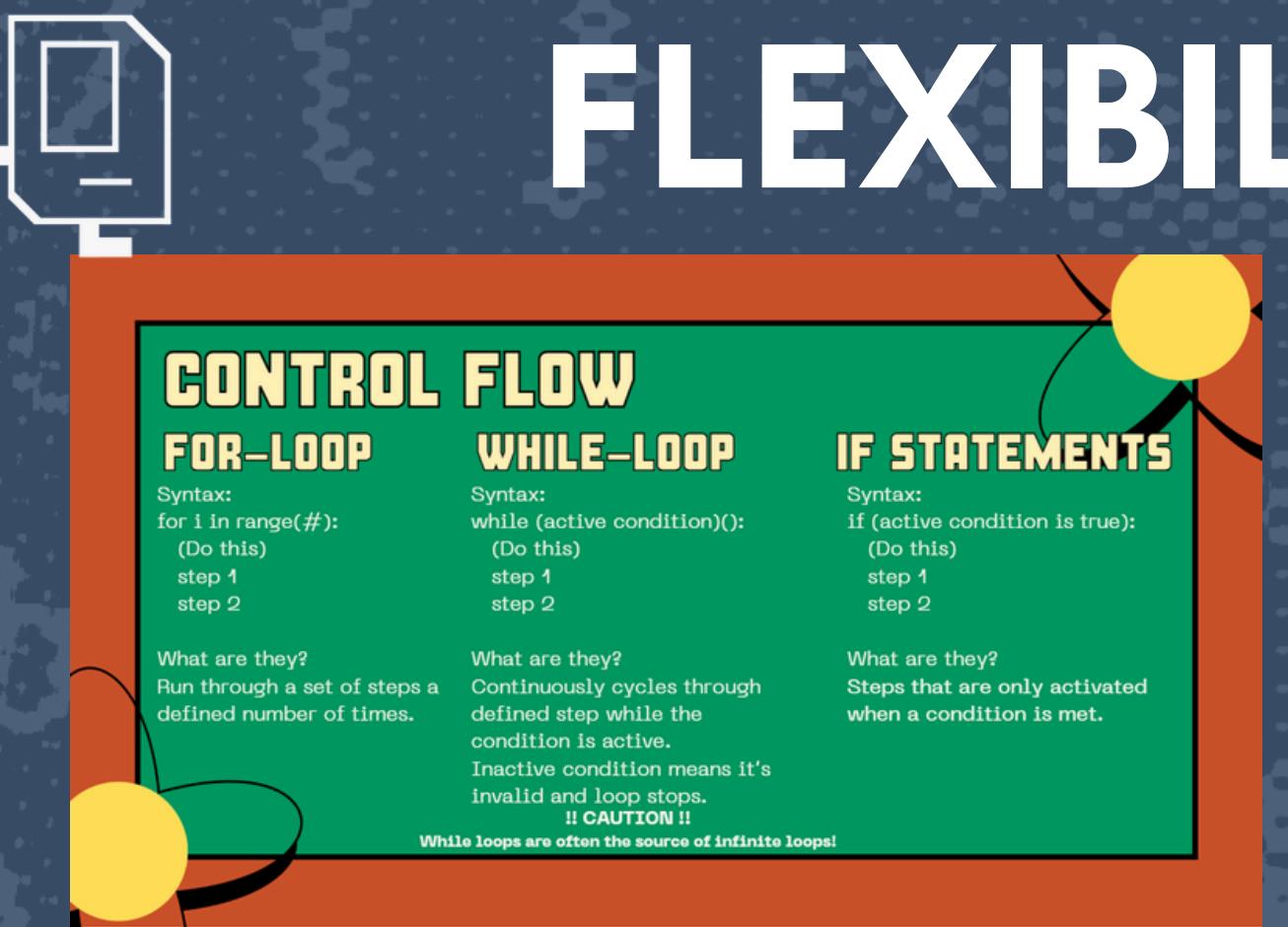
Syntax:
`if (active condition is true):
 (Do this)
 step 1
 step 2`

What are they?
Steps that are only activated when a condition is met.



Fun Environment

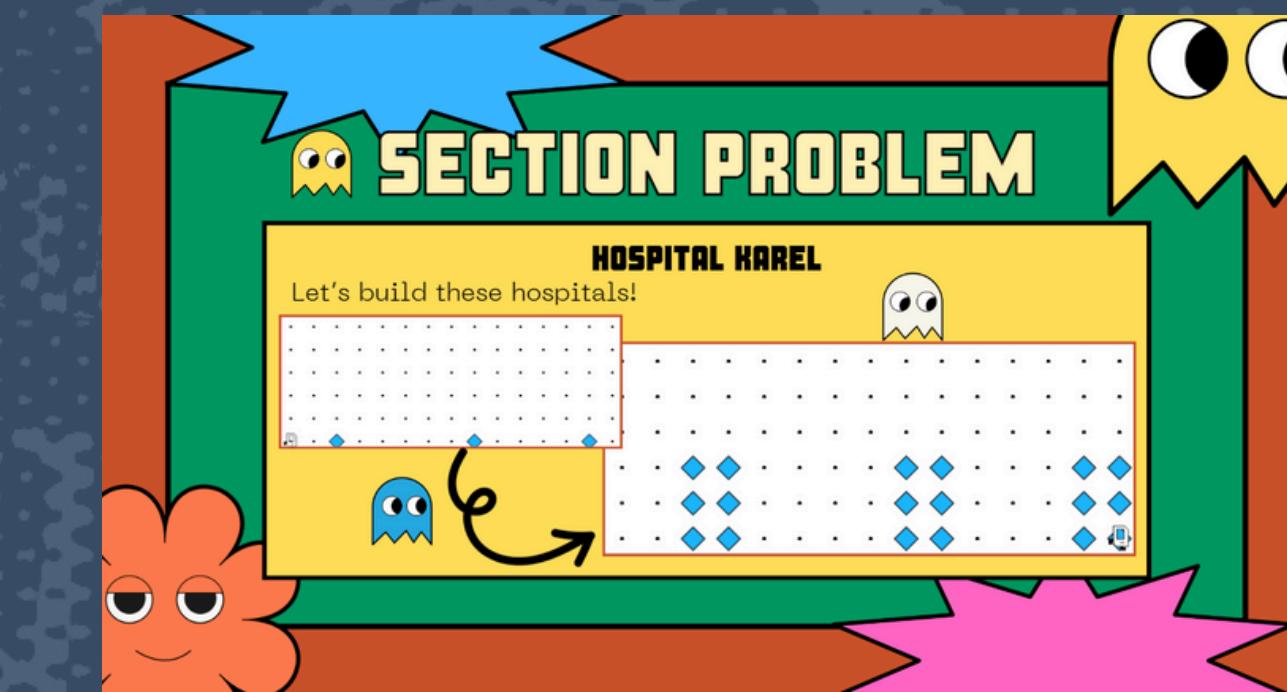
Section content follow CiP lesson plans, but are flexible to section leader teaching styles and provide a fun programming environment and challenges.



SECTION PROBLEM

HOSPITAL KAREL

Let's build these hospitals!



AGENDA

1

**What is Code
in Place?**

2

**Code in
Place Comes
to Foothill
College**

3

**Section
Leaders**

4

**Section is in
Session!**





**THANK
YOU!**

CONTACT US

Interest Form



MATTHEW RASCOFF
VICE PROVOST FOR DIGITAL EDUCATION
STANFORD UNIVERSITY

MIKE ACEDO
ASSISTANT DIRECTOR, PROJECT
INNOVATION & TECHNOLOGY
STANFORD UNIVERSITY
MACEDO@STANFORD.EDU

Session Feedback

We'd love to hear from you! Please let us know your thoughts on this session by filling in the feedback form in Whova. Each session feedback survey you complete in Whova will enter you into a prize drawing for a \$25 Amazon gift card (hard copy submissions are not eligible, but multiple Whova entries are allowed!). Your input is greatly appreciated!

Step 1: go to the Agenda and select the appropriate session

The screenshot shows the Whova app's agenda view for Wednesday, October 16, 2024. The agenda lists several sessions: 'Check-in' at 12:00 PM, 'Lunch Offered (Grab & Go)' at 12:30 PM, 'Data Science Grand Challenge Cohort Me...' at 1:00 PM, and 'Calculus Grand Challenge Cohort Me...' at 1:00 PM. Each session entry includes a timestamp, title, location, speaker information, and a 'Find a lunchmate!' button. At the bottom of the screen, there are navigation tabs for Home, Agenda, Attendees, Community, and Messages.

Step 2: select the Session Feedback button to fill out the brief survey for this session

The screenshot shows the Whova app's session details screen for the 'Data Science Grand Challenge Cohort Meeting' on Wednesday, October 16, from 1:00 PM to 5:00 PM in the Innovation Room. The screen displays the session agenda, speaker bios (Eric Van Dusen, Danay Weldegabriel, Kseniya Usovich), and various interaction buttons. A yellow arrow points to the 'Session feedback' button, which is highlighted in blue. Other buttons include 'Add to My Agenda (4 attending)', 'Q&A', 'Like', 'Polls', and 'Chat'.