Welcome CLPS 0950!

Spring 2022

Agenda

1. Getting to know each other
    a. Instructors
    b. TAs
    c. Students
2. Class Overview
3. Expectations
4. Code of Conduct
5. Problems for bonus
6. Tips for success

Instructors

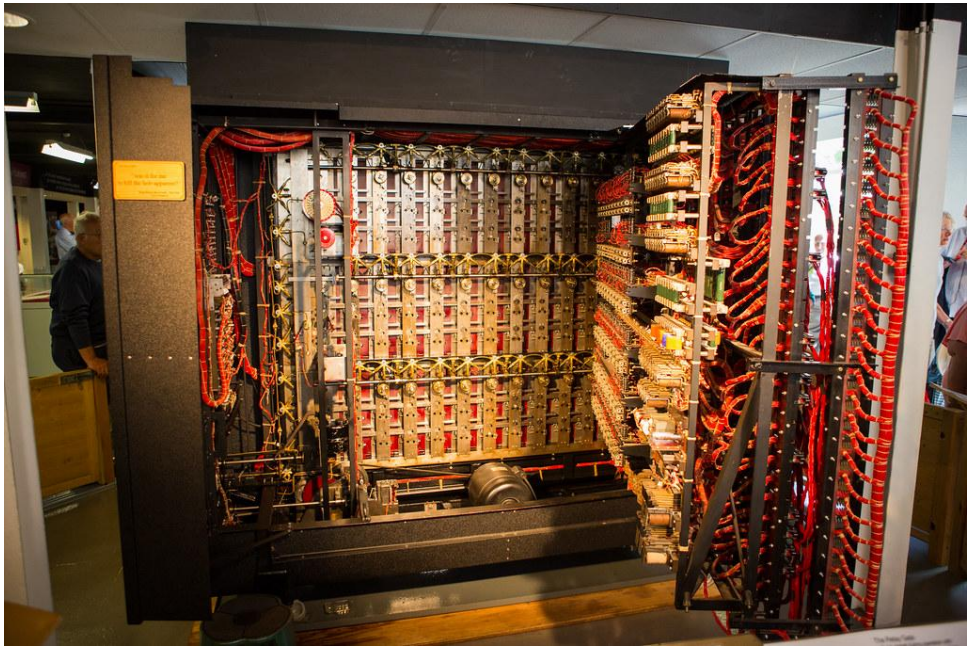
Teaching Assistants

Ice Breaker Activity

Get to know you!

Please make groups of 3-4 students.

- Find a word and a picture that would identify your group.
- Prepare to explain why this picture identify the group.



Class Overview!

General orientation to the course

This course provides an introduction to **programming and computational thinking**. We will use the Matlab programming language as a starting point because it is concise and easy to read. It is also one of the most popular scientific programming languages in psychology and the life sciences more broadly. We will then transition to Python which has now become the main programming language used in data science. Python will be needed for those of you wanting to take additional courses in computational modeling, data science, and machine learning.

Goal

Think computationally and to express solutions to problems so that they can be run on a computer write Matlab and Python programs to solve many common computing challenges associated with the study of the mind, brain, and behavior — from conducting sophisticated data analyses to parsing complex data files to implementing psychophysics experiments understand the deep connection between computational sciences and the brain and cognitive sciences.

Learning activities, assessments, and allocation

1. **completing a sequence of brief (online) tutorials.** At the end of each tutorial, students will be responsible for completing a tutorial assessment involving short programming questions to offer a chance for the student to test their own understanding of the material. These tutorials will also have a Coding in Context section with a mini-assignment/survey to help you understand how coding fits into the research world. **The complete tutorial will be due every Saturday at 11:59 pm.**
2. **Completing weekly programming assignments due every Wednesday at 11:59 pm (one per student).** Collaborations are encouraged but students are expected to produce their own solution and to turn in their own assignment. **If applicable, the name of all the group members who collaborated needs to be acknowledged with each student submission (we are using software to detect plagiarism, and any failure to disclose collaborations will be treated as plagiarism).**
3. **Completing 2 group projects.** These group projects offer an opportunity for students to apply their newly learned programming skills to a relevant problem in the study of the mind, brain, and behavior.

Course Time Allotment

We expect you to take advantage of the in person class time.

In general:

Monday: Questions.

Wednesday: Theory Review.

Friday : Class assignments.

Over the first 6 weeks, you should expect to **complete 6 weekly Matlab programming modules**. Each of these modules will **consist of approximately 2-5 hours of online programming tutorials** and **associated assessments and 8-12 hours of programming assignments (90 hours total).**

In **weeks 6 and 7**, you will be completing a **first group project (25-30 hours total)**.

We will then transition to Python in **weeks 8 and 9 consisting of 2 weekly programming** modules – again consisting weekly of **approximately 2-4 hours of online programming** tutorials and associated assessments and 8-12 hours of **programming assignments (25-30 hours total).**

The last 3 weeks of the course will consist of a “specialization track” where you will be able to choose a programming specialty to explore in more depth. Options will include computational linguistics, computational neuroscience, computational vision, machine learning, data science, and psychophysics. You will complete an online tutorial and programming assignment which will be extended as a group project (30-50 hours total).

Assessment

>90% A, 80–90% B, 70–80% C, <70% NC.

Final grades will be calculated as follows:

1. Weekly tutorial assessments (10): 20%
2. Weekly programming assignments (8 mandatories with additional optional ones; best 8 scores will be taken out of all programming assignments completed): 50%
3. Group projects (2): 15% each
4. Bonuses: Up to an extra 5%
5. 1% penalty for every late day past submission deadlines for tutorials and weekly programming assignments

Lateness

To provide additional flexibility to students, we give you 5 late-day credits which can be used flexibly for tutorials and programming assignments (but not group projects) throughout the semester without incurring any penalty. Beyond these 5 days, the equivalent of 1% of your final grade will be taken off per late day for each and every assignment.

Collaborations and Academic Honesty

Discussion, collaboration, and research on the web are encouraged. However, **the work being turned in for credit must be the student's own**. Sources and collaborators need to be acknowledged.

Plagiarism is considered a very serious offense and we take it seriously. We use software to detect plagiarism. Any violation of the code will be reported to the Dean of the College and may lead to academic sanctions.

Tips For success

It’s ok to make mistakes…

Coding will allow you to do amazing things, but it is not magic! You **will** make mistakes and that is **expected and okay**.

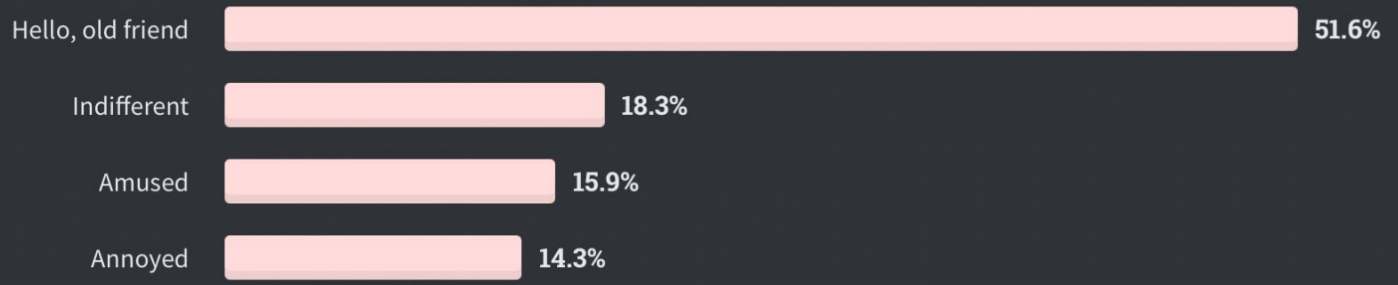
Biggest mistakes people make when they start coding are small but frustrating – things like spelling errors or assuming the computer knows what you want it to do.

Even professional programmers need help!

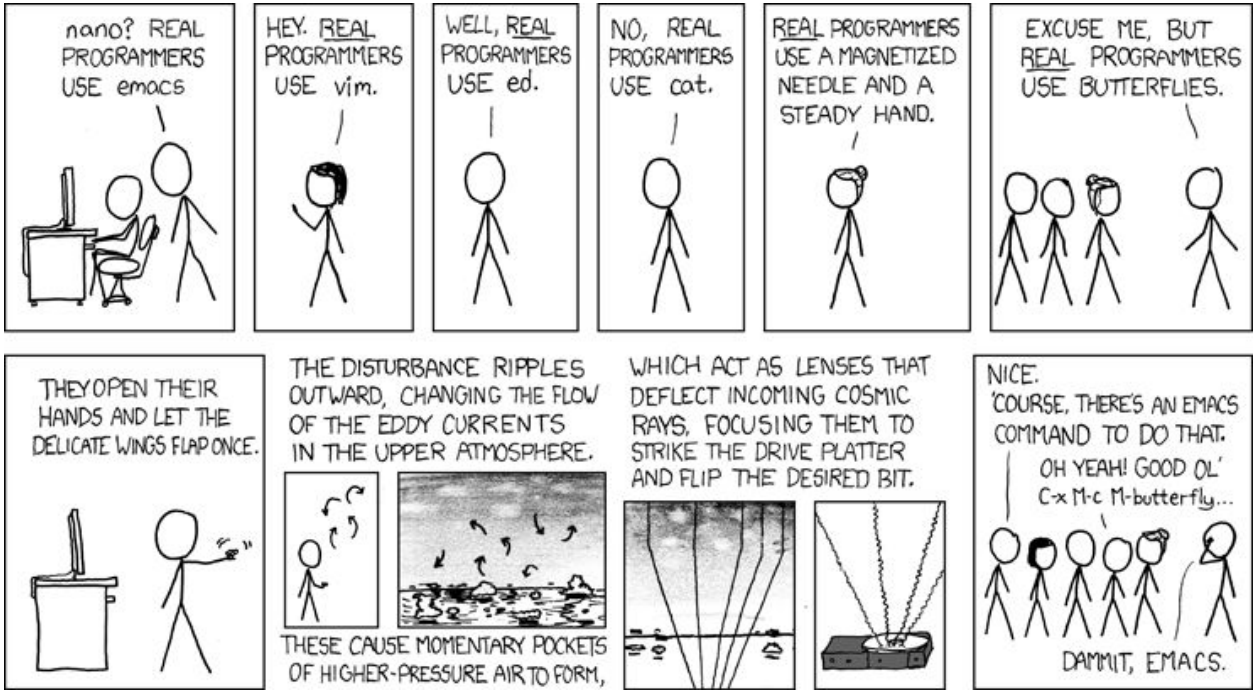
Already Visited Feeling

For the first time, we asked developers how they feel when they search for a coding solution online and the first result link is purple because they already visited the link. About half of respondents chose 'Hello, old friend', which suggests it may be a frequent occurrence for certain tasks. Perhaps this is why over 2.1 million people visited the 'How do I exit the Vim editor?' question on Stack Overflow.

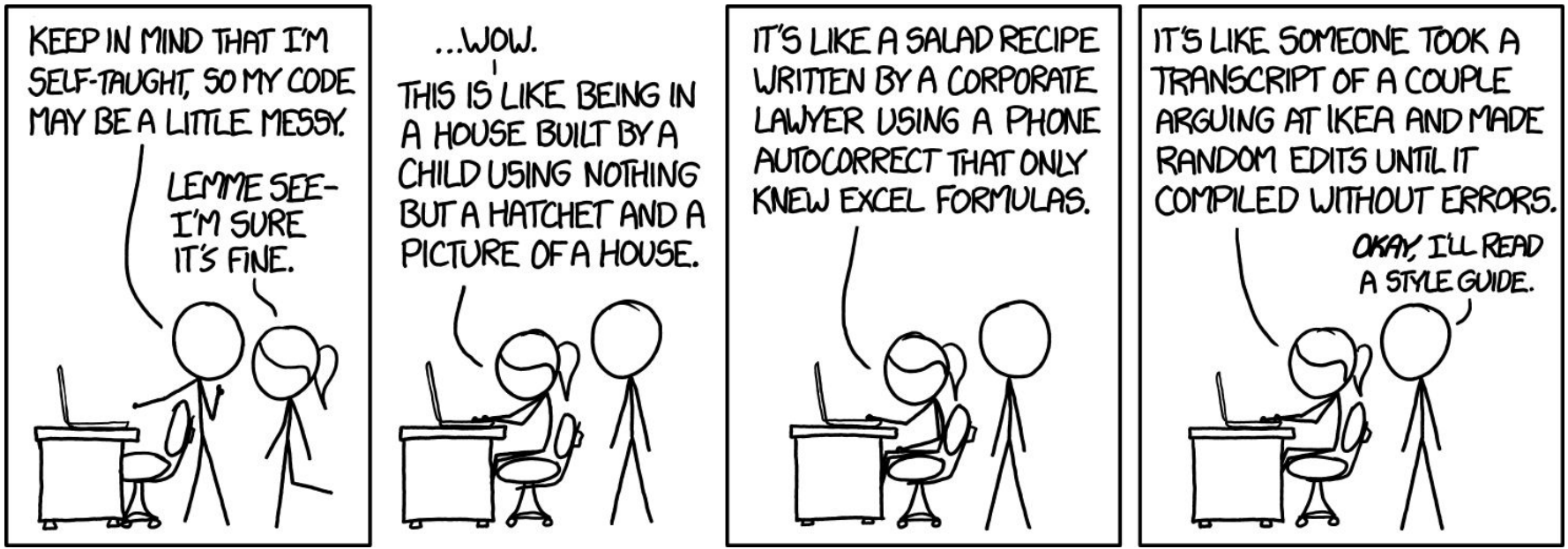
a stackoverflow poll – many people who are professional coders!



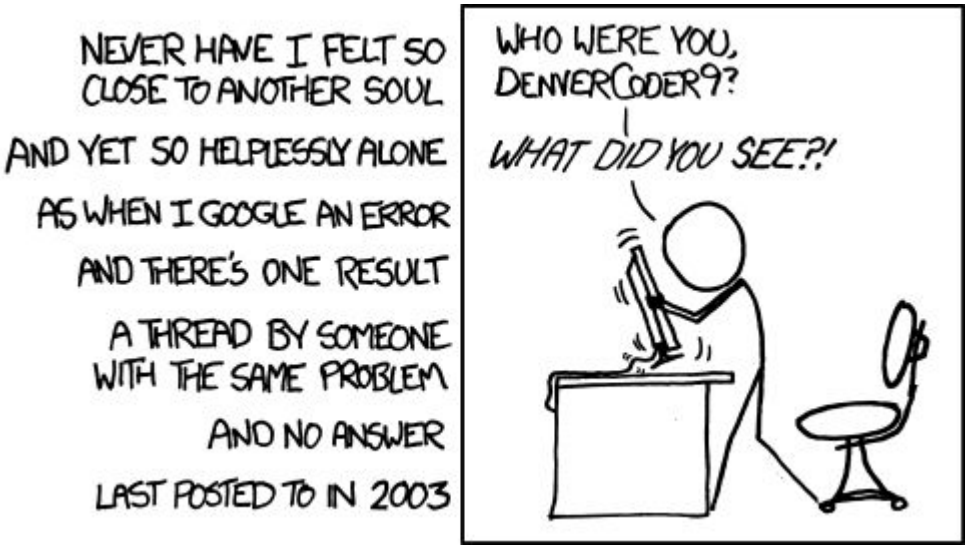
there's no such thing as a "real" programmer – if you are coding, you're a programmer! everyone likes different tools and none is more 'pure' or 'better' than others.



… but there are things that will make your code more useful!



you’re not alone – if you have a problem, it’s likely that someone else has too!



finally… coding is all about practice!

no one is born a “good coder” – but anyone can become one with practice!