CS294 Day 1 Whirlwind Tour of Usable Programming

Preface: We're going to be fitting a lot of activity into a short amount of time, but no need to stress! Today's activity is mostly meant to build shared context for discussions at the end of today's class and at the beginning of next week. So take it seriously—but not *too* seriously!

Stage 1: What's the problem we want to solve?

First, introduce yourselves! :)

Next, what's a programming challenge you and your partner would like to solve? Discuss. This process is probably going to be more fun if you come up with a problem yourselves, but if you've already chatted for 3-4 minutes and you're stumped, scroll to the last page to look at some ideas to get your own ideas flowing.

Make a Google doc that you share with your partner, the Project Doc. (Eventually I'll ask you to send these to me, but they don't have to be fancy!) By the end of this stage, write a 1-2 sentence description of the **target audience** and a 1-2 sentence description of the **programming goal** you'll help this audience reach.

Note: This doesn't have to be The One True Idea. As long as it's interesting enough that you're excited to think about it for the next hour or so, that's enough for our purposes. That said, take the full 8 minutes! No need to home in on just one idea too quickly.

Time: 8 minutes

Stage 2: How might we solve the problem we want to solve?

For the first 5 minutes, brainstorm individually, in silence. Then spend the rest of the time brainstorming together. At the end of this, you want to have 15 ideas for how you might solve your chosen problem. If you only have 10, we'll all understand. :) We're moving pretty quickly here! You should jot down all your ideas. It can be a few sentences, a sketch, whatever helps you remember the key idea.

Important rule: NO JUDGMENT! During this stage, you jot down the good ideas, the bad ideas, the in-between ideas, the horrible ideas, and the world-alteringly brilliant ideas just the same. No assessment. Just more ideas!

It's not quite as good as a whiteboard, but you can sketch collaboratively here: https://sketchtogether.com/ Just remember to take screenshots, since they'll throw away the sketches after an hour or so. Or if you're willing to go through a quick sign up, https://miro.com/ works quite well for collaborative brainstorming.

Time: 12 minutes
After the 12 minutes, take a minute to dump all your brainstorming materials into your shared Project Doc.

Stage 3: Start 'prototyping' one of your ideas!

With your partner, pick one of the ideas from stage 2. By the end of this class session, you'll have run a pilot "Wizard of Oz" study on your idea. A "Wizard of Oz" study is a study in which you test a language, tool, or interface before it's implemented by having a human play the role of the language, tool, or interface.

The goal for stage 3 is to think about the fastest, easiest way to test drive your idea. In stage 4, you and your partner will be testing your tool on another classmate, so your goal now is to figure out what that should look like. What documentation or tutorials will you give your "user" to teach them how to use your proposed tool? What tasks will you assign to your "user?" How will you simulate the experience of using your imagined language or tool?

Aim for your "study" to take about 7-10 minutes.

Some ideas:
- A landscape-orientation Google Doc with two columns can simulate a programming environment with an editor window on one side and the command line on the other. Your "user" can type in the editor window, and one of you can type the "output" into the command line side.
- You can simulate the experience of using a REPL with a single-column doc.
- If you want more of a GUI experience, a Google Slides presentation with some labeled rectangles for buttons might do the trick. Or maybe you can screenshot something that looks similar, then add some buttons of your own on top of the image. Your "user" can narrate what they want to do next, and one of you can move the elements of the interface in real time to show what would happen when the user clicks on X or types Y.

Time: 25 minutes
After the 25 minutes, take a minute to link to any tutorial materials or prototype materials you made in your shared Project Doc.

7 minute break

Stage 4: Wizard of Oz Study

Time to test it out! You're now paired with one other group. You'll test their idea, and they'll test yours. During each "study," there will be four roles.

- **The wizard.** One of the creators of the idea under test. This partner plays the role of the tool, according to your plan from stage 3.
- **The experimenter.** One of the creators of the idea under test. This partner walks the participants through the tutorial or documentation material and assigns the experimental tasks.
- **The participant.** A member of the other team. Will use the "prototype" and follow the directions of the experimenter.
- **The observer.** The observer jots down notes about what's working well and what could be better. This could be about the idea under test, the prototype, the design of the study tasks. In a pilot experiment, we want to learn about all of these! Add these notes to your own Project Doc. (So please also mention which team you're partnered with.)

Decide which idea to test first, assign roles, then go for it!

Time: 10ish minutes per idea, 20 minutes total

If you finish a little early, chat amongst yourselves about what worked well and what could have worked better in each study.

Please have *both* partners take the one-question poll: https://forms.gle/8oXr1bc4apsRNtyk7

Come back together as a class, post-processing discussion

Please have one partner email me your Project Doc! (to: schasins@cs.berkeley.edu subject: 294 Project Doc)

Problems to spark your own ideas!
(vaguer than what I'm asking you to write, but this should give you some ideas)

Audience: Social scientists
Goal: Parallelizing their data analysis scripts

Audience: Software engineers
Goal: Using an under-documented API or library

Audience: Doctors
Goal: In a set of semi-structured patient records, finding patients with a given set of conditions who've received different treatments

Audience: Psychology researchers
Goal: Developing therapy chatbots

Audience: First- and second-year CS students
Goal: Debugging their first assignment in which they use pointers

Audience: Math textbook designers
Goal: Generating diagrams for math problems

Audience: High school history teachers
Goal: Teaching students basic programming skills within a history context/assignment

Audience: Journalists
Goal: Extracting dataset of court cases from PDFs of court docket entries