ECSA-1 TA Guide

Beginning Computer Science

Developed by Paradigm Shift as part of the ECSA-Inspire Series

Have questions? Contact P.ShiftEducation@gmail.com

Urgently need a response during a workshop? Call 714-299-6604

Logistics

- First of all, thanks for being a part of this workshop. It wouldn't be possible without you and I'm sure you recognize the importance of what we're doing!

- Those of us at The Paradigm Shift have designed this workshop to introduce Computer Science as a critical subject that is interesting, fun, and ubiquitous. CS is *everywhere* and becoming one of the most important disciplines for both students' career prospects *and* their ability to think critically.

- This workshop is **50 minutes** long**.**

  - Within the **50 minutes**, students will cover **2 activities** and **2 lectures**

- There should be between **10 and 40 students** participating (grades 7-12).

  - Each student should have his or her **own computer** — headphones and earbuds are not necessary

- There are **no I.T. requirements** beyond relaxing the web filters. These are the websites the students must be able to access:
  - ParadigmShiftCS.org
  - Code.org

- There are contingency plans sprinkled throughout the TA guide in case things aren't going exactly as planned. Don't worry about it, just try to keep things going as smoothly as possible and follow the contingencies if things go wrong.

  - Here's the first contingency plan: if for any reason, a student has already completed this workshop before, have them go to https://codecombat.com/students?_cc=EastSickDay and work through that.

- Lastly, I'd like you to know that even if you have no CS experience and are just helping to facilitate this workshop, you *don't have to worry*. We've come up with a system that shouldn't require much of a technical background for anyone involved in the workshop at all, and I hope you can still be a friendly face that the students can associate with CS. Hopefully you can take something from this too! :)

Schedule

- **Pre-Workshop**

  ○ Login each workstation that the students will be using.

  ○ Ensure websites listed above can be accessed by a student workstation.

- **Workshop**

  ○ Play [Lecture 0](#) on the projector. It's a playlist of two videos so make sure you let both videos play!

  ○ **Activity 1** has two parts, and is an opportunity to give students an introduction on how programming languages work through an interactive demonstration.
    ==The second part of the activity exists if the workshop is running ahead of schedule — if 30 or fewer minutes remain following the completion of Part 1, skip Part 2. Go straight to Lecture 3.==

    ■ Part 1

      ● In this first part of the activity, the kids are going to try to instruct the TA on how to make a peanut butter and jelly sandwich.

      ● The goal of this activity is to show the students that although giving instructions may at first seem easy, there is always a lot of room for misdirection.

      ● To illustrate this, whenever they give you a direction, follow it literally.

        ○ For example, if the first instruction is "put the peanut butter on the bread," take the *jar* of peanut butter and put it on the loaf of bread. If an instruction says to "spread the peanut butter on the bread," use your fingers rather than a knife. If an instruction says to "cut the sandwich in half," be creative and cut it between the two slices of bread. In other words, your goal is to show that instructions need to be very precise, and most importantly, have fun with it!

      ● After this activity, discuss what problems the kids faced while trying to instruct the TA and discuss what solutions could possibly be used to

overcome these challenges when trying to instruct a computer, such as creating programming languages.

- *This activity requires a lot of energy! Do your best to get the kids really into it!* **Intentionally sabotage them as much as possible to illustrate the concepts we're getting at!**

- Contingency plan: if you don't have PB&J for whatever reason, you can do the same activity by making a paper airplane

- **Part 2:** In the second part of the activity, the kids are going to take the "Following Directions Quiz." They are going to have five minutes to complete this quiz. In the instructions for the quiz, it says that the kids should read all the questions carefully before answering any of them. If they are not paying attention, they will end up doing all the actions for each question, such as yelling "Hooray." The correct answer for this quiz is to only have the word "December" written in the top left hand corner of the paper.

  You should explain how this activity is meant to show how computers interpret each instruction literally and execute them in order. A computer cannot infer what you mean or make assumptions, which means you have to be very specific and meticulous when programming. We'll go into a little more depth in explaining this activity, but do your best to engage your students.

  Quiz Worksheet: goo.gl/Aa9l8N

- **Activity 2** gives the kids a chance to work with programming logic by developing their very own version of Flappy Bird. Have them follow the link bit.do/activity-2, which will redirect to the activity. Walk around while they are completing the activity and help students who might be struggling by talking them through problems they're stuck on. ==You should allow an appropriate amount of time for clean up and the final Conclusion/Demo lecture, so feel free to cut them off near the end of Activity 2 (even if some of them haven't finished) in order to get to AND FINISH the final Lecture.==

  - The **Redundancy Plan** for Activity 2 gives the kids a chance to code in Python while playing a game, through Code Combat. The students can go at their own speed, and there are a lot of challenges for them to try. For the most part, you should just be able to let the kids explore, and guide them through if they ever need any help.

Here is a complete teacher guide with the solution to all the challenges: Code Combat Teacher Guide.

For struggling students, try not to immediately give them the answers, but give them hints and try to guide them towards the solution if possible!

Class link: https://codecombat.com/courses?_cc=EastSickDay
Class code: "EastSickDay" (shouldn't be necessary but still included here)

This activity should continue until about 10-12 minutes remain in the workshop. When you stop them and move on, let them know that this is a full featured course and game that they can play out over several hours, and which will give them a good basis for writing actual code.

- Play Lecture 1 on the projector. It's a playlist of two videos so make sure you let both videos play!

- **Post-Workshop**

  - Please go ahead and fill out this TA Response Survey (it'll help us evaluate the effectiveness of the workshop, how we're supporting you, and help us expand and improve the workshop for TA's and students alike): http://goo.gl/forms/TvB3Tvlj4B