# Day One

**Activity:**

* [Program your partner activity](#_7bfog53qne15)
* [Function machine activity](#_3jtio1nrsjyi)
* [“Odds are” game](#_gctlsmqb6s7s)

**Discussion/Explanation Topics:**

* Everyone says “hi” to class pet Rocky
* Intro what is programming? -> leads into [program your partner activity](#_7bfog53qne15)
* Group discussion about programming.
  + Theoretical: “talking to rocks,” communicating with other sentient beings vs. with a non-sentient computer
  + Argument I make:
    - Speech is simplest form of communication – direct, one-to-one communication of ideas between two sentient beings.
    - Writing takes that a step further – now you can communicate to many more people, over much longer periods of time, much more efficiently.
* Why learn programming?
  + Solve problems in real life (show our own examples, Classfinder, etc.).
  + Or...programming is a prerequisite for computer science & really science in general
* Re-iterate “talking to rocks” definition. What is “good” speech/writing? What is a “good” program? A “clean” one? An “elegant” one? A “beautiful” one?
  + Goal of this course is to give you the grounding to make things *work*, but that doesn’t necessarily mean you’ll be able to make things work *well*/elegantly without a lot of experience
  + When writing code, always remember that you’re not *only* talking to the computer, but that both yourself and others will very likely need to read and understand your code later on
* **Break**
* Fundamental CS concept: abstractions
  + We don’t need to worry about connecting wires. Abstractions build up from first principles, with each layer of abstraction only needing to worry about a subset of the entire machine
  + Need a good diagram for this (and non-computer application, ideally)
* What level of abstraction do we work with? Javascript. Our tools include variables, control statements, and functions
* YAEPL vs. Javascript, overview how we’ll switch to Javascript on Day Two
* What are functions -> leads into [function machine activity](#_3jtio1nrsjyi)
* Control flow - if/then, loops, etc.
* Introduce a few other YAEPL functions (primarily rand-num and arrays)
* [Play & program “Odds Are” game](#_gctlsmqb6s7s)

**Projects:**

* [Function machine activity in YAEPL](#_3jtio1nrsjyi)
* [“Odds are” game in YAEPL](#_gctlsmqb6s7s)

**Homework:**

* [Write-a-recipe assignment](#_30b09gd04u88)
* [“I Like…” assignment in YAEPL](#_km125kosbpp8)

**Comments After Class:**

* The “Rocky” idea played well.
* I don’t think we went over arrays as in-depth as we should have.
* Some students came in with significantly more experience than others – I wish we had planned some “extra challenges” ahead of time for them, instead we came up with them on the spot.
* I think using YAEPLE today was, on the whole, a good idea – most of the kids picked it up all right, and it forced some of the kids who already knew another language to generalize their knowledge to a new setting.
* First day got started late, so we rushed near the end.
* They had to finish the “Odds Are” program at home (instead of the “I Like…” assignment originally planned), which was a good challenge for them.
* I do wish we had significantly more time to discuss/demonstrate jumps in class.
* Something we did after the first day of advanced CS that I wish we did for this class – have the students come up with questions for homework. That forced them to think more critically about the material.

# Program Your Partner Activity

* Everyone in the class gets a sheet of paper with a picture/drawing/word on it
* They must write step-by-step instructions using only ones like “Draw line connecting this to that” etc., maybe have a word bank they can use
* Each person then writes out a “program” for their partner to follow
* Partner up people (seat partners? random?)
* First partner gets 3 minutes to read instructions to the partner (NOTE - drawings kept secret!)
* Second partner gets 3 minutes to read their instructions to the partner
* Then each partner should guess what they drew (or find new partners to guess? Or show the entire class and the class guesses?)

# Function Machine Activity

* Get two numbers from the user and calculate their average ((X + Y) / 2)
* Some people in boxes as functions, other people outside passing slips between them
* Function machines:
  + Output
  + Input
  + Add
  + Subtract
  + Divide
  + Multiply
* Some of these are redundant - emphasize that the most challenging part of programming is knowing what to use and what not to use (just because your calculator has 100 buttons doesn't mean you have to use all of them)
* Pass values on little slips of paper

Try to come up with a better test for when we do this on the computer - math operations are kind of sketchy functions in Javascript

# Odds Are Activity

* Someone dares another person to do something - need to come up with a list of ‘appropriate’ dares
* That person picks a number
* Both people pick a number between 0 and that number
* If they pick the same number the person has to do it!
* Possible bets:
  + Tickle Matthew
  + Tell the person next you they’re beautiful
  + Stroke Adhiv’s hair for 30 seconds
  + Dance to a song with Will for 10 seconds

# Write-a-Recipe Assignment

* Find one thing that you do multiple times per week
* Write a recipe for it
* Ex.:
  + Every morning I make my bed
  + Step 1: Wake up
  + Step 2: Put both feet outside the bed
  + Step 3: Stand up
  + Step 4: Pull covers off bed
  + Step 5: Find top side of covers
  + Step 6: Put top side of covers on top of bed
  + Step 7: Flatten covers until bottom side is on bottom of bed
  + Step 8: Place pillows
  + Step 9: Tuck everything in tightly

# “I Like…” Assignment

* Prompt the user to enter something they like
* If they like the same thing you do, let them know!
* Otherwise, ask them one more time what they like