Self-Pacing Instructions

For those of you who are older and/or have some previous coding experience, this guide should keep you busy while us instructors are helping younger/less experienced students.

While self-learning, if you're stuck on something and need help:

1. Fix anything that's underlined in red
2. Review the previous slides to find any clues
3. Ask other students for help
    a. Hint: you *really* learn something best when you have to explain it to someone else, so this helps both of you!
4. Ask Google for help (stackoverflow.com has great answers)
5. Ask us (your instructors) for help
6. After fixing your code, compare it to the code we give you to see if there's any room for improvement

Challenges

1. Complete Day 1 → Day 3 of class content
    a. All starter code is provided in the repl.it classroom
    b. Follow along with the slides
        i. If you see a stop sign, pause! This is where you will complete a challenge, and the next slide will have a solution

2. Hangman
    a. When you're ready, ask us for the project handout
    b. Complete the logic diagram on the handout first
        i. Have us check this - it will help a lot while coding
    c. Write all of the code, then play against eachother to test!
    d. Call one of us over to try it out

3. Falling Sand Game
    a. Uses arrays and looping in an interesting way
    b. You can NOT run this using repl.it online
        i. Download a code editor like IntelliJ Community Edition
    c. More info on the challenge

4. Sorting algorithms; given an array of numbers, how do we sort them from smallest to largest?
    a. Write a bubble sort program to do this
    b. Write an insertion sort program to do this
    c. Explain WHY one might be better than the other

5. Recursion: make a program that prints numbers 1 → 10
    a. First using a FOR loop
    b. Then using recursion
        i. Ask one of us to explain this concept first
        ii. https://www.youtube.com/watch?v=fpuWkZs51aM
        iii. Also check out the Wikipedia page on recursion

6. Need more to do? Check out the Nifty Assignments from Stanford.
    a. All assignments here have been hand-picked as great projects. Look for CS0 or CS1 level projects to start with