<u>Why Computer Science (10 Minutes)</u>

How Does The Industry Work

Specialization - Computer Scientists have a large base of fundamentals and tools to draw on, You pick the tools that help accomplish a specific task and build a program that does something of value

Tools and Libraries - Some programs are built to help other programmers. You build a program that does something well, another programmer then uses your solution instead of writing their own

e.g. Matplotlib - Instead of figuring out how to draw graphics pixel by pixel, Matplotlib figured out how to transform input data into graphs, so we can just use it.

End User Programs - Other programs do something for the typical computer user. These programs almost always use other tools and libraries. e.g. Libraries almost always handle graphics, no end program draws things pixel by pixel

Point: It's very difficult to build a program for an end user that doesn't rely on some other program. Name of the game is specialization and using others' solutions when it would be too complex or costly to develop your own

Software Development is about finding problems and writing programs that solve them

What Industry

Big Tech Companies (Google, Microsoft, Facebook, Amazon) - Primarily focused on computer technology and/or the internet, useful programs ARE their business (Mention the cloud)

Startups (Quora, SnapChat, FiscalNote) - Newer companies typically focused on developing one major product to support a particular mission. Could be at various phases of maturity/funding.

Other Industries (Texas Instruments, Target) - Companies in Biotechnology, industrial operations management, and even Retail depend on computers, and therefore computer programmers to produce their products and ensure the most efficient service.

<u>Review on Loops (5 Minutes)</u>

For loop - Over range, over list, over indices
While loop

Ctrl - C = Kill infinite loop

Intro to Functions (15 Minutes)

Old Way: Get input from user using raw_input
New Way: Get input from program/programmer using function arguments

Functions are a way to bundle code in a way that accepts inputs and produces outputs

Show syntax with 0, 1, 2 arguments

0 - Print countdown
1 - String combine (combine a list of strings into one string)
2 - Euclidean Distance

Functions are just objects - print types
Functions point to code instead of values, only difference

Functions as Black Boxes - introduce Docstring and Specification (Mention how this plays into specialization, a programmer can provide functions for another programmer to use)

Inputs, Return Value, Side Effects
- be clear about return value vs. side effects

Style (10 Minutes)

Why Style Matters
- Help other people understand your code
- Help other people use your code
- Help yourself remember how to use it
- Help you debug

Comments
- Why Comments
- Single Line Comments: Hard to understand example vs. commented version
- Multi Line Comments, Docstring

Good Variable Names
- Why does it matter?
- Computer doesn't care
- Ease of Understanding: Hard to understand example vs. good variable names

Concise is good

Debugging Technique: Print statements to narrow position of error, run code a lot, Google stuff

Debugging Example: Changing list while iterating over it

Debugging Example: Finding location of problem with print statements

Debuggers: At least mention that they're a thing
Show debugger in action?

Common Mistakes (10 Minutes)

Return Value vs. Side Effect - list.append vs. list + list

Modification while iterating

Averaging (integer division)

Indentation (forgetting to indent due to forgotten colon)

Comparison vs. Assignment (== vs. =)

Non-Updating While Loop Condition (Countdown that doesn't count down)

Range Index Error (checking i+1 while iterating over indices)

Print vs. Return (also instance of Return vs. Side Effect; forgetting return statement)