Recap: Assembly to modern software

- Assembly is a low-level language that we can use to give the CPU instructions directly
- Assembly supports common programming constructs like arithmetic, reading and writing from memory, and calling functions
- However, writing assembly is difficult, so we have higher-level languages like C, C++, and Python that we can use instead, which are compiled (or interpreted) into machine code and then executed
- Most (if not all) modern applications use high-level languages, and many modern applications specifically use JavaScript due to wide usage of the language for sites on the web that has bled over to native applications

Overall recap:

- From semiconductors, we get transistors
- From transistors, we get NAND gates
- From NAND gates, we can get all logic gates
- From logic gates, we can construct the components of a CPU
- From the components of a CPU, we can construct a CPU, RAM, etc.
- We can write assembly to use these components to perform programming tasks, including writing high level languages to avoid writing assembly
- With high level languages, we can create modern software for our computers