**🧩 Section Title Ideas:**

* **🧠 CS Foundations: The Core of My Curiosity**
* **📦 Systems, Memory, and Metal – My Journey into Computer Science**
* **🧰 80/20 Foundations: A Deep Dive into the Essentials**

**📝 Introduction Blurb**

Before diving into advanced topics like AI or cybersecurity, I wanted to build a solid foundation — not just theory, but actual system-level fluency.

This 8-week learning path focused on the *80/20 principle*: learning the 20% of core concepts that unlock 80% of practical understanding in computer systems.

The goal?  
To go from curious explorer to confident builder — someone who *understands how machines think*, not just how to command them.

**📚 Topics Covered (use bullet icons or dividers)**

* **Week 1: What Even *Is* an Operating System?**  
  Processes, users, filesystems, and the grand illusion of multitasking
* **Week 2: What’s Inside the Box?**  
  CPUs, RAM, buses, and how electrons become logic
* **Week 3: Memory and Thought**  
  RAM, caches, swap, and how computers “forget” efficiently
* **Week 4: CPU & Execution**  
  Clock cycles, the fetch-decode-execute cycle, and performance bottlenecks
* **Week 5: File Systems & Storage**  
  Inodes, fragmentation, how data *actually* lives on disk
* **Week 6: Networks & Protocols**  
  IP, DNS, ping, packets — and mapping the unseen paths of data
* **Week 7: System Scripting**  
  Automating tasks in Bash and Python, working with the CLI like a power user
* **Week 8: Process Scheduling & Simulation**  
  Simulated Round-Robin and SJF schedulers written in Python  
  (Yes, I gave processes their turn — no bukkake jokes, I promise. Okay, one.)

**💻 Final Project: *Eely OS Helper***

I built a quirky command-line assistant named **Eely OS Helper** —  
a fully interactive Bash tool that:

* Backs up files
* Simulates CPU scheduling
* Displays system stats
* Cleans up your machine
* Shows you love with colored terminal messages and optional eel ASCII art

**🏅 Outcome**

This learning path gave me the confidence to poke at the machine's internals, not just its UI. I now script with ease, debug like a mini-sysadmin, and understand what happens under the hood of any app I write.

It was the beginning of building a real technical identity — and one eel of a good time.

**🔗 Optional Buttons or Callouts**

* View Round-Robin Scheduler on GitHub
* See Full Bash Helper Script
* Next: AI & Machine Learning Path →