

The 48-Book Audit

AUTHOR Mohammed Efaz | IDEA Roadmap | WORDS 2,173 |

READ TIME ~13 min

The 48-book audit behind my roadmap

I did not want the roadmap to become a list of every technical book I had collected. I audited all 48 relevant candidates in my Calibre library and gave each one a job, an optional place, or a reason to stay outside the route.

The result

- Calibre library root: `~/Documents/pkm/02_Areas/Calibre Library`
- Calibre database: `~/Documents/pkm/02_Areas/Calibre Library/metadata.db`
- Books and manuals audited: **48**
- Selected for the default route: **36**
- Optional reads: **1**
- Not selected: **11**

These counts describe fit. They are not ratings of the books.

How I made the choices

I considered the goal, the prerequisite order, the teaching style, the book's unique value, its overlap with the rest of the route, and whether I was likely to finish it and use what it taught.

The [rules that keep this roadmap mine](#) protect the choices that matter most.

What the labels mean

- **Default, Stage N:** I selected it for a specific stage in the roadmap.
- **Default, reference or light:** I selected it, but I will not treat it as a full technical textbook.
- **Optional read:** I may read it, but it cannot block the next stage.

- **Not selected:** It remains in Calibre but is outside the default route.

The complete inventory

Book	Author	My decision	Why	Format
<i>AI Systems Performance Engineering</i>	Chris Fregly	Default, Stage 12	I will use it as the full-stack performance capstone after ML, C++, and CUDA	EPUB
<i>Algorithmic Thinking, 2nd Edition</i>	Daniel Zingaro	Optional after Stage 2	I may use it for extra C problem practice, but its selective structure does not replace my Python DSA route	EPUB
<i>Algorithms, 4th Edition</i>	Robert Sedgewick and Kevin Wayne	Not selected	It is a strong book, but the Java approach overlaps with the Python DSA sequence I prefer	PDF
<i>Asynchronous Programming with C++</i>	Javier Reguera-Salgado and Juan Antonio Rufes	Default, Stage 7	I need practical work with concurrency, futures, promises, coroutines, and	EPUB

Book	Author	My decision	Why	Format
			asynchronous systems	
<i>Bare-Metal Embedded C Programming</i>	Israel Gbati	Default, Stage 6	It gives me practical hardware-near C through registers, startup code, interrupts, and peripherals	EPUB
<i>Build a Large Language Model (From Scratch)</i>	Sebastian Raschka	Default, Stage 10	I will implement the tokenizer, attention, transformer, training, and fine-tuning path myself	EPUB
<i>Build a Reasoning Model (From Scratch)</i>	Sebastian Raschka	Default, Stage 13	It gives the roadmap an applied endpoint in evaluation, inference-time scaling, RL, and distillation	PDF

Book	Author	My decision	Why	Format
<i>C Programming: A Modern Approach, 2nd Edition</i>	K. N. King	Default, Stage 1	This is my main C foundation and the clearest example of the teaching style that works for me	PDF
<i>C++ Memory Management</i>	Patrice Roy	Default, Stage 6	I want a deeper treatment of lifetime, ownership, allocation, RAII, arenas, and allocators	EPUB
<i>C++ Software Design</i>	Klaus Iglberger	Default, Stage 7	I will use its design ideas after I have real C++ projects to apply them to	EPUB
<i>Computer Systems: A Programmer's Perspective, 3rd Edition</i>	Randal E. Bryant and David R. O'Hallaron	Not selected	I dislike its textbook style, so I will not turn it into a mandatory Stage 3 gate	PDF
<i>CUDA Programming Guide, Release 13.3</i>	NVIDIA Corporation	Default, Stage 11	I need the authoritative CUDA programming model and a	PDF

Book	Author	My decision	Why	Format
			reference I can return to while profiling	
<i>Data Structures and Algorithms in Python</i>	Michael T. Goodrich, Roberto Tamassia, and Michael H. Goldwasser	Not selected	It is broad but more textbook-like than I want, and it repeats the role of the Wengrow sequence	PDF
<i>Domain-Specific Small Language Models</i>	Guglielmo Iozzia	Not selected	It is a narrow later specialisation, and the local record appears to be a personalised or prerelease copy	EPUB
<i>Grokking Relational Database Design</i>	Qiang Hao and Michail Tsikerdekis	Not selected	Database design is useful, but it sits outside the goal of this roadmap	EPUB
<i>Hacking and Security: The Comprehensive Guide to Penetration Testing and Cybersecurity</i>	Michael Kofler and collaborators	Not selected	I chose focused Kali and Linux-hardening books instead of another broad security reference	EPUB, PDF

Book	Author	My decision	Why	Format
<i>Hands-On Machine Learning with C++, 2nd Edition</i>	Kirill Kolodiazhnyi	Default, Stage 11, selective	I will use it as a native C++ bridge after Python ML, with attention on runtime and deployment boundaries	EPUB
<i>How Linux Works, 3rd Edition</i>	Brian Ward	Default, Stage 3	This is the readable Linux-internals book at the centre of Stage 3	EPUB, PDF
<i>Introducing C++</i>	Frances Buontempo	Default, Stage 4	It is a current modern C++ introduction and the bridge I need before CUDA and runtime work	EPUB
<i>Just Use Postgres!: All the Database You Need</i>	Denis Magda	Not selected	Database specialisation is outside the present goal	PDF
<i>Learn C++ by Example</i>	Frances Buontempo	Default, Stage 5	I want guided project practice after the C++ introduction	EPUB

Book	Author	My decision	Why	Formats
<i>Learning Kali Linux, 2nd Edition</i>	Ric Messier	Default, Stage 5	It adds testing methodology, analysis, vulnerability work, evidence, and reporting	EPUB
<i>Linux Basics for Hackers, 2nd Edition</i>	OccupyTheWeb	Default, Stage 4	It gives me a beginner-friendly transition from ordinary Linux into security-oriented Linux and Kali	EPUB, PDF
<i>Linux Kernel Programming, 2nd Edition</i>	Kaiwan N. Billimoria	Default, Stage 6	I want practical work with kernel builds, modules, tasks, memory, scheduling, and synchronisation	PDF
<i>Linux Pocket Guide, 4th Edition</i>	Daniel J. Barrett	Default, Stage 3 reference	I will use it for a structured command survey and keep it as a permanent lookup resource	PDF
<i>Linux System Programming, 2nd Edition</i>	Robert Love	Default, Stage 4	It gives me a concise first tutorial before I take on the	EPUB

Book	Author	My decision	Why	Formats
			deeper TLPI treatment	
<i>Linux: The Comprehensive Guide</i>	Michael Kofler	Not selected	It overlaps heavily with Shotts, Ward, Love, TLPI, shell scripting, and the selected security books	EPUB, PDF
<i>Machine Learning with PyTorch and Scikit-Learn</i>	Sebastian Raschka, Yuxi Liu, and Vahid Mirjalili	Default, Stage 9	This is my main practical ML foundation in the ecosystem I will later use for LLM and inference work	EPUB, PDF, TXT
<i>Mastering Linux Security and Hardening, 3rd Edition</i>	Donald A. Tevault	Default, Stage 8	This is my main defensive Linux security and hardening text	EPUB, PDF
<i>Mathematical Foundations of Reinforcement Learning</i>	Shiyu Zhao	Default, Stage 13 support	I need the relevant formal RL foundation before reasoning and post-training work	PDF

Book	Author	My decision	Why	Format
<i>Modern C++ Programming Cookbook, 3rd Edition</i>	Marius Bancila	Not selected	The recipe format overlaps with the structured C++ books already in the route	EPUB
<i>Modern C, 3rd Edition</i>	Jens Gustedt	Default, Stage 3	It updates King's foundation through C23, modern memory, threads, and atomics	EPUB
<i>Reinforcement Learning from Human Feedback</i>	Nathan Lambert	Default, Stage 13	I want post-training and RLHF depth after the LLM and RL foundations	EPUB
<i>Rust Web Programming, 3rd Edition</i>	Maxwell Flitton	Not selected	Rust and web specialisation would split my attention from the systems, security, and AI route I chose	PDF
<i>Statistics Every Programmer Needs</i>	Gary Sutton	Default, Stage 9 support	It gives me practical statistics for sound ML evaluation and experimentation	PDF

Book	Author	My decision	Why	Format
<i>The Linux Command Line, 3rd Edition</i>	William Shotts	Default, Stage 1	This is my main shell, command-line, and Bash foundation	EPUB
<i>The Linux Programming Interface</i>	Michael Kerrisk	Default, Stage 5 deep reference	I will use it as the authoritative userspace Linux and Unix API treatment after Robert Love	EPUB, PDF
<i>The Little Book of Deep Learning</i>	Francois Fleuret	Default, Stage 9 reference	I want a compact conceptual reference beside the practical and full theory books	PDF
<i>The Rust Programming Language, 3rd Edition</i>	Steve Klabnik, Carol Nichols, and Chris Krycho	Not selected	It is a good language book, but Rust is outside my current route	EPUB
<i>The Ultimate Kali Linux Book, 3rd Edition</i>	Glen D. Singh	Default, Stage 8	It gives me the main practical offensive-security lab progression	EPUB
<i>The Ultimate Linux Shell Scripting Guide</i>	Donald A. Tevault	Default, Stage 2	It extends Shotts into practical scripting, text	EPUB

Book	Author	My decision	Why	Format
			processing, interfaces, and debugging	
<i>Tiny C Projects</i>	Dan Gookin	Default, Stage 2	It turns C syntax and exercises into repeated finished programmes	EPUB
<i>Understanding Deep Learning</i>	Simon J. D. Prince	Default, Stage 9	This is my main rigorous deep-learning theory foundation	PDF
<i>Understanding Latency Hiding on GPUs</i>	Vasily Volkov	Default, Stage 11 advanced	I will use it for deeper latency, occupancy, and GPU-performance reasoning after practical profiling	PDF
<i>UNIX: A History and a Memoir</i>	Brian W. Kernighan	Default, Stage 1 light	I want the history connecting Unix, C, tools, shells, and the culture around them	EPUB, PDF
<i>Why Machines Learn: The Elegant Math Behind Modern AI</i>	Anil Ananthaswamy	Default, Stage 9 light	It gives me a readable intuition companion for the mathematics behind ML	EPUB

Book	Author	My decision	Why	Format
<i>A Common-Sense Guide to Data Structures and Algorithms in Python, Volume 1</i>	Jay Wengrow	Default, Stage 1	This is the approachable DSA foundation that fits my preferred language and teaching style	EPUB
<i>A Common-Sense Guide to Data Structures and Algorithms in Python, Volume 2</i>	Jay Wengrow	Default, Stage 2	It continues the Python DSA route and leads into LeetCode 75	EPUB

Edition checks

- My King PDF is the 2nd edition and predates C11. *Modern C, 3rd Edition* supplies the C23 update later.
- My copy of *The Linux Command Line* is the 3rd edition dated 2026.
- My *Linux Pocket Guide* is the 4th edition.
- My *Introducing C++* is the March 2026 O'Reilly edition.
- My *Hands-On Machine Learning with C++* is the 2nd edition with 2025 material.
- My CUDA guide is Release 13.3, dated 27 May 2026.
- My *Build a Reasoning Model* PDF contains 2026 material.

Decisions I want to preserve

- Python remains my main DSA and LeetCode language.
- *Algorithmic Thinking* remains optional.
- CS:APP remains unselected because its teaching style does not fit me.

- Security and reasoning-model work remain in the default route.
- The C++ cookbook and broad *Hacking and Security* book remain outside the route.
- Content-production planning remains outside the curriculum repository.

Keeping the library intact

The Calibre library is read-only during curriculum work. I will open and manage books through Calibre rather than rename, move, or edit its record directories by hand.