**🗓️ Week-by-Week Plan (Strict Chapter Order)**

| **Week** | **Chapter** | **Topic** |
| --- | --- | --- |
| 1 | **Chapter 1: Up and Running** | Compiling, toolchain, IDEs, program structure |
| 2 | **Chapter 2: Types** | Primitive types, type inference, conversions |
| 3 | **Chapter 3: Reference Types** | Pointers, references, memory layout |
| 4 | **Chapter 4: Object Life Cycle** | Constructors, destructors, RAII |
| 5 | **Chapter 5: Run-Time Polymorphism** | Inheritance, virtual functions |
| 6 | **Chapter 6: Compile-Time Poly.** | Templates, overloading, constexpr, generics |
| 7 | **Chapter 7: Expressions** | Arithmetic, logical ops, evaluation order |
| 8 | **Chapter 8: Statements** | Conditionals, loops, control flow |
| 9 | **Chapter 9: Functions** | Parameters, return types, overloading |
| 10 | **Chapter 10: Testing** *(Optional)* | Unit testing, doctest, TDD principles |
| 11 | **Chapter 11: Smart Pointers** | unique\_ptr, shared\_ptr, memory ownership |
| 12 | **Chapter 12: Utilities** | optional, variant, tuple, pair, chrono |
| 13 | **Chapter 13: Containers** | STL: vector, map, unordered\_map, set, etc. |
| 14 | **Chapter 14: Iterators** | Iterator traits, traversal, range-for loops |
| 15 | **Chapter 15: Strings** | std::string, string\_view, common manipulations |

**🧩 Optional Follow-Up / Bonus Chapters**

If you continue past 15 weeks (or assign final project enrichment), these are natural next steps:

| **Chapter** | **Title** | **Value** | **Suggested Use** |
| --- | --- | --- | --- |
| 10 | **Testing** | Can be slotted in Week 10 or used in project setup |  |
| 16 | **Streams** | I/O, stringstreams, file handling | Add to Week 15 or project week |
| 17 | **Filesystem** | std::filesystem, path handling | Optional post-Week 15 module |
| 18 | **Algorithms** | STL algorithms: sort, find, etc. | Final project or advanced unit |