Custom Roadmap: Learn DSA with Rust (Beginner to Advanced)

Phase 1: Master Rust Basics (Week 1-2)

Goals:

- Understand Rust syntax
- Get comfortable with Ownership, Borrowing, and Lifetimes
- Practice control flow, functions, and modules

Topics:

- Variables, Mutability
- Data Types (u32, i64, bool, char, tuple, array, slice)
- Functions & if-else, match
- Ownership, Borrowing, References
- Structs, Enums, Traits
- Vectors, Strings
- Error handling (Result, Option)

Resources:

- Let's Get Rusty YouTube Channel
- freeCodeCamp Rust Full Course (4.5 hrs)

Practice:

- exercism.io Rust Track
- Rustlings Exercises

Phase 2: Learn Core DSA Concepts (Week 3-4)

Goals:

- Understand and implement basic data structures
- Learn time & space complexity

Topics:

- Time & Space Complexity (Big-O)
- Arrays & Vectors
- Linked Lists
- Stacks & Queues
- HashMaps

- Binary Trees (Traversal)

Resources:

- DSA in Rust Playlist - Nathan Stocks
- William Fiset - Data Structures (for theory)

Practice:

- Re-implement each data structure in Rust
- Solve 2 LeetCode problems per topic

Phase 3: Dive Into Graphs & Advanced DSA (Week 5-6)

Goals:

- Work with graphs, trees, recursion
- Implement common algorithms

Topics:

- Recursion & Backtracking
- Trees (BST, AVL)
- Graphs (BFS, DFS)
- Sorting & Searching
- Greedy & DP Concepts

Resources:

- Crust of Rust - Jon Gjengset (Advanced Topics)
- Codeforces Practice with Rust

Practice:

- LeetCode/GeeksforGeeks problems in Rust
- Convert C++/Python DSA code to Rust

Phase 4: Build Projects + Competitive Prep (Week 7+)

Goals:

- Apply DSA in small Rust projects
- Strengthen problem-solving for contests

Topics:

- Todo CLI app (HashMap)
- Simple game (Graph logic)

- File storage system (Persistent data)

Resources:

- Exercism Rust Mentoring
- LeetCode Rust tag