Picking a Language

- Choose a Programming Language
- It does not matter what language you use, just continue with your current language if you have started coding already,
    - Popular choices: Python, C++, Java.
    - Why pick one? (Ease of learning, community support, job market)
    - **Recommendation:** Start with C++ or Python for beginners.

Learning Syntax and Basics

- **Syntax:** Rules of the language
- **Data Types:** int, float, string, bool, etc.
- **Input/Output:** How to take input and display output
- **Conditional Statements:** if, else, elif (or switch-case)
- **Loops:** for, while, do-while
- **Break and Continue:** Control flow in loops

Solving Pattern Problems

- **Why Patterns?** Improves logic and loop understanding
- **Examples:**
    - Right-Angled Triangle
    - Pyramid
    - Diamond
    - Number Patterns (e.g., 1, 12, 123, 1234)

Functions, Arrays, Pointers, and References

- **Functions:** Reusable blocks of code
    - Pass by Value
    - Pass by Reference
- **Arrays:** Storing multiple values
- **Pointers:** Memory addresses
- **References:** Aliases for variables

Solving Easy Array Problems

- Find the largest element in an array
- Reverse an array
- Find the sum of array elements
- Search for an element in an array

Use only variables, loops, and functions to solve these problems.

Object-Oriented Programming (OOP)

- **Classes and Objects:** Blueprint and instances
- **Access Modifiers:** Public, private, protected
- **Encapsulation, Inheritance, Polymorphism ->** Not needed for DSA, but is asked in interviews
- **Example:** Create a simple class (e.g., `Car` with attributes like `color`, `speed`)

Learn Time and Space Complexity

- **What is Time Complexity?**
    - Measures how the runtime of an algorithm grows as the input size increases.
    - Expressed using **Big O Notation** (e.g., $O(n)$, $O(\log n)$, $O(n^2)$).
- **What is Space Complexity?**
    - Measures how much memory an algorithm uses relative to the input size.
    - Also expressed using **Big O Notation**.
- **Why Learn It?**
    - Helps optimize code for performance.
    - Essential for solving problems in competitive programming and interviews.
- **Resources for Beginners**
    - https://www.geeksforgeeks.org/understanding-time-complexity-simple-examples/
    - https://www.interviewbit.com/courses/programming/time-complexity/
    - https://www.bigocheatsheet.com/

Data Structures from Easy to Hard

- **Easy:**
  - Arrays, Strings, Linked Lists
- **Medium:**
  - Stacks, Queues, Hash Maps, Heap, Trees
- **Hard:**
  - Graphs, Heaps, Tries, Advanced Trees (AVL, Red-Black)

Data Structures from Easy to Hard

- Learn the implementation
- Understand all the operations that can done on that Data Structure
- Learn that data structure's corresponding syntax and functions in C++ STL / Java Collections / Python Libraries.
- Learn the time complexity of each function.
- Solve a few easy problems
- Complete all the data structure in the above order

C++ STL: https://www.youtube.com/watch?v=RRVYpIET_RU

Algorithms from Easy to Hard

- **Easy:**
  - Searching, Sorting, 2-Pointer, and Sliding Window
- **Medium:**
  - Binary Search, Merge Sort, Quick Sort, Greedy, Recursion, Bit manipulation
- **Hard:**
  - Dynamic Programming, Backtracking, Graph Algorithms

Follow this sheet: (one of the best resources to learn DSA for beginners): https://takeuforward.org/strivers-a2z-dsa-course/strivers-a2z-dsa-course-sheet-2

📌 DSA Resources for Interview Prep

🔹 **Array & Hashing** – Striver's Playlist

🔹 **Two Pointers** – Striver's Playlist

🔹 **Sliding Window** – Aditya Verma's Playlist

🔹 **Stacks** – Aditya Verma's Playlist

🔹 **Binary Search**

- Aditya Verma's Playlist
- Striver's Playlist

🔹 **Heap / Priority Queue** – Aditya Verma's Playlist

🔹 **Recursion** – Aditya Verma's Playlist

🔹 **Dynamic Programming (DP)**

- Striver's Playlist
- DP on Trees - Aditya Verma

🔹 **Linked List** – Striver's Playlist

🔹 **Trees** – Striver's Playlist

🔹 **Graphs** – Striver's Playlist

🔹 **Tries** – Striver's Playlist

🔹 **Backtracking** – Aditya Verma's Playlist