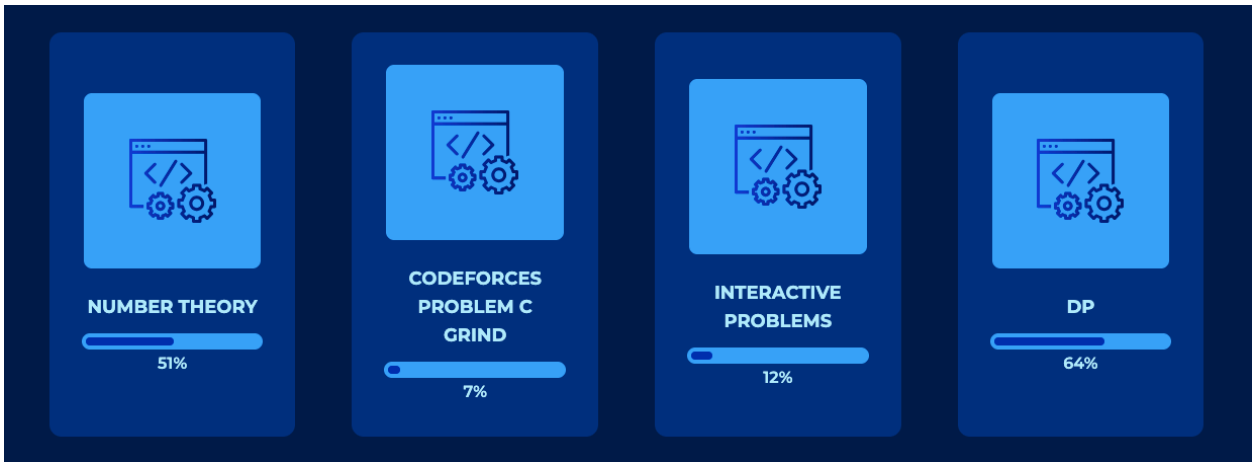
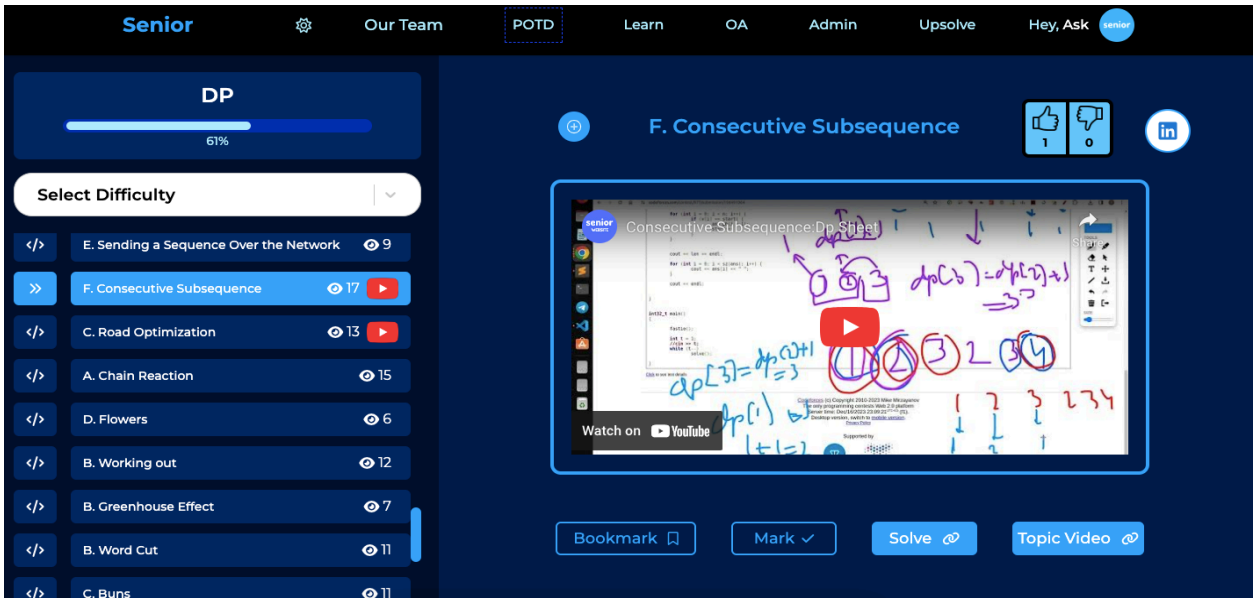
1. Choose any topic, say DP



2. You will see a list of problems. Each problem title has the number of views on the right.

3. Problem Names with a YT Icon show that they have a video editorial



4. The 4 buttons are self explanatory. Mark a problem as solved, or bookmark it to view it later. The color changes will help you identify them from the problem pane later on. Green indicates solved, Yellow indicates bookmarked.

I have taken it as an example for C++ STL but it works for any cards …. SAME RULE FOR ALL CARDS

Use this guide to **SOLVE C++ STL CARDS**. Follow the structured approach to explore, learn, and apply these concepts effectively. If you are not able to understand the answer because of new concepts then read this how to use it and so before solving a problem first understand the concepts, so how to understand the concept and what kind of things you should look for, so here is the answer read this blogs…

Key Questions to Explore:

1. **What is the Concept?**
    - **Definition**: What is the basic definition of the concept?
    - **Purpose**: Why is this concept used in C++ STL? What problem does it solve?
2. **How Does the Concept Work?**
    - **Underlying Mechanism**: What are the underlying mechanisms or data structures that support this concept?
    - **Key Operations**: What are the main operations associated with this concept, and how are they performed?
    - **Time Complexity**: What are the time complexities of these operations?
3. **Why Use the Concept?**
    - **Use Cases**: What are the practical use cases and applications of this concept?
    - **Advantages**: What are the advantages of using this concept over other alternatives?
4. **Features of the Concept:**
    - **Implementation in C++ STL**: How is this concept implemented in C++ STL?
    - **Customizations**: How can you customize the behavior of this concept (e.g., using custom comparators, handling different data types)?
    - **Examples**: What are some examples of using this concept in real-world scenarios?

Learning Steps:

1. **Search and Watch Videos:**
    - **Video Titles to Search**:
        - "Introduction to [Concept Name]"
        - "[Concept Name] in C++ STL"
        - "Understanding [Concept Name] with Examples"
    - **Watch at Least 2-3 Videos**: Ensure you watch multiple videos to get diverse explanations and examples.
2. **Take Notes:**
    - **Concepts**: Write down key points and explanations from the videos.
    - **Code Examples**: Note down any code examples and explanations of how they work.
3. **Attempt Problems:**
    - **AskSenior Sheets**: Try solving a problem related to the concept from the AskSenior problem sheets.
    - **Apply Concepts**: Use what you've learned from the videos to implement your solution.
4. **Watch AskSenior Video:**
    - **Compare Solutions**: After attempting the problem, watch the AskSenior video to see their approach.

    - **Identify Gaps**: Compare their solution with yours to find and understand any differences.
5. **Additional Help:**
    - **Join WhatsApp Group**: If you have doubts or need further clarification, join the provided WhatsApp group for additional support and discussions.
    - Join the WhatsApp Group