Master “C Language” in 30 Days Challenge

(By Tech Involvers)

Project 3: Voting Management System

Instructions:

- Read the problem carefully before trying to solve it.
- Do the tasks on your own. Don’t copy it.
- The output of your program must be the same as given in the sample run.

Overview:

Develop a Voting Management System (VMS) that performs various operations such as registering voters, casting votes, displaying results, and managing candidates.

Requirements:

1. **Voter and Candidate Structures:**
   - Use a `struct` to represent a voter and a candidate. Include fields like `voterID`, `name`, `age`, and `hasVoted` for voters. For candidates, include `candidateID`, `name`, and `votes`.
2. **File Handling:**
   - Store voter and candidate data in files to persist the data even after the program ends.
   - Implement functions to read from and write to the files.
3. **Functions:**

- Register Voter: Register a new voter.
- Add Candidate: Add a new candidate.
- Display Voters: Display all registered voters.
- Display Candidates: Display all candidates.
- Cast Vote: Allow voters to cast their vote.
- Display Results: Display voting results.

4. **Enumerations and Macros:**
   - Use `enum` to define different statuses or roles if needed.
   - Use macros for constants like the maximum number of voters and candidates.
5. **Pointers and Arrays:**
   - Use pointers for dynamic memory allocation if needed.
   - Use arrays for storing voter and candidate records in memory.
6. **Error Handling:**
   - Implement error handling for file operations and invalid inputs.

Features Covered:

Structs and Typedef:

- Voter and Candidate structures.

File Handling:

loadVoters, saveVoters, loadCandidates, and saveCandidates functions.

Functions:

- Registering voters, adding candidates, displaying voters and candidates, casting votes, and displaying results.

Master “C Language” in 30 Days Challenge

Enumerations and Macros:

- Use of MAX_VOTERS and MAX_CANDIDATES macros.

Pointers and Arrays:

- Using arrays to store voter and candidate records.

Input/Output and Operators:

- Using scanf and printf for input and output, and various operators within functions.