

TO-DO LIST

APPLICATION

NAME: AYANNA NEGI

BATCH: 60

SAP ID: 590027891



PROBLEM DEFINITION

The Problem:

- Difficulty in tracking multiple tasks simultaneously
- Risk of missing important deadlines
- Lack of priority-based task organization
- Need for reliable task status monitoring

Why This Matters:

- Students juggle assignments, projects, exams
- Professionals manage meetings, deliverables, deadlines
- Manual methods (paper lists) are prone to loss and disorganization

Project Scope:

- Single-user console application
- File-based persistent storage
- Complete task lifecycle management (Create, Read, Update, Delete)
- Priority and deadline tracking
- Status-based filtering



METHODOLOGY:

- Data Structure Design: Used struct-based approach for tasks with deadlines
- File I/O System: Plain text file storage for persistence
- Menu-driven Interface: Interactive command-line interface

KEY MODULES:

- Task creation and input validation
- Display functions (all/filtered views)
- Update and completion tracking
- File read/write operations



IMPLEMENTATION

Written in C language Uses structures, loops, functions, file handling Key Functions:

- addTask()
- displayTasks()
- displayFiltered()
- updateTask()
- markCompleted()
- deleteTask()
- saveTasksToFile()
- loadTasksFromFile() Create and manage TASKS.txt file

TESTING AND RESULT

- Tested with multiple tasks (20+ entries)
- Verified add, update, delete operations
- Tested mark completed functionality
- Validated file save and load operations
- Confirmed proper filtering (pending/completed)
- Tested ID assignment and management
- Gives output in "tasks.txt"

SAMPLE OUTPUT SCREENSHOT

- Main menu
- Task addition process
- Task list display
- Filtered views
- Update/delete confirmations



```
--- TO-DO MENU ---  
1. Add Task  
2. Show All  
3. Show Pending  
4. Show Completed  
5. Update  
6. Mark Completed  
7. Delete  
8. Exit  
Choice: 2  
  
ID: 1  
Title: eat  
Priority: Medium  
Deadline: 11-12-2025 22:30  
Status: Completed  
  
ID: 2  
Title: sleep  
Priority: Medium  
Deadline: 04-12-2025 07:30  
Status: Pending
```

NOVELTY / CONTRIBUTIONS

Unique Features:

1. Structured Deadline System: Separate fields for date and time rather than simple string storage
2. Auto-incrementing IDs: Automatic task ID generation based on last task
3. Pipe-delimited Format: Human-readable yet parser-friendly file format
4. Filtered Views: Separate displays for pending vs completed tasks
5. In-place Updates: Non-destructive task editing with data preservation

Design Choices:

- Why C? Lightweight, fast execution, minimal dependencies
- Why file-based? No database installation needed, portable across systems
- Why pipe delimiter? Handles titles with spaces, easy to parse

Improvements Over Basic Approach:

- Persistent storage (survives program restarts)
- Priority levels for task organization
- Status tracking (pending/completed)



FEATURES SUMMARY

- Add tasks with title, priority, and deadline
- View all tasks or filter by status
- Update existing task details
- Mark tasks as completed
- Delete tasks
- Automatic data persistence
- Auto-generated unique IDs



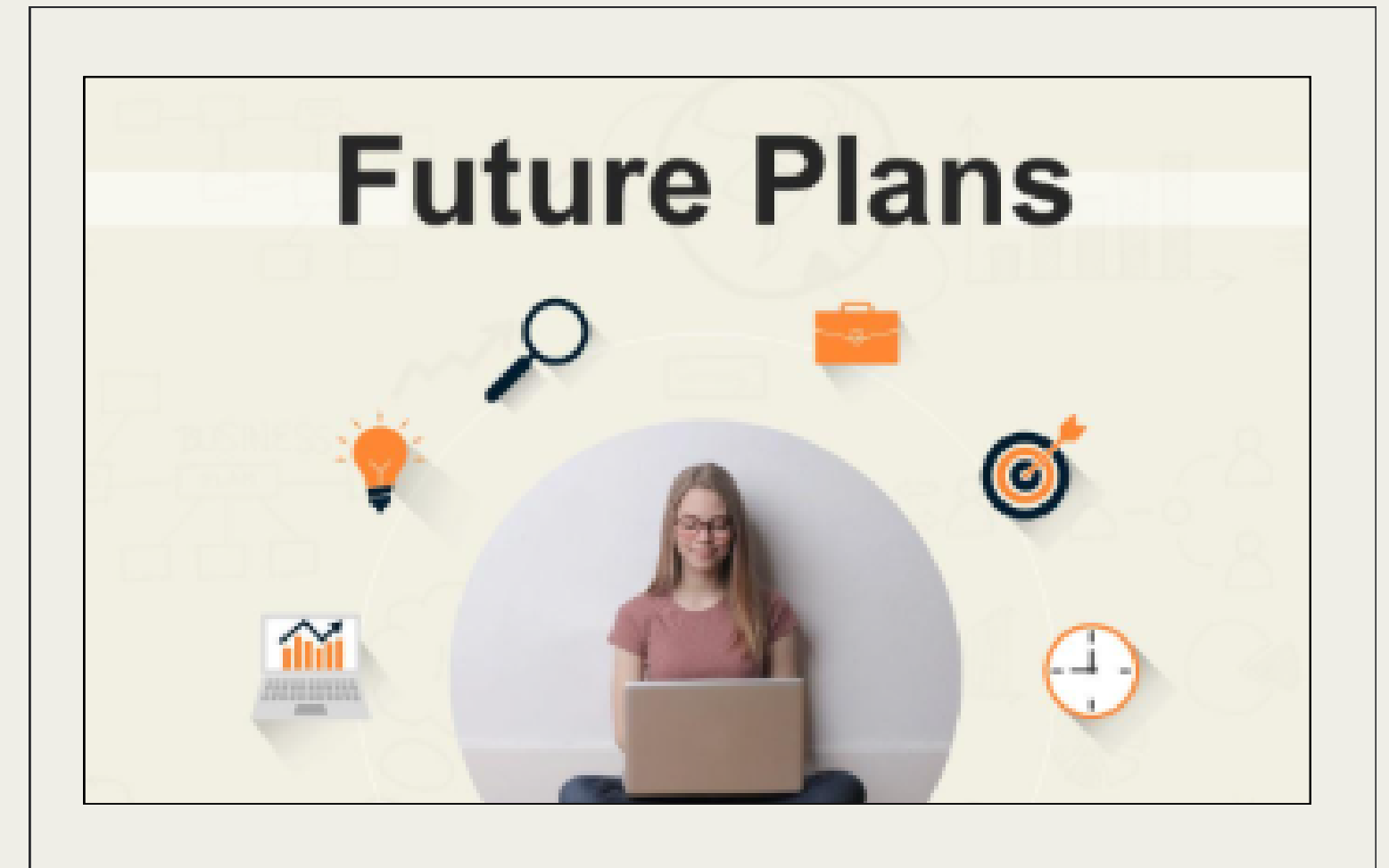
LIMITATIONS & FUTURE ENHANCEMENTS

Current Limitations:

- No input validation for dates
- Fixed maximum of 100 tasks
- No search functionality
- No sorting options

Future Enhancements:

- Add search by title/priority
- Sort by deadline or priority
- Input validation for dates
- Color-coded priority display
- Recurring tasks support
- Category/tag system



REFERENCES

Kernighan & Ritchie - The C Programming Language (2nd Edition)

Yashavant Kanetkar - Let Us C (17th Edition)

GeeksforGeeks - C Programming Tutorials

<https://www.geeksforgeeks.org/c-programming-language/>

Programiz - C Structures and File Handling

<https://www.programiz.com/c-programming>

TutorialsPoint - C Arrays and Pointers

<https://www.tutorialspoint.com/cprogramming/>

Lecture Notes

Class demonstrations and guidance

VS Code - Code editor

GCC Compiler - Code compilation



Thank you!

