C Programming Mini Projects Collection

This repository contains a set of mini-projects built using the C programming language. These projects demonstrate fundamental programming concepts such as conditionals, loops, arrays, functions, structures, and file handling.

Each project is designed to be self-contained and compiled independently.

Projects Overview

1. Text-Based Adventure Game

A simple console-based adventure game where the player navigates through rooms using directional commands.

Key Features:

- Room-to-room navigation (North, South, East, West)
- Interactive text prompts
- Endless play loop

Run Instructions:

```
gcc adventure_game.c -o adventure_game
./adventure_game
```

2. Hotel Menu System

Simulates a restaurant menu that displays food items based on the meal type selected by the user.

Key Features:

- Menu for Breakfast, Lunch, Dinner, and Snacks
- Uses switch-case structure
- Simple CLI interaction

Run Instructions:

gcc hotel_menu.c -o hotel_menu

./hotel_menu

3. CR Election Voting System

A basic voting system to simulate a class representative election between three candidates.

Key Features:

- Vote casting system

- Real-time vote count

- Winner declaration logic

Run Instructions:

gcc voting_system.c -o voting_system

./voting_system

4. Student Management System

Allows user to manage student records including names, roll numbers, and marks.

Key Features:

- Add and display students

- Search by roll number

- Delete student records

- Structured using arrays and basic CRUD logic

Run Instructions:

gcc student_management.c -o student_management

./student_management

5. Bank Management System

Provides functionality for managing bank accounts and performing basic operations.

Key Features:

- Create accounts

- Deposit and withdraw money

- View and search accounts

- Menu-driven with input validation

Run Instructions:

gcc bank_system.c -o bank_system

./bank_system

Getting Started

To run any of the programs:

1. Clone or download this repository.

2. Navigate to the directory containing the .c file you want to run.

3. Compile the code with a C compiler like gcc.

4. Execute the compiled file.

Example:

gcc student_management.c -o student_management

./student_management

Requirements

- GCC compiler (or any C compiler)

- Basic terminal or command line usage

License

Author

Developed by a student practicing C programming fundamentals. Contributions and improvements are welcome!