1. Student Grading System

- **Description**: Create a program that calculates grades for students based on their marks in various subjects. The system should take input for marks, calculate the average, and assign a grade (A, B, C, etc.) based on predefined ranges.
- **Key Concepts**: Conditional statements, loops, input/output handling.

2. Library Management System

- **Description**: Build a simple library system that allows adding, removing, and viewing books. Users should also be able to search for books by title or author.
- **Key Concepts**: Arrays or lists, loops, functions, file handling (for saving and retrieving data).

3. Bank Account Management

- **Description**: Design a program that allows users to create an account, deposit, withdraw, and check the balance. The project should also track transactions.
- **Key Concepts**: Conditionals, functions, loops, and potentially classes (if object-oriented programming is introduced).

4. Payroll System

- **Description**: Develop a basic payroll system that calculates salaries based on hours worked and hourly rates, applying tax deductions if required.
- **Key Concepts**: Functions, loops, conditional statements, and simple calculations.

5. Encryption and Decryption Program

- **Description**: Build a program that can encrypt and decrypt simple messages using a basic cipher (like Caesar Cipher).
- **Key Concepts**: Strings, loops, functions.

6. Tic-Tac-Toe

- **Description**: Develop a two-player Tic-Tac-Toe game. Players take turns marking X and O on a 3x3 grid until someone wins or the game is a draw.
- **Key Concepts**: 2D arrays, loops, conditionals.

7. Snake Game (Text-Based)

- **Description**: Build a simplified snake game where the player collects food to grow the snake's length. The game ends if the snake runs into the wall or itself.

- **Key Concepts**: Arrays or lists, loops, conditional statements.

8. Race Game

- **Description**: Create a simple race between two players, where each turn advances the player a random distance forward. The first player to reach the finish line wins.
- **Key Concepts**: Randomization, loops, conditionals.

9. Maze Runner (Text-Based)

- **Description**: Design a simple text-based maze where the player navigates using commands (e.g., N for north, S for south) to reach the exit.
- **Key Concepts**: 2D arrays, loops, conditionals.

10. Catch the Thief

- **Description**: Set up a grid-based game where a thief moves around randomly, and the player tries to catch them within a set number of moves.
- **Key Concepts**: 2D arrays, randomization, loops.

11. Typing Speed Test

- **Description**: Create a game that displays random words, and the player has to type them as quickly as possible. Measure and display typing speed at the end.
- **Key Concepts**: Strings, loops, timing functions (optional).

12. Catch the Falling Objects

- **Description**: Simulate objects "falling" down a screen, and the player has to "catch" them by typing in the correct command or input.
- **Key Concepts**: Loops, conditionals, arrays.

13. Hotel Reservation System

- **Description**: Design a hotel booking system that manages rooms, bookings, cancellations, and customer details.
- **Key Concepts**: Classes, dynamic memory allocation, file handling, multithreading (for handling multiple bookings simultaneously).

14. Library Management System with Search and Sorting

- **Description**: Build a library management system that allows adding, deleting, searching, and sorting books using algorithms like binary search and merge sort.
- **Key Concepts**: Sorting algorithms, binary search, classes, file handling.

15. Flight Reservation System with Seat Allocation

- **Description**: Build a flight reservation system with seat allocation. Include features like seat selection and waiting lists.
- **Key Concepts**: Data structures (trees, queues), file handling, classes, dynamic memory allocation.

16. Hospital Management System with Patient Records

- **Description**: Develop a hospital management system for managing patient data, doctor schedules, and treatment records.
- **Key Concepts**: Classes, file handling, STL containers, inheritance.