

# C Programming (IT, 2022) – Lab Projects

## Project List with Description

No.	Title and Description
1	ATM Interface
	<p>Design a simple text-based interface that allows a user to access ATM functions. There should be pre-stored data of 10 customers. The functions to be covered by the program are:</p> <ul style="list-style-type: none"><li>• Login using a valid A/c number and password (choose validity of your own choice)</li><li>• Check balance</li><li>• Withdraw cash</li><li>• Deposit cash</li><li>• Exit and print bill of all transactions of one session.</li></ul>
2	Periodic Table Display
	<p>Design an interface that allows a user to access one or more elements of the Modern Periodic Table. The functions to be covered are:</p> <ul style="list-style-type: none"><li>• Add an element</li><li>• Search an element</li><li>• Update an element</li><li>• Access specially named rows and columns of elements, such as noble gases or alkali metals</li><li>• Exit</li></ul>
3	Quiz game
	<p>Design a mini quiz that allows a user to play through 10 questions. The functions to be covered by the program are:</p> <ul style="list-style-type: none"><li>• Provide entry to a valid user</li><li>• Menu to start, view high score, reset score, help, quit</li><li>• Play ten questions</li><li>• View score</li><li>• View high score</li><li>• Reset scores</li><li>• Exit</li></ul>
4	Library Management
	<p>Design a simple text-based interface that allows a user to access a virtual library as the librarian. There should be pre-stored data of 20 books. The functions to be covered by the program are:</p> <ul style="list-style-type: none"><li>• Login with valid ID and password</li><li>• Update password</li><li>• Add books</li><li>• Search and edit books</li><li>• View books</li><li>• Issue books</li><li>• Return books</li><li>• Delete books</li><li>• Exit</li></ul>
5	Bank Account Management
	<p>Design a simple bank management interface that allows a user to login as appropriate designation and perform bank tasks. The functions covered are:</p>

	<ul style="list-style-type: none"> <li>• Customer: <ul style="list-style-type: none"> <li>○ Create New account</li> <li>○ Login</li> <li>○ Check Balance</li> <li>○ Deposit</li> <li>○ Withdraw</li> <li>○ Modify account</li> <li>○ Close account</li> <li>○ Exit</li> </ul> </li> <li>• Manager: <ul style="list-style-type: none"> <li>○ Login</li> <li>○ Check accounts</li> <li>○ Generate statement</li> <li>○ Exit</li> </ul> </li> </ul>
6	Traffic Signal Simulation
	Design a text-based simulation of a traffic signal area, showing which lane is red/yellow/green basing on the Indian traffic system (green changes to yellow and finally red, instead of reverse). Put random instances of ambulances approaching and show that the system can handle it.
7	Full-feature calculator
	Design a full-fledged calculator that performs the following arithmetic functions: <ul style="list-style-type: none"> <li>• Add</li> <li>• Subtract</li> <li>• Multiply</li> <li>• Divide (with precaution)</li> <li>• Find power</li> <li>• Find binary value</li> <li>• Find square root</li> <li>• Find reciprocal</li> <li>• Find logarithm</li> <li>• Find factorial</li> <li>• Find modulus</li> <li>• Find absolute value</li> <li>• Other functions of your choice (but the above mentioned are compulsory)</li> </ul>
8	Phone Directory
	Design a simple phone directory to store contact details of 1000 persons. The functions covered are: <ul style="list-style-type: none"> <li>• Add new contact</li> <li>• View all contacts</li> <li>• Search a contact</li> <li>• Modify a contact</li> <li>• Delete a contact</li> <li>• Exit</li> </ul>
9	Power Plant Control Simulation
	Design a simulation of how the various units of a power plant are monitored and controlled. Use custom functions to show which parameter is controlled in what manner, and also simulate emergency situations and how they are brought under control, with text-based outputs.
10	Bus Reservation
	Design a simple reservation system for bus travel. An agent should be able to:

	<ul style="list-style-type: none"> <li>• Login with appropriate ID and password</li> <li>• Check buses and their seating and bookings</li> <li>• Book tickets</li> <li>• Update or cancel tickets</li> <li>• Print bills</li> <li>• Exit</li> </ul>
11	Company Employee Management
	<p>Design a simple interface for a company HR manager to handle their employee database. The functions covered in the program are:</p> <ul style="list-style-type: none"> <li>• Build an employee table for 20 employees</li> <li>• Insert new records</li> <li>• Update records</li> <li>• Delete records</li> <li>• Search records</li> <li>• View records</li> <li>• Exit</li> </ul>
12	Fuel Station Management
	<p>Design a simple system that manages the billing of a fuel station (petrol pump). The user can perform the following functions:</p> <ul style="list-style-type: none"> <li>• Select fuel</li> <li>• Enter car details</li> <li>• Show filling of a tank</li> <li>• Print bill</li> <li>• Exit</li> </ul>
13	Calendar app
	<p>Design a simple calendar display that allows a user to view various months of different years from 2000 to 2025. The user should be able to:</p> <ul style="list-style-type: none"> <li>• View different months</li> <li>• Search for a date</li> <li>• See events on that date</li> <li>• Book events on some date</li> <li>• Exit</li> </ul>
14	Multi-level Parking Lot Interface
	<p>Design a simple system that manages a multi-level parking lot. The user can perform the following tasks:</p> <ul style="list-style-type: none"> <li>• Login with appropriate ID and password</li> <li>• Enter new vehicle details and issue parking ticket</li> <li>• Check slots and levels (floors) for parking space</li> <li>• Print parking exit bill</li> <li>• Exit</li> </ul>
15	Hostel Management
	<p>Design an interface that manages hostel bookings. The user should be able to:</p> <ul style="list-style-type: none"> <li>• Check lodgings</li> <li>• Update lodgings</li> <li>• Add new reservations</li> <li>• Check hostel resources being issued</li> <li>• Check-in and Check-out of customers</li> </ul>
16	Electronic Voting Machine Simulation

	Design the simulation of an Electronic Voting Machine. The system should allow two users – one is the voter who will see the candidates and cast their vote; the other is the counter who will see the updating score.
17	Pet Adoption Center Management
	<p>Design a simple management system for a pet center, where two users can login –</p> <ul style="list-style-type: none"> <li>• Customer: <ul style="list-style-type: none"> <li>○ Browse pets</li> <li>○ Adopt pets</li> <li>○ Receive adoption certificate</li> <li>○ Exit</li> </ul> </li> <li>• Manager: <ul style="list-style-type: none"> <li>○ Add pets</li> <li>○ Update status of pets</li> <li>○ Delete records</li> <li>○ Exit</li> </ul> </li> </ul>
18	Police FIR registry
	<p>Design a simple registry for a police station where an agent can:</p> <ul style="list-style-type: none"> <li>• View existing FIRs</li> <li>• View solved cases</li> <li>• Add new FIR</li> <li>• Update FIR</li> <li>• Resolve FIR</li> <li>• Depute personnel</li> <li>• Exit.</li> </ul>
19	Room Automation Simulation
	<p>Design a simulation of a home room automation system, where various appliances are automated and controlled using a home network. Use custom functions and random running instances. Control the following appliances and show the monitoring with text:</p> <ul style="list-style-type: none"> <li>• Air conditioner</li> <li>• Fan</li> <li>• Television</li> <li>• Two lights</li> </ul>
20	University Management
	<p>Design a simple university management system, where a user can login and do the following tasks:</p> <ul style="list-style-type: none"> <li>• Add student record</li> <li>• Update records</li> <li>• Search and view records</li> <li>• Delete records</li> <li>• Display a menu for these functions along with exit</li> </ul>
21	Movie Ticket Booking
	<p>Design a simple booking system that allows a user to book tickets for a movie theatre. The following functions are to be covered:</p> <ul style="list-style-type: none"> <li>• Login</li> <li>• Search movie by name/genre/timing</li> <li>• View schedules</li> <li>• Book tickets by selecting seats and timing</li> <li>• Add perks</li> <li>• Print ticket and bill</li> <li>• Exit</li> </ul>

22	Unit System Converter
	<p>Design a unit converter that converts metric system units to imperial units and vice versa as per preference. The user should be able to:</p> <ul style="list-style-type: none"> <li>• View a menu for <ul style="list-style-type: none"> <li>○ Metric to imperial conversion</li> <li>○ Imperial to metric conversion</li> </ul> </li> <li>• Choose quantity to convert: <ul style="list-style-type: none"> <li>○ Length</li> <li>○ Weight</li> <li>○ Volume</li> <li>○ Others (optional)</li> </ul> </li> <li>• Provide appropriate conversion options and calculate.</li> <li>• Exit</li> </ul>
23	Food Ordering System
	<p>Design a simple restaurant system that allows a user to:</p> <ul style="list-style-type: none"> <li>• View the menu</li> <li>• Choose dishes</li> <li>• Choose quantity</li> <li>• Provide other customizations</li> <li>• Get the bill</li> <li>• Exit</li> </ul>
24	Fruit Shop Billing System
	<p>Design a simple text-based Point-Of-Sales system for the cashier of a fruit shop. The functions covered are:</p> <ul style="list-style-type: none"> <li>• Login</li> <li>• Check Prices</li> <li>• Update Prices</li> <li>• Make new bill</li> <li>• Add to inventory</li> <li>• Delete from inventory</li> <li>• Print Report</li> <li>• Exit</li> </ul>
25	Wordle/Mastermind
	<p>Design a simple interface to simulate either the letter-based game Wordle or the color-based game Mastermind. The functions covered are:</p> <ul style="list-style-type: none"> <li>• Start a new game</li> <li>• Show the game board</li> <li>• Display the progress in the game</li> <li>• Show the final try and win/loss</li> <li>• Show score</li> <li>• Refresh scores</li> <li>• Exit</li> </ul>