

ELECTRICITY BILL GENERATOR

Name: Lakshya Bisht

SAP ID: 590027070

B.Tech CSE

UPES

PROBLEM DEFINITION

Electricity bill calculation is generally done through complex tariff rules including slabs, fixed charges, government duties, and late payment charges.

Manually calculating these charges is time-consuming and often leads to errors.

Many users also find it difficult to track and manage electricity usage records for multiple consumers.

OBJECTIVE OF THE PROJECT

Project Objectives

- The objective of this project is to:
 - Automate electricity bill calculation
 - Reduce human calculation errors
 - Apply real tariff slabs in logic
 - Generate a formatted electricity bill
 - Store and manage multiple customers
 - Enable searching customer data
 - Improve understanding of C programming concepts

SCOPE OF THE PROJECT

Project Scope

This project demonstrates:

- Slab-wise electricity billing
 - Customer data management
 - Late payment charge handling
 - Searching by ID or Name
 - Structured programming design
 - Function modularity
 - User input validation
 - Personalized bill generation
- This project simulates real-world billing used by power companies under residential connections.

CONCEPTS USED

C Programming Concepts Implemented

- The following concepts are implemented:
 - Structures
 - Functions
 - Arrays
 - Conditional statements
 - File handling
 - String handling
 - Input validation
 - Modular programming
 - Pointers to structures

DATA STRUCTURE USED

Customer information is stored using a structure:

- `typedef struct`
- `{`
- `int id;`
- `char name[50];`
- `float units;`
- `float load; //Connected load in kW`
- `float energyCharge;`
- `float fixedCharge;`
- `float duty; //Electricity duty`
- `float surcharge;`
- `float totalBill;`
- `} Customer;`

BILL CALCULATION LOGIC

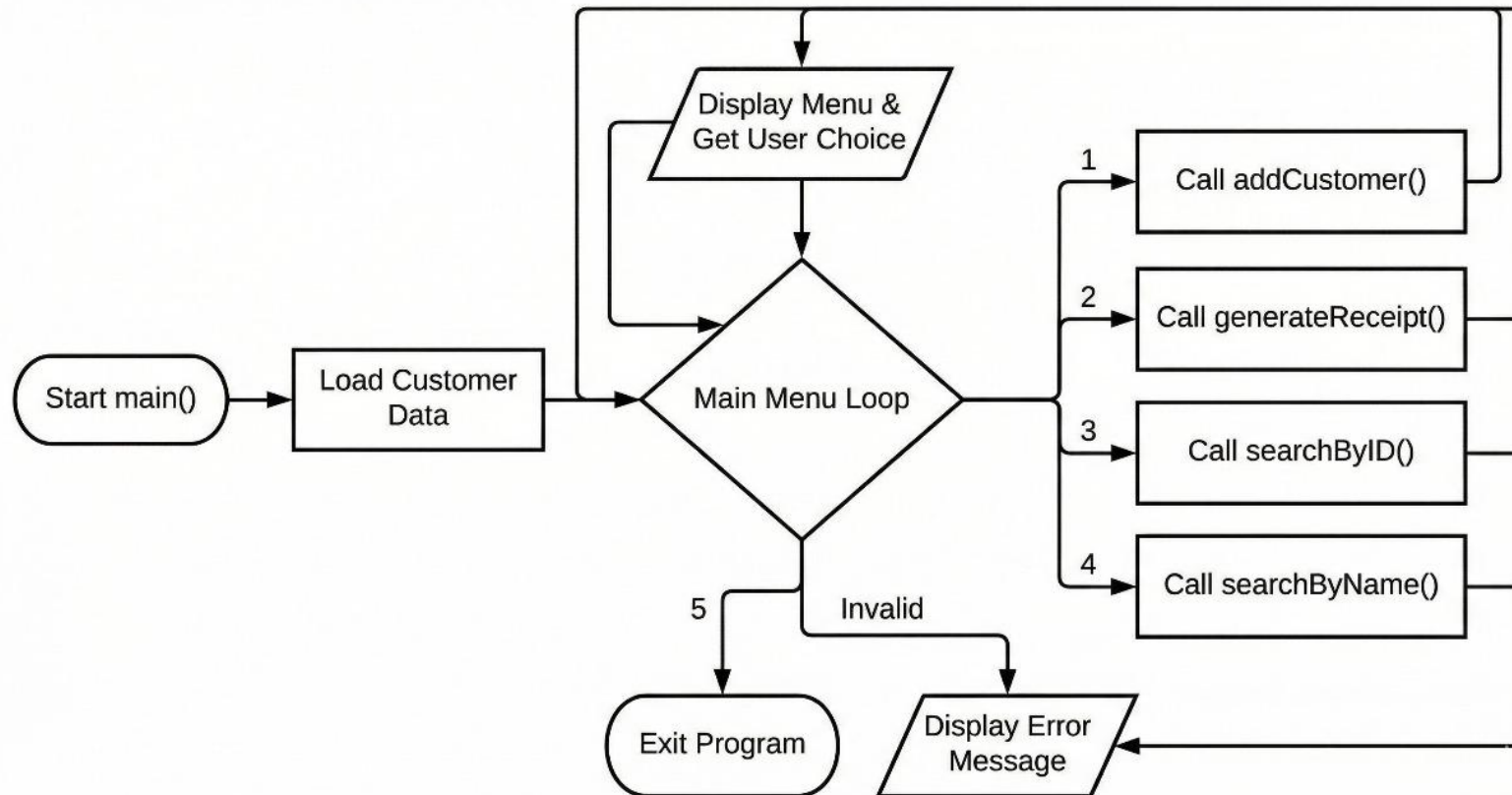
- **Tariff Calculation Logic**
- **Energy Charges (Slabs):**
 - Below 100 units → ₹3.65/unit
 - 101–200 units → ₹5.25/unit
 - 201–400 units → ₹7.15/unit
 - Above 400 units → ₹7.80/unit
- **Additional Charges:**
 - Fixed charge based on load
 - Government duty (15%)
 - Surcharge (₹0.10/unit)
 - Late payment fee (1.25%)

PROGRAM FLOW

Program Flow

1. User selects option from menu
2. Customer inputs details
3. Bill is calculated
4. Data is stored
5. Receipt is generated
6. Search operations are performed
7. Output displayed in formatted form

FLOWCHART



FUNCTIONS USED

FUNCTION

- `calculateBill()`
- `addCustomer()`
- `generateReceipt()`
- `searchByID()`
- `searchByName()`

PURPOSE

- Calculates complete bill
- Stores user data
- Displays formatted bill
- Finds customer by ID
- Finds customer by name

SEARCH FEATURE

- This project allows:
 - Searching customer by ID
 - Searching customer by Name
 - Quick record access
 - Improved data management
- This makes the project closer to real utility software.

SAMPLE OUTPUT-I

ELECTRICITY BILL SYSTEM MENU:

1. Add New Customer
2. Generate Electricity Bill
3. Search customer records by ID
4. Search customer records by Name
5. Exit

Please select an option: 1

ADD NEW CUSTOMER

Enter Customer ID: 590027070

Enter Customer Name: Lakshya Bisht

Enter Units Consumed: 404.77

Enter Load Capacity (in kW): 4.5

Is it late payment? (1=yes, 0=no): 1

Customer added successfully.

ELECTRICITY BILL SYSTEM MENU:

1. Add New Customer
2. Generate Electricity Bill
3. Search customer records by ID
4. Search customer records by Name
5. Exit

Please select an option: 2

Enter Customer ID to generate receipt: 590027070

UTTARAKHAND POWER CORPORATION LTD. (UPCL)

Customer ID : 590027070

Customer Name : Lakshya Bisht

Connected Load: 4.50 kW

Units Consumed: 404.77 units

Bill Breakdown

Energy Charge : INR 2357.21

Fixed Charge : INR 450.00

Duty (15%) : INR 353.58

Surcharge : INR 40.48

TOTAL BILL : INR 3241.28

Thank you for using UPCL Billing System

SAMPLE OUTPUT-2

ELECTRICITY BILL SYSTEM MENU:

1. Add New Customer
2. Generate Electricity Bill
3. Search customer records by ID
4. Search customer records by Name
5. Exit

Please select an option: 1

ADD NEW CUSTOMER

Enter Customer ID: 590027070

Enter Customer Name: Lakshya Bisht

Enter Units Consumed: 411.72

Enter Load Capacity (in kW): 4.75

Is it late payment? (1=yes, 0=no): 1

Customer added successfully.

ELECTRICITY BILL SYSTEM MENU:

1. Add New Customer
2. Generate Electricity Bill
3. Search customer records by ID
4. Search customer records by Name
5. Exit

Please select an option: 3

Enter Customer ID to search: 590027070

Customer Found:

ID: 590027070

Name: Lakshya Bisht

Units: 411.72

Load: 4.75 kW

Total Bill: INR 3330.42

ELECTRICITY BILL SYSTEM MENU:

1. Add New Customer
2. Generate Electricity Bill
3. Search customer records by ID
4. Search customer records by Name
5. Exit

ERROR HANDLING

Input Validation

- Handled cases:
 - Duplicate ID
 - Negative units
 - Invalid load capacity
 - Wrong late-payment input
 - Limit check for customers
- This makes the program reliable and stable.

NOVELTY

Project Highlights:

- • Slab implementation
- Realistic tariff
- Late fee system
- Search feature
- Bill breakdown
- Customer management system
- Professional user interface
- Professional GitHub repository

LIMITATIONS

Current Limitations

- • No monthly billing history
 - No date and time on bill
- • No Graphical Interface (GUI)
 - Single month billing

FUTURE SCOPE

Future Enhancements:

- Billing history
- Export to PDF
- GUI application
- Date and Time stamp on bill
- Admin panel (password protected)
- Online Payment Feature

REFERENCES

- • Classroom lectures/notes
- Let Us C Book
- UPCL Official Tariff
- Youtube lectures
- Project discussion with faculty

CONCLUSION

This project helped me understand how real-world electricity billing works and how it can be implemented using C programming. Through this project, I also improved my understanding of structures, functions, and logical conditions in C. Handling customer records and generating formatted bills made the project more practical and meaningful.

Overall, this project was a good learning experience and helped me gain confidence in implementing logic-based programs in C.

THANK YOU