Experiment No. 3
Automated Discount and Clothing Billing System
Using Decision-Making Constructs in C

Aim: Developing an automated billing and discount system for a clothing store. The system needs to determine the **discount based on customer type and total purchase amount, and calculate the final bill accordingly**. Different customers receive different discounts based on the conditions set by the store. Demonstrates all three concepts (if-else, if-else if-else, switch-case)

Learning Outcomes:

- Enhance problem-solving skills by translating store discount policies into logical programming structures.
- Design an automated billing system that dynamically calculates discounts based on customer type (e.g., Regular, Premium, VIP) and purchase amount.
- Apply conditional logic in programming by implementing if-else, if-else if-else, and switch-case statements to handle real-world billing scenarios.

Theory: In real-world applications like retail and e-commerce, billing systems often need to calculate the final payable amount after applying discounts. These discounts may vary depending on the type of customer (e.g., Regular, Premium, VIP) and the total purchase amount. To implement such rules in programming, we use decision-making constructs that allow the program to choose different actions based on conditions.

Decision-Making Constructs: Decision-making constructs are control structures that help a program take different paths of execution depending on given conditions. The main constructs demonstrated in this experiment are:

a) if-else

- The if-else statement is used for binary (yes/no or true/false) decisions.
- Example use case: Checking if the total purchase amount qualifies for a basic discount.

```
if (amount > 500)
  discount = 0.05;
else
  discount = 0.00;
```

b) if-else if-else

- The if-else if-else structure is used when there are multiple ranges or conditions to evaluate.

- Example use case: Applying different discount percentages based on purchase amount slabs.

```
if (amount > 5000)
  discount = 0.20;
else if (amount > 2000)
  discount = 0.10;
else
  discount = 0.05;
```

c) switch-case

- The switch-case statement is useful when a decision is based on a fixed set of categories or values.
- Example use case: Assigning discounts based on customer type (1-Regular, 2-Premium, 3-VIP).

```
switch(customerType) {
  case 1:
           discount = 0.05;
           break;
  case 2:
           discount = 0.10;
           break;
  case 3:
           discount = 0.20;
           break;
  default: discount = 0%;
}
```

Program Code:

Students are expected to implement this experiment by writing the complete C program for the Clothing billing system, making use of the various decision-making statements in C.

Output:

Students are expected to execute the program and provide two sample outputs for the same code to demonstrate its functionality with different input values.

Conclusion: This experiment bridges theory and practice by showing how real-world business rules (discount policies) can be automated using decision-making statements. The outcome is a functional billing system that not only calculates the final payable amount but also illustrates the importance of selecting the right control structure for efficient program design.