Case Scenario: XYZ Software Solutions is developing an employee management system using C++. The system should manage employee records, including their names, IDs, and salaries. The company wants to implement Object-Oriented Programming principles such as encapsulation, inheritance, and polymorphism to make the system more efficient and maintainable.

As a software developer, you are required to design and implement the core functionalities of the employee management system using OOP principles in C++.

Assignment Tasks:

1.Introduction (5 Marks)

Define Object-Oriented Programming and its significance.

-It is a computer programming model that organizes software design around data or objects.

-It helps model any real world entity into software block.

-Enables code visibility.

-Enhances modularity.

-Enhances maintainability.

Explain the key OOP principles (Encapsulation, Inheritance, Polymorphism, and Abstraction) with examples.

-Encapsulation-restores data access to an objects and allows manipulation example class with private attributes.

-Inheritance-enables code reusability by allowing one class to inherit from another example inheritance with constructors.

-Polymorphism-allows same method to have different implementations in derived classes example operator overloading.

-Abstraction-allows hiding implementation details and exposing only necessary functionality example a remote control to operate a tv without knowing its interna circuitry.

2.Analysis of the Case Scenario (5 Marks)

Identify the key functional requirements of the employee management system.

-To implement functionalities like search ,add and display of employees details.

-To support employee types with specific attributes.

-To manage employee records with attributes such as name,IDs and salary.

Discuss how OOP principles can be applied to design the system effectively.

-encapsulation-protecting data from unauthorized users through information hiding.

-Inheritance-allows new classes to inherit properties from existing ones.

-Polymorphism-used by developers to write more efficient code and redefine methods for derived classes.

4. Conclusion and Future Recommendations (5 Marks)

Summarize the importance of OOP in software development.

-Enhances code modularity ,reusability and scalabity.

-Scalability-makes it easier to scale a program.

-Simplifies maintainance and debugging.

-Encourages organized and organized structured programming.

Provide recommendations on how the employee management system can be further improved using advanced OOP concepts.

-Database intergration -to store and retrieve employee records effectively.

-Graphical User Interface-to improve user interaction.

-Advanced polymorphism-implement function overloading for better flexibility.

-Use of file handling-to persist employee data across program executions.