1. **What is OOP? List OOP concepts**

* Object-Oriented Programming aims to implement real-world entities like inheritance, hiding, polymorphism etc. in programming.
* The main aim of OOP is to bind together the data and the functions that operate on them so that no other part of the code can access this data except that function.
* Here are some key OOP concepts:-

1. Class
2. Object
3. Encapsulation
4. Inheritance
5. Polymorphism
6. Abstraction
7. Constructor
8. Destructor
9. Method Overloading
10. Method Overriding
11. **What is different between OOP and POP?**

|  |  |  |
| --- | --- | --- |
| **Feature** | **Object-Oriented Programming (c++)** | **Procedural Programming (C Language)** |
| Key Focus | Object and their interactions | Procedures and functions |
| Data Management | Data and behavior are encapsulated in objects | Data and behavior are separate entities |
| Abstraction | Encourages the use of abstract classes and interfaces | Does not emphasize abstraction |
| Inheritance | Supports inheritance, allowing classes to inherit properties and methods | Does not support inheritance |
| Polymorphism | Allows objects of different types to be treated as the same type | Does not provide inherent polymorphism |
| Code Reusability | High level of code reusability through inheritance and composition | Relies on functions and subroutines for code reusability |
| Code organization | Follows a modular approach, with objects as self-contained modules | Relies on functions and procedures for code organization |
| Flexibility | Provides flexibility through polymorphism and dynamic binding | Relies on structured programming, offering less flexibility |
| Complexity Management | Encourages managing complexity through encapsulation and abstraction | Manages complexity through modular code organization and stepwise design |
| Real-World Modeling | Well-suited for modeling real-world entities and their interactions | May not align well with real-world modeling, focusing more on processes |