**Question: 01**

**Differences**

| **C++** | **Java** |
| --- | --- |
| **Memory Management:** | |
| Requires manual memory management using pointers and dynamic memory allocation (e.g., new and delete). | Utilizes automatic memory management through garbage collection, which helps prevent memory leaks. |
| **Pointers and References:** | |
| Provides pointers and references, allowing direct memory manipulation and pointer arithmetic. | Uses references for objects but does not have pointers. It provides a safer memory model. |
| **Multiple Inheritance:** | |
| Supports multiple inheritance, allowing a class to inherit from more than one parent class. | Implements single inheritance for classes but allows multiple inheritance for interfaces through the use of interfaces. |
| **Operator Overloading:** | |
| Supports operator overloading, allowing you to define custom behaviors for operators. | Does not support operator overloading, which can make code more predictable but also limits flexibility in some cases. |
| **Compilation and Execution:** | |
| Requires a separate compilation step (compilation to binary code) before execution. | Compiles to bytecode and uses a Just-In-Time (JIT) compiler at runtime, which can lead to slower startup times but can optimize code for the specific platform. |

**Similarities**

| **Object-Oriented Programming (OOP):** |
| --- |
| Both C++ and Java are object-oriented languages, which means they support the principles of encapsulation, inheritance, and polymorphism. |
| **Classes and Objects:** |
| Both languages use classes and objects as fundamental building blocks for designing software. |
| **Multi-threading:** |
| Both C++ and Java support multi-threading and provide libraries for concurrent programming, allowing the creation and management of multiple threads of execution. |
| **Exception Handling:** |
| C++ and Java provide mechanisms for handling exceptions to gracefully manage runtime errors in programs. |
| **Method Overloading:** |
| They both allow method overloading, enabling multiple methods with the same name but different parameter lists within a class. |