**Integration and System Testing (6 Marks)**

**1. Integration Testing (3 marks)**

**Definition:**

Integration testing checks the **interaction between modules** after they are individually tested (unit tested). It ensures that the **modules work together** as intended.

**Purpose:**

* To detect **interface errors**, **data mismatches**, or **incorrect module interactions**.

**Approaches:**

* **Top-down:** Test from the top-level module to lower ones.
* **Bottom-up:** Test from the bottom modules upwards.
* **Big Bang:** All modules are combined and tested at once.
* **Incremental:** Modules are integrated and tested gradually.

**Example/Application:**

In an **online shopping system**, after testing modules like "Cart", "Payment", and "Inventory" individually, integration testing ensures that when a user adds a product and checks out, data flows correctly from cart to payment and inventory updates correctly.

**2. System Testing (3 marks)**

**Definition:**

System testing tests the **entire software system** as a whole to ensure it meets the specified requirements.

**Purpose:**

* To validate **functional and non-functional requirements**.
* Covers performance, security, usability, and reliability.

**Types of system testing:**

* **Functional Testing**
* **Performance Testing**
* **Security Testing**
* **Compatibility Testing**

**Example/Application:**

Continuing with the **online shopping system**, system testing would simulate a real user placing an order from start to finish, including searching products, adding to cart, payment, order confirmation, and email receipt — ensuring the full process works end-to-end.