

# Class – 9 (Computer Applications)

## Chapter 1

### Principles of Object-Oriented Programming –

#### A. Tick (✓) the correct answer

1. Which of the following is a pillar of Object-Oriented programming?  
**Answer:** c. Polymorphism
2. \_\_\_\_\_ allows multiple objects of different subclasses to be treated as objects of a single superclass.  
**Answer:** c. Polymorphism
3. Abstraction and data hiding maintains the \_\_\_\_\_ of data as only necessary data is provided.  
**Answer:** c. Security
4. The other name of Subclass is \_\_\_\_\_ class.  
**Answer:** b. Derived
5. \_\_\_\_\_ is about hiding unwanted details and showing only the essential information.  
**Answer:** a. Abstraction

#### B. Fill in the blanks

1. **Procedure-Oriented Programming (POP)** has global data sharing of functions.
2. **Compiler** converts source code into the object code.
3. **Procedure-Oriented Programming (POP)** divides the whole problem into smaller programs known as functions or methods.
4. **Programming paradigm** refers to an approach used to classify the programming languages based on their features.
5. C++ is an example of **Object-Oriented Programming (OOP)**.

#### C. Answer the following questions

##### 1. Write the difference between POP and OOP.

POP	OOP
Focuses on procedures/functions	Focuses on objects
Uses Top-Down approach	Uses Bottom-Up approach
Less secure	More secure
Data is less protected	Data is hidden (encapsulation)
Example: C, Fortran	Example: C++, Java

##### 2. Define Polymorphism with a real-life example.

**Answer:**

Polymorphism means “many forms”. It allows an object to behave differently in different situations.

**Example:**

A car can be used as a private car, taxi, or school vehicle. Same object (car) behaves differently.

### **3. Write disadvantages of Object-Oriented Programming.**

**Answer:**

- Requires more lines of code
- Needs more memory
- Execution is slower than POP
- Requires proper planning and design

### **4. Write the difference between Polymorphism and Encapsulation.**

#### **Polymorphism**

Means many forms

Same object behaves differently

Focuses on behavior

#### **Encapsulation**

Means data hiding

Data and methods are wrapped together

Focuses on security

### **5. Define Inheritance with a real-life example.**

**Answer:**

Inheritance is the process of acquiring properties of one class into another class.

**Example:**

A child inherits features like eyes, hair, etc., from parents.

### **6. Write the difference between Assembly language and Machine language.**

#### **Machine Language**

Uses 0s and 1s

Hard to understand

No translator needed

Machine dependent

#### **Assembly Language**

Uses mnemonics

Easier than machine language

Needs assembler

Machine dependent

### **7. Write down two advantages of Polymorphism.**

**Answer:**

- Improves code flexibility
- Allows code reuse

### **8. Write down any two disadvantages of Machine-Level Language.**

Answer:

- Difficult to understand
- Difficult to debug

## D. Picture Study

1. The above diagram represents:  
**Answer:** b. Object-Oriented Programming
2. Principles on which the above programming works on is/are \_\_\_\_\_  
**Answer:** d. All of these
3. \_\_\_\_\_ is the example of the above.  
**Answer:** a. C++
4. The above uses the concept of \_\_\_\_\_ approach.  
**Answer:** a. Bottom-Up
5. \_\_\_\_\_ is a feature of the above picture which generate a new class from another already created class.  
**Answer:** b. Inheritance

## E. Case Study

1. The above principle is known as:  
**Answer:** b. Inheritance
2. The car is called \_\_\_\_\_ in the given case study.  
**Answer:** a. Subclass
3. \_\_\_\_\_ class uses the extends keyword.  
**Answer:** a. Subclass
4. The concept of \_\_\_\_\_ goes with the Inheritance principle.  
**Answer:** b. Reusability
5. Extends keyword is used to inherit the properties of \_\_\_\_\_ to \_\_\_\_\_.  
**Answer:** a. Superclass to subclass

## F. Assertion and Reasoning Based Questions

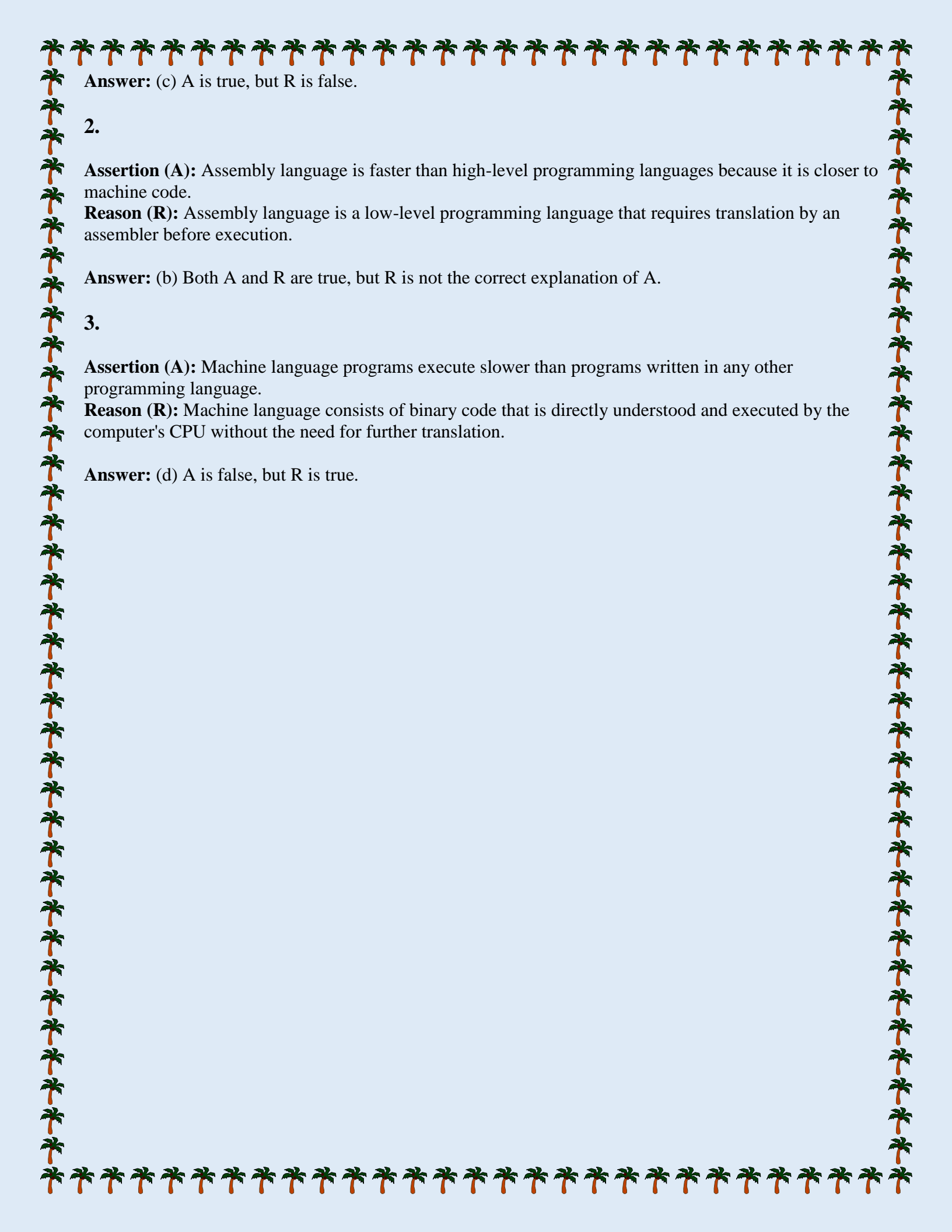
Options:

- (a) Both A and R are true, and R is the correct explanation of A.
- (b) Both A and R are true, but R is not the correct explanation of A.
- (c) A is true, but R is false.
- (d) A is false, but R is true.

1.

**Assertion (A):** A compiler translates the entire source code of a program into machine code before execution.

**Reason (R):** Interpreters also translate the entire source code into machine code before execution, just like compilers.



**Answer:** (c) A is true, but R is false.

2.

**Assertion (A):** Assembly language is faster than high-level programming languages because it is closer to machine code.

**Reason (R):** Assembly language is a low-level programming language that requires translation by an assembler before execution.

**Answer:** (b) Both A and R are true, but R is not the correct explanation of A.

3.

**Assertion (A):** Machine language programs execute slower than programs written in any other programming language.

**Reason (R):** Machine language consists of binary code that is directly understood and executed by the computer's CPU without the need for further translation.

**Answer:** (d) A is false, but R is true.