

**BACHELOR OF TECHNOLOGY (CBCS) (2021-COURSE)**  
**B. Tech. Sem - III Computer Science & Engineering : WINTER: 2025**  
**SUBJECT: ITC-I: OBJECT ORIENTED METHODOLOGY**

Day : Friday  
Date : 19/12/2025

**W-25314-2025**

Time : 10:00 AM-01:00 PM  
Max. Marks : 60

**N.B.**

- 1) All questions are **COMPULSORY**.
- 2) Figures to the **RIGHT** indicate **FULL** marks.
- 3) Assume suitable data **WHEREVER** necessary.
- 4) Draw neat diagrams **WHEREVER** necessary.
- 5) Use of non-programmable **CALCULATOR** is allowed.

**Q.1** Explain the internal structure of a basic Java program with an example. Describe the role of the main() method. (10)

**OR**

**Q.1** Discuss decision-making and branching statements in Java with examples. How does switch differ from if-else? (10)

**Q.2** Define encapsulation. How is encapsulation achieved in C++ using classes and access modifiers? Illustrate with an example. (10)

**OR**

**Q.2** Create a class called Student with data members like rollNo, name, and marks. Write appropriate methods to:  
i) Read student details.  
ii) Display student details.  
iii) Return grade based on marks. (10)

**Q.3** Compare and contrast String, StringBuffer, and StringBuilder classes in Java in terms of mutability and performance. Include code snippets to support your answer. (10)

**OR**

**Q.3** How does Java handle memory management using garbage collection? Explain the concept of finalizers and how they relate to object cleanup. (10)

**Q.4** Describe the role of constructors in inheritance. How is a base class constructor invoked from a derived class? Explain with a suitable code example using the super keyword. (10)

**OR**

**Q.4** Define polymorphism in object-oriented programming. Differentiate between compile-time and run-time polymorphism with relevant Java code examples. (10)

**Q.5** Describe inter-thread communication in Java. Explain how wait(), notify(), and notifyAll() methods work with a program example that shows two threads communicating through a shared resource. (10)

**OR**

**Q.5** Explain the concept of exception propagation in Java. Write a program to show how an exception is propagated through the call stack and caught in an outer method. (10)

**Q.6** Describe the Applet architecture in Java. How does the browser or applet viewer interact with an applet during its execution? (10)

**OR**

**Q.6** Explain how advanced Swing components such as JTree and JTable can be used to enhance a Java application's user interface. Provide examples to illustrate their use and structure. (10)

\* \* \* \* \*