Ø=ÜØ AI-Generated Question Paper

1. **Java Programming Examination** **Section A – Multiple Choice Questions**

2. Which Java keyword is used to prevent method overriding? A) final B) static C) abstract D) private

3. What will `System.out.println(10 + 20 + "30")` output? A) 3030 B) 102030 C) 30 D) 303030

4. Which collection class is synchronized by default? A) ArrayList B) LinkedList C) Vector D) HashSet

5. Identify the correct interface for lambda expressions with a single abstract method: A) Serializable B) Cloneable C) Runnable D) Comparable

6. In exception handling, which block executes regardless of an exception? A) catch B) throw C) finally D) throws

7. What is the default value of a boolean instance variable? A) true B) false C) null D) 0

8. Which annotation marks a method for overriding a superclass method? A) @Deprecated B) @SuppressWarnings C) @Override D) @SafeVarargs

9. Which class provides thread-safe atomic operations? A) java.util.concurrent.atomic.AtomicInteger B) java.util.concurrent.ThreadPool C) java.util.concurrent.locks.ReentrantLock D) java.util.concurrent.Semaphore

10. What does the `transient` modifier achieve? A) Prevents variable serialization B) Enables variable synchronization C) Marks a constant value D) Restricts variable scope

11. Which design pattern uses a private constructor to control object creation? A) Factory Method B) Observer C) Singleton D) Decorator **Section B – Short Answer Questions**

12. Differentiate between `String`, `StringBuilder`, and `StringBuffer` in terms of mutability and thread safety.

13. Explain the purpose of the `equals()` and `hashCode()` methods. Why must they be overridden together?

14. Describe the difference between checked and unchecked exceptions. Provide one example of each.

15. How does polymorphism manifest in Java? Illustrate with a code snippet using method overriding.

16. What is type erasure in Java generics? Provide a limitation it imposes. **Section C – Long Answer Questions**

17. Discuss the principles of encapsulation and abstraction in object-oriented programming. Explain how Java implements these concepts using access modifiers and interfaces/abstract classes. Include code examples to justify your answer.

18. Compare `ArrayList` and `LinkedList` in terms of internal data structures, performance for common operations (insertion, deletion, traversal), and memory overhead. Recommend scenarios for using each.

19. Elaborate on the producer-consumer problem in multithreading. Design a solution using `wait()`, `notify()`, and a shared buffer. Highlight potential pitfalls (e.g., deadlock) and mitigation strategies.