

**Howard University**  
**College of Engineering and Architecture**  
**Department of Electrical Engineering & Computer Science**  
**Large Scale / Object-Oriented Programming**

**Midterm Exam – Part I**  
March 14, 2024

Name : Araj Shah

**This exam is two parts:**

**Part I:        70 pts. (T/F, multiple choice, essays)**

**Part II:       30 pts. (programming)**

**Instructions:**

- Please return this exam in Word format, do not convert to pdf.
- Submit completed exam to your github repository. Create package:  
org.howard.edu.lsp.midterm  
Verify that the commit completed successfully
- OPEN BOOK, OPEN NOTES. THERE IS NO COLLABORATION ON THIS EXAM

**Section 1: True/False. Highlight or write your answer. Each question is 1 pt.**

1. **T F** The ability to hide the implementation details of an object is called encapsulation
2. **T F** Java code is a compiled language that creates binary code for a specific architecture and must be recompiled when moved to another machine
3. **T F** Polymorphism works in Java because method calls are bound to their method definitions at compile time
4. **T F** Every object in Java explicitly derives from class Object using extends.
5. **T F** A class in Java may implement only one interface
6. **T F** In Java, when an instance of a class, or object, is specified as a parameter to a method, a reference to the said object is passed to the method
7. **T F** When designing a class, conceptually, each class should be designed to have multiple goals so that your overall design can have less classes
8. **T F** In Java, it is impossible to have memory leaks because of garbage collection
9. **T F** A Java class can extend from multiple classes
10. **T F** Testing does not ensure that our programs will never have any bugs
11. **T F** According to principles of encapsulation, instance variables should always be declared as public
12. **T F** Google naming conventions recommend that a Java class should begin with a lowercase letter and methods with an uppercase letter
13. **T F** A subclass can add behavior that is not present in the superclass.
14. **T F** If B is a subclass of A, then a B object may always be assigned to a variable of type A
15. **T F** If class A extends class B, class A is a subclass of B and B is a superclass of A
16. **T F** A constructor can be invoked multiple times once an object is instantiated
17. **T F** A checked exception represents an error that a program has to handle
18. **T F** In Java, the **this** keyword can be used to refer to both the current instance of the class and other instances of the class within the same scope
19. **T F** Methods are said to be *overridden* if they are in the same scope and have the same names but different signatures
20. **T F** A method that *overrides* another must have the same name but a different signature
21. **T F** A *static* method can refer to any instance variable of the class
22. **T F** Refactoring code involves changing the functionality of the code without modifying its external behavior
23. **T F** A Java method returns void if it does not return a value
24. **T F** An ArrayList in Java will not automatically change its size as needed.

25. **T F** Polymorphism works in Java because method calls are bound to their method definitions at compile time.

**Section 2: Multiple Choice, type answer below each question. Each question is 1 pt.**

1. Which of the following is also known as run-time binding or late binding?
  - a) Dynamic typing
  - b) Dynamic loading
  - c) **Dynamic binding**
  - d) Data hiding
2. Which principle suggests that each module should perform a single, well-defined task?
  - a) **High Cohesion**
  - b) Low Coupling
  - c) Stepwise Refinement
  - d) Information Hiding
3. A good design should strive for:
  - a) a tight cohesion and a tight coupling between modules
  - b) **a tight cohesion and a loose coupling between modules**
  - c) a loose cohesion and a tight coupling between modules
  - d) a loose cohesion and a loose coupling between modules
4. Which of the following is true about composition in object-oriented programming?
  - a) It is a form of inheritance.
  - b) It allows objects to inherit properties and behaviors from another class.
  - c) **It is a way of creating complex objects by combining simpler ones.**
  - d) It is used to hide the implementation details of a class.
5. In Java, when can the garbage collector collect the memory allocated to an object?
  - a) **when it can prove that there is no reference to that object from any other object**
  - b) when the reference initialized with "new" goes out of scope
  - c) when the method in which the object was created returns
  - d) only when the program terminates
  - e) never, java puts the responsibility on developers to deallocate the memory programmatically.



6. What is the purpose of the Single Responsibility Principle (SRP) in object-oriented design?
- a) To ensure that each class has only one method
  - b) To ensure that each class has only one instance variable
  - c) To ensure that each class has only one responsibility and reason to change
  - d) To ensure that each class has only one constructor
7. What is an abstract class?
- a) A class that has direct instances, but whose descendants may have direct instances
  - b) A class that has direct instances, but whose descendants may not have direct instances
  - c) A class that has no direct instances, but whose descendants may have direct instances
  - d) All of the mentioned
8. What is true about "has-a" and "is-a" relationships? (Choose two)
- a) instance variables can be used when creating a has-a relationship
  - b) inheritance represents an is-a relationship
  - c) inheritance represents a has-a relationship
  - d) instances must be used when creating a has-a relationship
9. How does Arthur Riel's principle of "information hiding" contribute to better software design?
- a) By minimizing the dependencies between modules
  - b) By encapsulating the implementation details of a module
  - c) By ensuring that each module has only one responsibility
  - d) By maximizing cohesion within modules
10. According to Arthur Riel's design principles, what should be the relationship between modules in a well-designed system?
- a) Modules should be tightly coupled to each other
  - b) Modules should be loosely coupled to each other
  - c) Modules should have low cohesion
  - d) Modules should be independent of each other
11. Which principle suggests that each module should perform a single, well-defined task?
- a) High Cohesion
  - b) Low Coupling
  - c) Stepwise Refinement
  - d) Information Hiding

12. What principle suggests that subclasses should be substitutable for their base classes without affecting the correctness of the program?

- a) Open/Closed Principle
- b) Liskov Substitution Principle
- c) Single Responsibility Principle
- d) Interface Segregation Principle

13. What is the primary purpose of inheritance in object-oriented programming according to Arthur Riel's principles?

- a) To increase the complexity of the software system
- b) To promote code reuse and minimize redundancy
- c) To encourage tight coupling between classes
- d) To decrease the number of classes in the system

14. Given the following. What is true?

```
public class Room {  
    public int roomNr;  
    private Date beginDtm;  
    private Date endDtm;  
  
    public void book(int roomNr, Date beginDtm, Date endDtm) {  
        this.roomNr = roomNr;  
        this.beginDtm = beginDtm;  
        this.endDtm = endDtm;  
    }  
}
```

- a) the code demonstrates polymorphism
- b) the class is fully encapsulated
- c) the variable roomNr breaks encapsulation
- d) variables beginDtm and endDtm break polymorphism
- e) the method book breaks encapsulation

15. What can directly access and change the value of the variable roomNr?

```
package com.mycompany;  
  
public class Hotel {  
    protected int roomNr = 100;  
}
```

- a) only the hotel class

- b) any class
- c) any class in `com.mycompany` package
- d) any class that extends `Hotel`

16. What is the output of the following?

```
public class Example {  
    public static void main(String[] args) {  
        String str1 = "hello";  
        String str2 = new String("hello");  
        System.out.println(str1 == str2);  
    }  
}
```

- a) true
- b) false
- c) Compilation Error
- d) Runtime Error

17. Which of the following statements is incorrect in Java?

- a) public members of a class can be accessed by any other class
- b) private members of a class can be inherited by a sub class, and become protected members in sub class
- c) protected members of a class can be accessed by any class in the same package
- d) private members of a class can only be accessed by members of that same class it was declared

18. Which of the following allow us to define an IS-A relationship in Java? Circle **all** that apply.

- a) interfaces
- b) classes
- c) local variables
- d) dynamic binding
- e) none of the above

19. Which of these can be overloaded? (choose one or more answers)

- a) methods
- b) constructors
- c) classes
- d) interfaces

20. Given:

```
public interface Jumper { public void jump(); }  
public class Animal { }  
public class Dog extends Animal { protected Tail tail; }  
public class Beagle extends Dog implements Jumper {  
    public void jump() {};  
}  
public class Cat implements Jumper { public void jump(); }
```

Which of the following are true?

- a) Cat is-a Animal
  - b) Cat is-a Jumper
  - c) Dog is-a Animal
  - d) Dog is-a Jumper
  - e) Cat has-a Animal
  - f) Beagle has-a Tail
  - g) Beagle has-a Jumper
21. Which of the following statements is true about checked exceptions in Java?
- a) They must be caught or declared in the method signature using the "throws" clause
  - b) They are subclasses of RuntimeException
  - c) They occur at runtime
  - d) They do not need to be handled explicitly.
22. What happens if an exception is thrown within a "try" block but is not caught by any "catch" block?
- a) The program continues to execute normally
  - b) The program crashes with a runtime error
  - c) The "finally" block is executed
  - d) The exception is automatically caught by the JVM
23. A team manager needs data about the changes that have taken place for AWS resources in his account during the past two weeks. Which AWS service can help get this data?
- a) AWS Config
  - b) CloudWatch
  - c) Amazon Inspector
  - d) AWS Cloud Trail
24. A company wants to establish a private, dedicated connection between AWS and its on-premises data center. Which AWS service is the right choice for this requirement?
- a) Amazon CloudFront
  - b) Amazon API Gateway
  - c) AWS Direct Connect
  - d) AWS Site-to-Site VPN
25. Which of the following will help you control the incoming traffic to an Amazon EC2 instance?
- a) Network access control list (network ACL)
  - b) Security Group
  - c) Route Table



- d) AWS Resource Group
26. Which of the following statements is true about interfaces in Java?
- a) Interfaces can have constructors
  - b) A class can implement multiple interfaces
  - c) Interfaces can contain method implementations
  - d) Interfaces cannot contain constants
27. A company is looking at real-time processing of streaming big data for their ad-tech platform. Which of the following AWS services is the right choice for this requirement?
- a) Amazons Simple Queue Service (SQS)
  - b) Amazon Redshift
  - c) Amazon Kinesis Data Streams
  - d) Amazon EMR
28. An e-commerce application sends out messages to a downstream application whenever an order is created. The downstream application processes the messages and updates its own systems. Currently, the two applications directly communicate with each other. Which service will you use to decouple this architecture, without any communication loss between the two systems?
- a) Amazon Kinesis Data Streams
  - b) Amazon Simple Queue Service (SQS)
  - c) AWS Lambda
  - d) Amazon Simple Notification Service (SNS)
29. Which of the following AWS services will help provision a logically isolated network for your AWS resources?
- a) Amazon Virtual Private Cloud (Amazon VPC)
  - b) AWS PrivateLink
  - c) Amazon Route 53
  - d) AWS Firewall Manager
30. Which of the following are the security best practices suggested by AWS for Identity and Access Management (IAM)? (Select two)
- a) Enable AWS Multi-factor Authentication (AWS MFA) on your AWS root account. MFA helps give root access to multiple users without actually sharing the root login credentials
  - b) Do not change passwords and access keys once created. This results in a failure of connectivity in the application logic
  - c) Do not share security credentials between accounts, use IAM roles instead



- d) When you create IAM policies, grant the least privileges required to perform a task
- e) Share your AWS account root user credentials only if absolutely necessary for performing an important billing operation

**Section 3: Short essay questions. Please limit your responses to a few sentences.**

1. You are writing an external service that a client program attempts to connect to. What type of exception (check or unchecked) do you believe we should use when the client program provides an invalid web address? Explain why.  
(5 pts.)

**Answer:**

Using a checked exception for handling an invalid web address in a client program is appropriate. Checked exceptions ensure that the client program explicitly handles the exceptional condition, either through a try-catch block or by declaring the exception in its method signature. This promotes robustness and reliability by compelling developers to acknowledge and deal with potential errors. Additionally, checked exceptions facilitate clear communication between the caller and the callee, enhancing application maintainability.

2. Java programs are said to be write once, run anywhere. Please explain this statement.  
(5 pts.)

**Answer:**

"Write once, run anywhere" refers to Java's platform independence. Java source code is compiled into bytecode, which runs on any Java Virtual Machine (JVM), abstracting away hardware and operating system differences. This bytecode is portable, allowing Java applications to run on any device with a compatible JVM, regardless of the underlying platform. Java's platform independence enables developers to write code once and deploy it across various devices and operating systems without modification, facilitating cross-platform development and deployment.

3. Explain the difference between abstract classes and interfaces in object-oriented programming. Discuss when it is appropriate to use each construct and the implications for code design.  
(5 pts.)

**Answer:**

Abstract classes in Java can contain method implementations and instance variables, while interfaces only define method signatures. Abstract classes are used when providing a common base implementation for subclasses, promoting code reuse. Interfaces are used to enforce contracts, allowing for multiple inheritance-like behavior, making them suitable for defining API specifications. Thus, abstract classes are appropriate for sharing code and providing a common base implementation, while interfaces are appropriate for defining contracts and supporting multiple inheritance-like behavior.