



Name:

Vnumber:

SENG 275 MID TERM EXAM 3 (10%)-O3, O6

Instructor: Dr. Navneet Kaur Popli, Date: 26 July 2023, Time: 12:30-01:20 PM PST

Mode: Pen-Paper, Synchronous, Timed, closed book, to be done in the question paper

Total Marks: 35, Total Pages: 7, Total questions:5

Note: This is an individual activity. Copying or cheating of any kind is not allowed.

Q1) Consider a Travel website using a UserService. The UserService is responsible for adding, deleting, and modifying user records. It is also responsible for travel planning for the users. The UserService keeps the user data in a UserRepository. The UserRepository is implemented using MongoDB and is implemented in the following manner: (Total 12M) O3

```
public class UserService
```

```
{    private UserRepository ur;

    public UserService()
        { ur=new UserRepository();}
    ur.addUser();
    ur.userPlan();    }
```

A tester looks at the code and points out that this code has low testability.

a) Why does the tester think this code is less testable? Explain. (2M)

b) The tester also says that the testability can be increased using dependency injection and inversion of control. What are these two concepts and explain how they increase testability in an application. (2+2=4M)

c) Refactor the above code to implement the dependency injection concept. (2M)

d) Explain how this refactoring increased the testability of the code? (2M)

e) How does this refactoring improve the controllability of the test for the UserService? (2M)

Q2) Consider the java code for an e-commerce application with various user, product, order, and other functionalities. Identify any two technical debts in the code and explain ways to reduce those debts. (No solution code is to be implemented) (4M) O3

```
public class ECommerceApplication {  
    // User-related functionalities  
    public void createUser(String username, String password) {  
        // Code to create a new user in the database  
        // ... (database operations)  
    }  
  
    public User getUserById(int userId) {  
        // Code to fetch user from the database  
        // ... (database operations)  
        return user;  
    }  
  
    // Product-related functionalities  
    public void createProduct(String productName, double price) {  
        // Code to create a new product in the database  
        // ... (database operations)  
    }  
  
    public Product getProductById(int productId) {  
        // Code to fetch product from the database  
        // ... (database operations)  
        return product;  
    }  
}
```

```
// Order-related functionalities
public void createOrder(int userId, List<Integer> productIds) {
    // Code to create an order for the user in the database
    // ... (database operations)
}

public Order getOrderById(int orderId) {
    // Code to fetch order from the database
    // ... (database operations)
    return order;
}

// Other functionalities...
}
A2)
```

Q3) Consider the following java code which reads from a file. What is the reason for the resource optimism smell in the code. Suggest one way in which this smell can be removed. (2M) O3

```

import java.io.BufferedReader;
import java.io.FileReader;
import java.io.IOException;

public class ReadingFromFile {
public static void main(String[] args) {
    String filename = "example.txt";
    BufferedReader reader = new BufferedReader(new FileReader(filename));
    String line;
    while ((line = reader.readLine()) != null){
        System.out.println(line);}
    reader.close();
    }}
A3)

```

Q4) Consider the code to find the prime factors for an integer n.

```

1. class PrimeFactor{
2. public static void primeFactors(int n)
3. {
4. while(n%2==0){
5. System.out.println(2+ “ “);
6. n/=2;
7. }
8. for (int i=3, <=Math.sqrt(n); i+=2){
9. while(n%i==0){
10. System.out.println(i+” “);
11. n/=i;
12. }
13. }
14. If(n>2)
15. {System.out.println(n);}
16. }

```

a) Draw the control flow graph for the code. (5M)

b) Find cyclomatic complexity for the code. (1M)

c) What can you say about the testability of the code by the value of cyclomatic complexity?
Find out all independent paths from the graph. (6M)

Q5) Consider the following code to find factorial of a positive integer.

```
public class Factorial {  
    public int calculateFactorial(int n) {  
        if (n == 0) {  
            return 1;  
        } else {  
            return n * calculateFactorial(n - 1);  
        }  
    }  
}
```

Find any 5 possible mutations for the given code. (Complete mutated code is not required to be shown) (5M) O6

A5)

END