SDJ2 – exam – A21

Please be at the examination room at least 30 minutes early, in case of cancellations.

Each student has 20 minutes. This includes setting up, the actual examination, grading and feedback. You are allowed (expected) to have presentations.

Before you enter the exam room, please have everything ready: the computer charged and turned on, presentations/IntelliJ open or at least you have found the folder.

Below are 6 questions, each with 2 topics.
You will draw the question at the start of the exam.

You may present the topics in the order of your choice.

The exam will proceed in three steps:

1) Spend ~7 minutes on the first topic
2) Spend ~7 minutes on the second topic
3) Leave the room, get called back in, get your grade

Below you will find an overview of the topics, followed by a detailed overview of what should be covered for each topic.

Question 1: Sockets + Strategy
Question 2: MVVM + Observer
Question 3: RMI + Proxy
Question 4: State pattern + JUnit
Question 5: Readers/Writers + Singleton
Question 6: Producer/Consumer + Adapter

Sockets

Explain the main parts of a communication between computers using sockets.
Present your own example from an assignment, sep2, or exercise

- Use UML to present an overview for your solution
- Show the related code fractions for socket connections.
- Explain how to handle multiple clients
  - Why is multithreading needed on the server side?
  - Why is multithreading needed on the client side?

Strategy design pattern

Describe the pattern using UML diagrams.

- What is the purpose?
- What are the different parts involved? How do they interact?

Present your own example from an assignment, sep2, or exercise

- Use UML and describe how your solution follows the pattern
- Show Java code – how the pattern is implemented.
- Show Java code for how the pattern can be used.

Model-View-ViewModel (MVVM)

Describe the pattern using UML diagrams.

- What is the purpose?
- What are the different parts involved and their purpose?
- How do the different parts interact?

Present your own example from an assignment, sep2, or exercise

- Use relevant UML diagrams to present how your solution follows the pattern
- Show Java code
  - How the MVVM pattern is implemented.
  - Relevant for this pattern including the databinding between View and View-Model.

Observer design pattern

Describe the pattern using UML diagrams.

- What is the purpose?
- What are the different parts involved? How do they interact?

Present your own example from an assignment, sep2, or exercise

- Use UML and describe how your solution follows the pattern
- Show Java code – how the pattern is implemented.
- Show Java code for how the pattern can be used.

RMI

Explain the main parts of a communication between computers using RMI.
Include among other things in your description: Remote interface, Registry and Stub.
Present your own example from an assignment, sep2, or exercise

- Use UML to present an overview for your solution
- Show the related code fractions for RMI communication for your presented solution.
- Relate it to your general description from earlier.
- Describe e.g. by an example, how to handle a server call-back to a client.
- Describe e.g. by an example, how to handle a server broadcast to all clients.

Proxy design pattern

Describe the pattern using UML diagrams.

- What is the purpose?
- What are the different parts involved?

Present your own example from an assignment, sep2, or exercise

- Use UML and describe how your solution follows the pattern
- Show Java code – how the pattern is implemented.
- Show Java code for how the pattern can be used.

State design pattern

Describe the pattern using UML diagrams.

- What is the purpose?
- What are the different parts involved?

Present your own example from an assignment, sep2, or exercise

- Use UML and describe how your solution follows the pattern
- Show Java code – how the pattern is implemented.
- Show Java code for how the pattern can be used.

JUnit testing

Discuss the difference between black-box and white-box testing.
Present your own example from an assignment, sep2, or exercise

- Give examples of a class or methods you want to test and describe what and how to test.
- Relate this to equivalence partitioning, boundary value analysis, ZOMB+E and branch testing.
- Show the class or method you are testing
- Present your test strategy
- Show the JUnit test methods and relate it to your test strategy and describe how we can be sure that the tested method is correctly implemented

Readers-Writers problem

Explain the Readers-Writers problem (Touch upon monitors, wait/notify, critical sections, guarded blocks and the relation to threads as needed)
Present your own example from an assignment, sep2, or exercise

- Use UML to present an overview for your solution to a Readers-Writers problem
- Show the related code fractions for your presented solution.
- Relate it to your general description from earlier.

Singleton

Describe the singleton pattern using UML diagrams.

- What is the purpose?
- What are the different parts involved?

Present your own example from an assignment, sep2, or exercise

- Use UML and describe how your Singleton solution follows the pattern
- Show Java code – how the Singleton is implemented.
- Show Java code for how the Singleton can be used.

Producer-Consumer problem

Explain the Producer-Consumer problem, including the blocking queue (Touch upon monitors, wait/notify, critical sections as needed).
Present your own example from an assignment, sep2, or exercise

- Use UML to present an overview for your solution to a Producer-Consumer problem
- Show the related code fractions for your presented solution.

Adapter design pattern

Describe the pattern using UML diagrams.

- What is the purpose?
- What are the different parts involved?

Present your own example from an assignment, sep2, or exercise

- Use UML and describe how your solution follows the pattern
- Show Java code – how the pattern is implemented.
- Show Java code for how the pattern can be used.