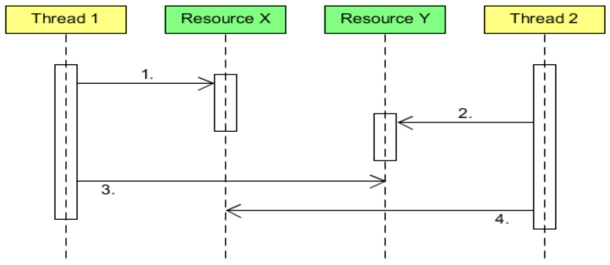
# Threads

1. Sequence diagrams illustrate interactions between objects on a time axis going from the top to the bottom of the diagram. The lifeline of an object is the dashed line. The messages passed (that is, the method calls) between the objects are shown as the arrows.

On the figure the arrows denote locking resources. We know that both threads need both resources.



At which method call can we be sure that there is a deadlock?

2. What will be the result of calling the wait() method of an object if the calling thread doesn't posess the monitor lock? throws an IllegalMonitorStateException

wait give up the lock and sleep until another thread enters the monitor and notify. the sleepy thread will release the lock and ask for it after next release

3. What is the default priority of threads? 5

4. Which possible outputs can this running code produce?

```
public class Tester extends Thread {
    int code = 9;
    public void run() {
        this.code = 7;
    }
    public static void main(String[] args) {
        Tester thread = new Tester();
        thread.start();
        for (int i = 0; i < 5; i++) {
            System.out.print(thread.code);
        }
    }
}
```

97777

5. What will be the result when running this code?

```
public class MyThread extends Thread {
    volatile private int x = 3;
    public static void main(String[] args) throws Exception {
        new MyThread().foo();
    }
    public MyThread() {
        x = 10;
        start();
    }
    public void foo() throws Exception {
        join();
        x = x - 1;
        System.out.println(x);
    }
    public void run() { x *= 2;  }
}       19
```

6. What will be the result when running this code?

```
public class Test extends Thread {
    public void run() {
        System.out.println(isAlive());
    }
    public static void main(String[] args) {
        Test test = new Test();
        System.out.println(test.isAlive());
        test.run();
    }
}  false , false
```

7. Which of these are atomic operations?

A. Reading/writing reference variables
B. Reading/writing of all primitive types
C. Both
D. None of the above

8. Which statement is true of this code?

```
public class Test extends Thread {
    private static int x;
    public synchronized void foo() {
        int current = x;
        current++;
        x = current;
    }
    public void run() {    foo();    }
}
```

A. We can make the class thread-safe, if we make the run method synchronized.
B. The class is thread-safe.
C. We can make the class thread-safe, if we make the foo method static.
D. This code throws an exception at runtime.

9. Which of these statements about immutable classes is not true?

A. The class should be final to avoid inheritance.
B. All the attributes of the class should be declared as private final.
C. We have to make a copy of the objects stored in attributes, when we return it in a getter method.
D. The class should not contain any kind of methods, that change the state after the constructor.

## JDBC

1. Which of these SQL statements are DML statements?

A. update [table] set ...
B. delete from [table] ...
C. alter table ...
D. insert into [table] ...

2. Which of these is an incorrect way to access the data in a ResultSet?

```
A. String value0 = rs.getString(0);
B. String value1 = rs.getString(1);
C. int    value2 = rs.getInt(2);
D. int    value3 = rs.getInt(“ADDR_LN1");
```

3. What information cannot be obtained from the SQLException object?

A. SQL status code.
B. Driver / Database specific error code.
C. Database request causing the error.
D. Description of the error that occurred.

5. Which of the following statements is true?

A. CallableStatement extends the PreparedStatement interface. This interface can be used to call SQL stored procedures.
B. Statement extends the PreparedStatement interface and is used when the SQL query does not need to be run multiple times.
C. PreparedStatement is used to start static queries (eg select * from table), therefore PreparedStatements cannot be parameterized.
D. Batch processing of SQL statements is possible using PreparedStatement.

6. How to start a new database transaction?

A. By requesting a Transaction object from Connection and calling that begin () method.
B. By requesting a Transaction object from Connection and setting its autoCommit property to false.
C. By calling the beginTransaction method of the Connection.
D. By setting the autoCommit property of the Connection to false and executing an SQL statement.

7. When do we use the transaction on databases?

A. To run stored procedures
B. To perform multiple operations atomically
C. For a linked table query
D. To name transfers

8. Which connection is usually interpreted with a connection table?

A. 1-n connection
B. m-1 connection
C. m-n connection
D. D connection

9. Which one is not part of the Entity-Relationship diagram?

A. Attribute
B. Entity
C. Keys
D. Class

10. Which one is not true about the relationship between JTable and the model?

A. JTable does not contain data
B. The model can be redefined
C. The model cannot notify JTable of the change
D. The representation of the model may differ from the data queried by JTable

## Software Technology / UML

1. What are the principal quality metrics of software?
   A. Delivery time, implementation cost, hardware and software requirements.
   B. Maintainability, reliability, safety, efficiency, usability.
   C. Modifiability, Extensibility, Resolution, Reusability, Reliability.
   D. Ergonomics, usability, compatibility, hardware and software requirements.

2. Which of these is the structure of a user story?

A. USER … IN USE CASE … WITH RELATION …
B. AS A … USE … TO …
C. WHEN … APPLYING … IN ORDER TO …
D. GIVEN … WHEN … THEN …

3. What is the correct order of requirements analysis steps?
   A. feasibility analysis, requirement exploration, requirement specification, requirement validation
   B. requirement exploration, requirement specification, requirement validation, feasibility analysis
   C. requirement exploration, requirement validation, requirement specification, feasibility analysis
   D. requirement exploration, requirement specification, feasibility analysis, requirement validation

4. Which of these is not a software development process?

|  |  |
|---|---|
| A. waterfall | B. evolution |
| C. scrum | D. prototyping |

5. Which of these are the relations of a UML use case diagram?
   A. dependency, composition, usage, nesting
   B. precedes, include, usage, generalization
   C. usage, nesting, import), dependency
   D. interface, implementation, include

6. What is the UML deployment diagram used for?
   A. Describe the sequence of steps required to install the software on a given machine.
   B. It depicts all the error possibilities that you may encounter during installation.
   C. Describe the software components according to how to install them in an installation package.
   D. Depict the physical placement of software components (on different machines) with the required software environment.

7. Which OOP design principle is violated by the singleton design pattern? The singleton pattern guarantees that there is only a single instance created from a class which can be accessed through a static method.
   A. Single Responsibility Principle
   B. Open/Closed Principle
   C. Liskov Substitution Principle
   D. Dependency Inversion Principle

8. Which of these techniques can be used to implement dependency inversion?
   A. observer
   B. MVC (modell-view-controller)
   C. dependency injection
   D. generalization

9. What is Test Driven Development (TDD)?
   A. A software development method in which tests are written before the actual program code is written.
   B. Test method to ensure that test cases cover all program units and are performed in the appropriate order.
   C. A general principle that states that all instructions in program code should be verified using unit tests (100% code coverage).
   D. A testing method in which unit tests are first performed on classes (and their methods), then integration tests are used to check the combined behavior of the classes, and finally system tests are used to check the behavior of the entire software.

10. Which the following features is not provided by unit testing frameworks?
   A. Create manual test cases in separate program units (classes).
   B. Generate test cases automatically, that covers all scenarios, by analyzing the code.
   C. Assert statements, which compare the expected and the return values.
   D. Creation of test reports, which lists the passed/failed tests.

12. In the case of Unit testing, the parts of the program must be separated from each other and boundaries must be established between them. One possible solution to this is to use objects that mimic the behavior of other objects. What are these objects called?
   A. Mock objects
   B. Modules
   C. Units
   D. Atoms

13. What is the starting point of object-oriented design?

|  |  |  |  |
|---|---|---|---|
| A. | Functions | B. | Activities |
| C. | Entities and relations | D. | Architecture |

14. How do we express that an object is related to several objects of a class?
   A. With composition
   B. With multiplicity
   C. With an array type attribute
   D. With aggregation

15. Which of these can we assign constraints to on a class diagram?

|  |  |  |  |
|---|---|---|---|
| A. | Relation | B. | Attribute |
| C. | Method parameters | D. | All |

16. Which of these relations is between classes?

A. Association B. Dependency
[✓ C]. Inheritance D. Aggregation

17. Which diagram is not part of the dynamic model?
A. State diagram
B. Sequence diagram
C. Activity diagram
[✓ D]. Component diagram

18. What cannot have a state on a state diagram?
A. Name
B. Invariant
C. Precondition
D. Parameter

19. Which method can reduce the complexity of the state diagram?
A. Generalization
B. Aggregation
[✓ C]. By generalization and aggregation
D. None of the others

20. Which statement is true?
[✓ A]. The invariant of generalization is the disjunction of the invariants of states
B. The invariant of generalization is the conjunction of the invariants of states
C. The invariant of aggregation is the disjunction of the invariants of states
D. The states obtained by the two methods have no invariant

21. Which diagrams are part of the static model?
[✓ A]. Class Diagram, Object Diagram
B. Class Diagram, State Diagram
C. Object diagram, Sequence diagram
D. Object Diagram, Activity Diagram

22. Based on the following implementation, which relationship exists exactly between the car and the engine?

```
public class Car {
    private Engine engine;

    public Car(Engine engine){
        this.engine = engine;
    }

    public Engine getEngine(){
        return engine;
    }
}
```

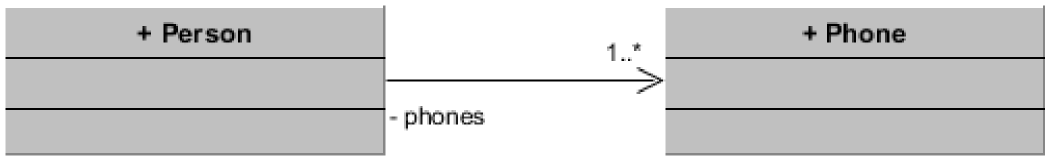
A. Association
[✓ B.] Aggregation
C. Composition
D. Generalization

24. Based on the following implementation, which relationship exists exactly between the car and the engine?

```
public class Car {
    private final Engine engine;

    public Car(int engineCapacity, int engineSerialNumber) {
        this.engine = new Engine(engineCapacity, engineSerialNumber);
    }

    public int getEngineCapacity() {
        return engine.getEngineCapacity();
    }

    public int getEngineSerialNumber(){
        return engine.getEngineSerialNumber();
    }
}
```

**A.** Association
**B.** Aggregation
[✓ **C.**] Composition
**D.** Generalization

25. What is the relationship between the Person and Phone classes in the class diagram below?



A. Association
B. Aggregation
C. Composition
D. Generalization

## Implementation related questions

1. What kind of methods do we have to implement when we would like to use instances of our own class as keys in a HashMap?
A. == operator
B. hashCode(…) method
C. equals(…) method
D. hashCode(…) and equals(…) methods

2. Which of the following options should you choose if you want to apply mainly an index-based search to a dynamically changing data set where the same element can occur more than once? (We only want to add a new item to the end of the collection, we don't want to delete it from the collection often.)
E. ArrayList
F. LinkedList
G. Array
H. HashSet

3. Which implementation to choose from the following options if you want to use a collection that does not contain duplicate items and you do not need to store the items in the order of insertion or in ascending order of values?
A. List
B. TreeSet
C. LinkedHashSet
D. HashSetArrayList

4. Which of the following implementations of interfaces are used to store key-value pairs?
A List
B. Set
C. Map
D. Collection

final and abstract don't mix

final class forbid the derivation of it

5. Which of these statements is true?

A. A final abstract class must have at least one abstract method

B. An abstract class has at least one abstract method

C. All the attributes of a final class are final

D. The derived class of an abstract class can be also abstract

6. Which of these statements is false?

A. We cannot derive from a final class

B. Interfaces cannot be derived

C. A class can implement multiple interfaces

D. We have to implement all methods of the interface

7. Which of these can be a generic parameter?

A. Primitive type

B. Class

C. Interface

D. Class, that implements the methods used in the generic

8. Which collection can be indexed?

A. HashSet

B. HashMap

C. Vector

D. TreeMap

9. What can be static in JAVA?

A. Class field

B. Method

C. Class / interface   nested classes -> class field

D. Enumeration   implicit static

10. What does Java support about multiple specialization and multiple generalizations?

A. Generalization

B. Specialization

C. Both

D. None

## Software development methodologies and models

1. Which of the following is not an agile principle?

A. Prefer the methodology instead of the tools
B. Prefer the operating software instead of a comprehensive documentation
C. Prefer cooperation with the client instead of enforcing the contractual negotiations
D. Prefer responding to changes instead of following the plan.

2. Which statement is not true about the sprint?

A. The product is all designed, coded and tested within the sprint.
B. The result of the sprint is a working code representing business value.
C. Once tasks and times have been defined, only the product owner will be involved in the work of the team.
D. The Scrum team works in a self-organizing way throughout the sprint.

3. Which statement is true for the Scrum Master?

A. Scrum Master is the manager of the Scrum team
B. The Scrum Master leads the daily Scrum
C. Scrum Master is not responsible for protecting the work of the SCRUM team from outside influences
D. Scrum Master is responsible for the processes

4. Which statement is true for daily Scrum?

A. The daily Scrum is led by the Scrum Master.
B. During the daily Scrum, team members report to the Scrum Master on their progress.
C. During the daily Scrum, the goal is to remove obstacles that affect the team.
D. The daily Scrum can last up to 15 minutes.

5. Test-driven development (TDD) is a software development methodology according to which…

A. the tests must be written before the actual program code is implemented.
B. tests must be performed on each unit after the actual program code has been implemented.
C. new program code can be implemented after approval by the test colleague.
D. the test report must be assigned to the documentation one day before the implementation of the new program code.

## Version control

1. What is the purpose of the continuous integration (CI) practical method?

A. Immediate, automated filtering of possible errors and integration problems, feedback to the developer. (Self-check build)
B. Automatically re-run failed integration tests until they are repaired.
C. Facilitates the transition to an object-oriented programming language.
D. Complete replacement of manual testing.

2. What are the disadvantages of centralized version control systems (Example: SVN, Perforce, CVS)?

A. The privileged role of the server. (In the event of a failure, the system becomes unusable until the server is repaired.) In addition, version control requires a network connection.
B. File-based operation
C. Local storage
D. Concurrency management realized by exclusive locks.

3. Which statement is not true for distributed version control systems (Example: Git, Mercurial)?

A. A decentralized, distributed network model is used, where concurrency management is typically done through post-submission merging.
B. Each client has a complete repository and version history. The operations of the revision management tool take place locally on the client's storage.
C. Communication is peer-to-peer, but it is also possible to set up dedicated servers.
D. A set of file-based operations is typical, where concurrency management is typically done by merging before submission.

4. Is it true that there can be no conflict with Git merge?

A. True, as conflict can only occur with rebase.
B. False, because there is a conflict between colleagues for every merge.
C. True, since each merge is also another commit.
D. False, as git may not be able to resolve changes automatically. (Example: Two different commit stores a change on the same line in a file.)

5. Which of the following are the build tools?
A. Ant, Maven, Gradle
B. Ant, Git, Subversion (SVN)
C. Ant, Maven, Subversion (SVN)
D. Maven, Gradle, Git

## Design principles, Design patterns

1. What does the DRY principle say?

A. Don’t implement code in advance that “will need it in the future” because we will almost certainly never need it.
B. Each piece of knowledge must have a single, clear and reliable representation within a system.
C. Perfection is not best approached when we can no longer add anything to a system, but when we do not know what to take from it.
D. It is safe to say that the quality of our code base will improve if we always leave our current code there a little “better”, a little cleaner than we found it.

2. Which object-oriented principle is violated in the code snippet below?

```
public enum Status {
    UNRESOLVED, IN_PROGRESS, RESOLVED, CLOSED
}

public class Task {
    protected Status status;

    public void close() {
        this.status = Status.CLOSED;
    }
}

public class ProjectTask extends Task {

    @Override
    public void close() {
        if(status == Status.IN_PROGRESS) {
            throw new RuntimeException("You cannot close …");
        }
        super.close();
    }
}
```

A. Liskov Substitution Principle
B. Dependency Inversion Principle
C. KISS
D. DRY

3. Which of the following is not a SOLID principle?

A. Liskov Substitution Principle
B. Open / Closed Principle
C. Single Responsibility Principle
D. Separation of Concerns Principle

4. Which object-oriented principle is violated in the code snippet below?

```
public class DamageCalculator {
    private int damageFactor;

    public DamageCalculator(int damageFactor) {
        this.damageFactor = damageFactor;
    }
    public int totalDamage(int baseDamage, int damageAlgorithm){
        switch(damageAlgoritm) {
            case 0: return baseDamage * damageFactor;
            case 1: return baseDamage * 4;
            default: return baseDamage;
        }
    }
}
```

A. Dependency Inversion Principle
[✓ B]. Open/Closed Principle
C. Interface segregation Principle
D. Liskov Substitution Principle

5. Given a Lamp class. The lamp has a color and can be switched on / off. In our apartment there is a switch on the wall, which was implemented as follows. What could be the problem with this implementation?

```
public class Switch {
    private Lamp lamp;

    public Switch(Lamp lamp) {
        this.lamp = lamp;
    }

    public void useSwitch(){
        if(lamp.isOn()) {
            lamp.turnOff();
        } else {
            lamp.turnOn();
        }
    }
}
```

A. The switch violates the Liskov Substitution Principle
B. The switch violates the Open/Closed Principle
C. The switch is at a higher abstraction level than the lamp, so it violates the Dependency Inversion Principle
[✓ D]. The switch violates the Single Responsibility Principle

6. Which design pattern provides a solution to the problem of notifying multiple objects when another object changes state.

A. Singleton
B. Observer
C. Adapter
D. Factory

7. Which design pattern can be used to avoid constructors with long parameter lists?

A. Observer
B. Factory
C. Builder
D. Command

8. Which design pattern implementation can be recognized in the following code snippet?

```
public MyPattern withName(String name) {
    this.name = name;
    return this;
}

public MyPattern withNumber(int number) {
    this.number = number;
    return this;
}
```

A. Singleton
B. Builder
C. Command
D. Adapter

9. Which design pattern can we use if we want to provide an interface for creating a family of related or interdependent objects without specifying a specific class?

A. Factory method
B. Adapter
C. Builder
D. Abstract Factory

10. Which design pattern can be used in case we want to transfer the instantiation of a given class to the corresponding subclasses?

A. Factory method
B. Builder
C. Command
D. Observer

11. Which design pattern does the following code snippet implement?

```java
public class MyPattern {
    private MyPattern(){}

    private static class MyPatternInstance {
        private static final MyPattern INSTANCE = new MyPattern();
    }
    public static MyPattern getInstance() {
        return MyPatternInstance.INSTANCE;
    }
}
```

A. Singleton
B. Factory
C. Builder
D. Adapter

12. Which design pattern does the following code snippet implement?

```java
public abstract class Maze {
    private final List<Room> rooms = new ArrayList<>();

    public Maze() {
        Room room1 = makeRoom();
        Room room2 = makeRoom();
        room1.connect(room2);
        rooms.add(room1);
        rooms.add(room2);
    }
    protected abstract Room makeRoom();
}

public class MagicMaze extends Maze {

    @Override
    protected Room makeRoom() {
        return new MagicRoom();
    }
}
```

A. Factory method
B. Command
C. Adapter
D. Abstract Factory