

### Wichtige Hinweise für die Klausur:

- Identifizieren Sie pro Aufgabe je die beiden relevanten Design Patterns.
  - **Zeichnen Sie das Klassendiagramm** ggf. mit einem dokumentenechten Stift nach.
  - **Definieren Sie notwendige Attribute und geeignete Datentypen.**
  - Getter- und Setter-Methoden sind nicht zu modellieren.
  - Nicht zu wertende Zeichnungen und/oder Skizzen sind eindeutig zu kennzeichnen.
- 
- Zeitansatz pro Aufgabe:
    - **Modellierung: maximal 12 Minuten**
    - Implementierung: maximal 25 Minuten
  - **Die Implementierung dient dem zusätzlichen Training und ist nicht klausurrelevant.**
  - **K** = Aufgabe hat hohe Relevanz für Klausur.
  - **Empfehlung:** Bitte bearbeiten Sie alle Aufgaben für ein optimales Klausurtraining.

### 01 // Wetterstation

Zwei Wissenschaftler beobachten kontinuierlich eine Wetterstation. Diese Wetterstation sammelt Daten zur Temperatur. Für einen Export der Daten meldet sich der Wissenschaftler über einen Proxy an, da ein direkter Zugriff auf die Daten der Wetterstation nicht möglich ist.

### K02 // Computer und USB-Schnittstelle

Ein Computer hat eine USB2-Schnittstelle. Ein DVD-Brenner hat eine USB3-Schnittstelle. Der DVD-Brenner wird über einen Adapter an den Computer angeschlossen. Ein Systemprozess im Computer beobachtet kontinuierlich die USB2-Schnittstelle und generiert eine Meldung, sobald ein Gerät (in diesem Fall der Adapter) an die USB2-Schnittstelle angeschlossen wurde.

### 03 // Auto

Ein Auto ist mit einem Motor ausgestattet. Bei einem Motor kann je nach Kundenwunsch zwischen 75 PS und 90 PS variiert werden. Das Drehen des Zündschlüssels erzeugt ein Kommando und der Motor wird an- oder ausgeschaltet. Das Drehen des Zündschlüssels nach vorne schaltet den Motor an. Das Drehen des Zündschlüssels nach hinten schaltet den Motor aus.

#### **04 // Taschenrechner**

---

Ein Taschenrechner hat den Status an oder aus. Durch Betätigen einer Taste wird der Taschenrechner – in Abhängigkeit von dem aktuellen Status – an- oder ausgeschaltet. Der Taschenrechner bietet die Rechenstrategien Addition und Subtraktion an.

#### **K05 // Passkontrolle**

---

An einem Flughafen existieren für die Passkontrolle drei Schalter (i) EU-Bürger, (ii) Nicht-EU Bürger und (iii) Diplomaten. Basierend auf dem Pass ist der korrespondierende Schalter zuständig. Die Passkontrolle an den drei Schaltern wird kontinuierlich von der Bundespolizei überwacht. Wird ein ungültiger Pass bei der Kontrolle erkannt, erhält die Bundespolizei automatisch eine Nachricht.

#### **06 // Datenspeicher**

---

Ein Sortierprozess beobachtet kontinuierlich einen Datenspeicher. In dem Datenspeicher befinden sich ganze Zahlen in aufsteigender Sortierung. Bei Änderung eines Wertes im Datenspeicher werden die Werte automatisch durch den Sortierprozess wieder sortiert. Dem Sortierprozess stehen zwei Sortierstrategien (i) QuickSort und (ii) MergeSort zur Verfügung.

#### **07 // Tower**

---

Bei einem Tower am Flughafen sind drei Flugzeuge F1, F2 und F3 sowie die beiden Startbahnen R1 und R2 zentral registriert. Der Tower sendet das Kommando Taxi mit der Angabe einer Startbahn an das (registrierte) Flugzeug.

#### **08 // Datenbank**

---

Eine vereinfachte Datenbank wird über eine Konsole durch die Kommandos einfügen und rueckgaengig gesteuert. Mit rueckgaengig wird die letzte Änderung rückgängig gemacht.

#### **09 // Flugzeug**

---

Ein Flugzeug hat zwei Triebwerke, die jeweils mit einem Temperatursensor ausgestattet sind. Standardmäßig wird die Temperatur in Fahrenheit angegeben. Für die Temperaturangabe in Celsius existiert ein Adapter. Vom Cockpit werden Daten von den beiden Temperatursensoren kontinuierlich überwacht.

#### **10 // Mikrowelle**

---

Eine Mikrowelle mit einer Mikrowelleneinheit hat einen Schalter. Standardmäßig ist die Mikrowelle ausgeschaltet. Bei Drücken des Schalters wird die Mikrowelle angeschaltet. Bei erneutem Drücken des Schalters wird die Mikrowelle ausgeschaltet. Die Mikrowelleneinheit existiert in zwei Varianten M1 und M2.

## **K11 // Flugzeugtür**

---

Eine Flugzeugtür wird über einen Schalter bedient. Initial befindet sich der Schalter im Status S0 (offen) und die Flugzeugtür ist geöffnet. Wird der Schalter im Status S0 gedrückt, wechselt der Status von S0 nach S1 (geschlossen). Wird der Schalter im Status S1 gedrückt, wechselt der Status von S1 nach S2 (gesperrt). Wird der Schalter im Status S2 gedrückt, wechselt der Status von S2 nach S3 (entsperrt). Wird der Schalter im Status S3 gedrückt, wechselt der Status von S3 nach S0 (offen). Eine Anzeige beobachtet kontinuierlich den Status der Flugzeugtür und wird bei Änderungen des Status automatisch informiert.

## **12 // Scheinwerfer**

---

Ein Auto hat zwei Scheinwerfer, die über einen Schalter gesteuert werden. Der Schalter sendet direkt ein Kommando an die beiden Scheinwerfer. Initial befindet sich der Schalter im Status S0 (Licht aus). Wird der Schalter im Status S0 gedrückt, wechselt der Status von S0 nach S1 (Parken). Wird der Schalter im Status S1 gedrückt, wechselt der Status von S1 nach S2 (Abblendlicht). Wird der Schalter im Status S2 gedrückt, wechselt der Status von S2 nach S3 (Fernlicht). Wird der Schalter im Status S3 gedrückt, wechselt der Status von S3 nach S0 (Licht aus).

## **13 // Abspielgerät**

---

Ein Abspielgerät hat eine Taste. Initial befindet sich die Taste im Status S0 (aus). Wird die Taste im Status S0 gedrückt, wechselt der Status S0 nach S1 (an). Wird die Taste im Status S1 gedrückt, wechselt der Status von S1 nach S0 (aus). Standardmäßig spielt das Gerät MP4 ab. Für das Abspielen von MP3 wird vom Abspielgerät ein Adapter genutzt.

## **14 // Lawinensensor**

---

Ein Sensor beobachtet kontinuierlich die Oberfläche eines Schneebretts. Zulässige Messwerte des Sensors sind 0 und 1. Der Sensor sendet Kommandos an eine Ampel. Die Ampel hat ein Licht in der Farbe rot. Bei einem Messwert von 0 sendet der Sensor das Kommando gruen an die Ampel und das Licht der Farbe rot wird ausgeschaltet. Bei einem Messwert von 1 sendet der Sensor das Kommando rot an die Ampel und das Licht der Farbe rot wird angeschaltet.

## **15 // Turbine**

---

Eine Turbine hat 96 Schaufeln. Es werden zwei Typen von Schaufeln (i) 48x Titan und (ii) 48x Karbon. Die erste Schaufel ist vom Typ Titan und hat den Index 1. Die Typen der Schaufeln wechseln sich einander ab. Die zweite Schaufel ist folglich vom Typ Karbon und hat den Index 2. Für jeden Typ existiert ein eigenes Prüfverfahren. Ein Roboter prüft sukzessive – beginnend mit der Schaufel Titan und Index 1 – jede Schaufel mit dem geeigneten Prüfverfahren. Über die Kommandos start, stop, naechste wird der Roboter von einem Mechaniker über eine Fernbedienung gesteuert.

## 16 // Virens Scanner

---

Eine Verzeichnisstruktur besteht aus Hauptverzeichnis, Unterverzeichnis und Datei. Eine Datei ist vom Typ Verzeichnis. Es existieren die Dateitypen exe, dll und doc. Der Virens Scanner ermöglicht das Scannen aller Dateien oder eines bestimmten Dateityps.

## 17 // Ladestation

---

Eine Ladestation hat drei Fächer. Jedes Fach hat eine Kapazität für eine Batterie. Die Batterien werden von einem Systemprozess verwaltet und sukzessive geladen. An der Ladestation befindet sich eine Anzeige. Die Anzeige beobachtet kontinuierlich den Systemprozess und den Fortschritt. Initial hat die Anzeige die Farbe rot. Sind alle Batterien geladen, wechselt die Farbe nach grün.

## 18 // Containerschiff

---

Ein Containerschiff hat einen Laderaum. Dieser Laderaum besteht aus 6 Hauptsegmenten mit je zwei Segmenten. Ein Segment hat eine Kapazität für zwei Container. Die Hauptsegmente werden sukzessive mit einem Kran beladen. Der Kran wird über eine Fernbedienung und den Kommandos verriegeln, anheben, absenken und entriegeln gesteuert.

### Wichtige Hinweise für die Bearbeitung:

- **Pro Student** werden **zwei Aufgaben** bearbeitet.
- Die **Zuordnung** der **Aufgaben** zu einem Studierenden erfolgt **mit** einem **Zufallsgenerator**.
- **Bearbeitung** der Aufgaben **lokal** auf den Rechnern und Nutzung der Templates.
- **Studium** der **Struktur** und **Funktionsweise** der beteiligten **Design Patterns**.
- **Modellierung** eines Klassendiagramms in Enterprise Architect.  
Bitte
  - nutzen Sie das **Theme „DHBW“** (Visual Style → Visual Appearance → Diagram).
  - benennen Sie das Klassendiagramm mit task<id>, z.B. task50.
  - benennen Sie die Datei mit <matrikelnummer>\_<task\_id>.eap, z.B. 3378535\_50.eap.
  - **exportieren** Sie das Klassendiagramm als **PDF**-Datei mit einer A4-Seite im Querformat.
- **Implementierung** einer einwandfrei lauffähigen Applikation in Java 8.  
Bitte
  - erstellen Sie ein **Paket** mit der Bezeichnung task<id>, z.B. task50.
  - nutzen Sie die **camelCase-Notation**, um die Lesbarkeit zu vereinfachen.
- **Erstellung einer vollständigen und verschlüsselten 7-Zip-Datei unter Beachtung des Prozedere für die Abgabe von Prüfungsleistungen und der Namenskonvention.**
- **Zeitansatz:** 10 Stunden
- **Abgabetermin:** Sonntag, 25.02.2018
- **Bewertung:** Testat