

Bitte tragen Sie individuell bis 08:00 Uhr am Vortag des Seminars über den Checklistenbaustein in Opal ein, welche Aufgaben Sie präsentieren können. Für die Präsentation sollten Sie ein Dokument im Querformat vorbereitet haben, das Sie dann entsprechend im Seminar in BigBlueButton hochladen können. Auch müssen Sie den virtuellen Klassenraum mit Mikrofon betreten.

Aufgabe 1: Entwurfsmuster: Observer

In der GUI eines Zeichenprogramms seien mehrere gezeichnete Linien markiert. Wenn in einer Eingabemaske die Farbeinstellung verändert wird, iteriert das Programm über alle Linien des Canvas, prüft ob die Linie markiert ist und ändert ggf. die Farbe der Linie mit einer Methode `setColour()`.

- Zeichnen Sie die vorgegebene Situation in einem Sequenz- und einem Klassendiagramm.
- Es soll eine alternative Lösung mit dem Observer-Pattern angedacht werden, bei der die markierten Linien die Eingabemasken beobachten sollen. Wie sehen jetzt Klassen- und Sequenzdiagramm aus?

Aufgabe 2: Entwurfsmuster: Decorator

Ein Kreis (Mittelpunkt (x/y) und Radius r) ist standardgemäß durchsichtig und hat einen dünnen schwarzen Rand. Er kann wie folgt modifiziert werden:

- mit einer Füllfarbe f und dem Grad der Transparenz t und/oder
- einem Rand mit der Linienbreite b und der Farbe f .

- Setzen Sie dies mit dem „Decorator pattern“ als Klassendiagramm um.
- Zeigen Sie ein Objekt als Objektdiagramm, das mit einer Füllfarbe und zwei Rändern versehen ist.

Aufgabe 3: Entwurfsmuster: Adapter

In einer Software werden zweidimensionale Positionen als Polarkoordinaten (Länge r und Winkel ϕ) gespeichert. Eine existierende Klasse `Position` mit der Methode `setPosition(double r, double phi)` soll jetzt durch eine Klasse `NeuePosition` mit der Methode `setPosition(double x, double y)` ersetzt werden. Letztere Methode erwartet allerdings Eingaben im kartesischen Koordinatensystem.

- Setzen Sie dies mit dem „Adapter“-Pattern als Klassendiagramm um.
- Geben Sie den Quelltext der Adapterklasse konkret an. Hinweis: es gilt $x = r \cdot \cos \phi$ und $y = r \cdot \sin \phi$.

Aufgabe 4: Function-Points und COCOMO

Eine Software mit 20 komplexen Eingaben (EI), 5 mittleren und 25 komplexen Ausgaben (EO) und 30 mittleren Anfragen (EQ), 2 einfachen und 10 komplexen internen Dateien (ILF) sowie einer komplexen externen Datei (ELF) soll als semi-detached Projekt in der Sprachen C++ mit $MOD = 1$ realisiert werden. Berechnen Sie die Personenmonate, Projektlaufzeit und Teamgröße.

Aufgabe 5: COCOMO

Sie sind Projektmanager für die Entwicklung einer embedded Software mit 200000 Zeilen Quelltext. Sie haben die Wahl zwischen sehr guten Programmierern ohne Kenntnis der benutzten Programmiersprache und niedrig qualifizierten Programmieren mit guter Erfahrung in der Programmiersprache. Welche Entscheidung treffen Sie? Welche Teamgröße benötigen Sie und wie lange wird das Projekt laufen (wenn keine weiteren Besonderheiten auf das Projekt zutreffen)?