

Prüfung: 1853 und 1798
Prüfer: Prof. Dr. Steimann
Datum: 06.03.2006
Note: 1,0

Die Prüfung lief in einer sehr angenehmen Atmosphäre ab. Ich habe auf die Fragen immer etwas ausführlich geantwortet. Ich kann Prüfer und Prüfung nur empfehlen.

Frage (F): Mit welchem Thema wollen Sie anfangen?
Antwort (A): Interfaces.

F: Dann fangen Sie mal an.

A: Trennung von Schnittstelle und Implementierung. Verschiedene Arten des Gebrauchs (Nutzniesser, Umfang ...).

F: Nennen Sie ein paar Interfaces in Java als Beispiele hierzu.

A: runnable, printable, compareable, serializeable.

F: Kann man serializeable auch anders ausdrücken?

A: Annotationen (erfragen über Introspektion)

F: Dann erzählen Sie mal etwas über Metaprogrammierung.

A: Vor- und Nachteile. Bsp. Mutierende Programme.

F: Besonderheiten der Introspektion.

A: Introspektion, keine Veränderung des Ablaufs möglich.

F: Was ist bzgl. Programme und Daten

A: Programme haben die Form von Daten.

F: Was versteht man unter Refactoring?

A: Ändern ohne Ablauf zu verändern. Wartbarkeit, Lesbarkeit usw. verbessern.

F: Wo werden diese gebraucht?

A: XP. XP ist ... daher Refactoring.

F: Was ist ein Pattern?

A: Nutzen (Wortschatz, Programmiererfahrung festhalten, Blaupausen). Ich habe das Composite Pattern als Beispiel erwähnt.

F: Wodurch zeichnet sich ein Anwendungsframework in der objektorientierten Programmierung aus?

A: Klassen die Kernfunktion zur Verfügung stellen ... Bsp. AWT.

F: Wo ist der Unterschied zur Software Architektur?

A: Framework beschreibt einen Aspekt/Subsystem. Software Architektur das Design grob (Abstraktion) im Ganzen. Ziele und Nutzen der Softwarearchitektur. Habe noch Tools erwähnt.

F: Haben Sie schon mit solchen Tools gearbeitet?

A: Nein, die gibt es leider noch nicht.

Ende.

Zu JUnit und Design by Contract wurde nichts gefragt. AspectJ, COM und EJB wurden auch nicht erwähnt.

Zu Kurs 1853:

Thema Interfaces

- ... welche Sorten gibt es
- ... Unterschiede, was macht man mit welchen, anhand von Beispielen
- ... Syntaktische vs. semantische Kompatibilität

Design-by-Contract als Ansatz für Beschreibung der Schnittstellensemantik

- ... was bedeutet Design-by-Contract
- ... Auffassungen über Invarianten
- ... Problematik der Änderung des Programmablaufs

Testen

- ... Welche Tests gibt es
- ... Tests weisen Fehler nach, nicht Fehlerfreiheit
- ... Wie macht man Unit-Tests mit JUnit
- ... Qualitätssicherung der Tests (Test der Tests)
- ... Test-first Ansatz

XP

- ... Welche Ziele
- ... Welche Restriktionen
- ... Refactoring
- ... Refactoring: Was ist Refactoring, welche Rolle im XP

Zu Kurs 1798:

Was ist SW-Architektur

- ... Top-Level Strukturen eines SW-Systems usw.
- ... Eine Teildisziplin des SW-Engineering
- ... Phase in vielen Prozessmodellen (auch als Grobdesign bezeichnet)

Funktionale vs. nichtfunktionale Anforderungen

- ... welche bestimmen SW-Architektur mehr

Wiederverwendung auf Architekturebene

- ... Architektur-Pattern
- ... Architekturframeworks
- ... Whitebox vs. Blackbox-Reuse

Prof. Dr. Steimann hat mit eher allgemeinen Fragen begonnen und ließ Raum für ausführliche Antworten.