



Sonder-Übungsblatt



Aufgabe S1 *Probleme und Herausforderungen der Softwaretechnik* (6 Bonuspunkte)

In der Vorlesung wurde die Softwarekrise in den 60er/70er Jahren besprochen. Die Kosten der Softwareentwicklung und -wartung stiegen im Verhältnis zu den Kosten der Hardware überproportional.

Der relative Anstieg der Softwarekosten wurde seitdem durch gegenläufige Entwicklungen abgeschwächt. Dennoch ist der Kostenanteil für Softwareentwicklung und Softwarewartung nicht wesentlich zurückgegangen, da sich neue Herausforderungen ergeben haben.

- Welche gegenläufigen Entwicklungen, die zur relativen Reduzierung der Software- und insbesondere der Softwarewartungskosten geführt haben könnten, hat es bis heute gegeben?
- Welche neuen Herausforderungen haben sich ergeben?

Nennen Sie jeweils mindestens drei Entwicklungen bzw. neue Herausforderungen und erläutern Sie deren Auswirkungen auf die Kosten der Softwareentwicklung und Softwarewartung.

Aufgabe S2 *Phasenmodell*

(6 Bonuspunkte)

Welche der folgenden Aspekte werden durch ein Phasenmodell festgelegt und welche nicht?

- (a) Beginn des Projekts
- (b) Reihenfolge des Arbeitsablaufs
- (c) Arbeitszeiten

- (d) Jeweils durchzuführende Aktivitäten
- (e) Definition der Teilprodukte
- (f) Fertigstellungszeitpunkt
- (g) Fertigstellungskriterien
- (h) Mitarbeiterqualifikationen
- (i) Mitarbeiterauswahl
- (j) Verantwortlichkeiten
- (k) Verwaltung
- (l) Methoden, Werkzeuge
- (m) Richtlinien, Standards
- (n) Programmiersprache

Aufgabe S3 *Programmer's Apprentice*

(8 Bonuspunkte)

In der Vorlesung wurde der alternative Ansatz zur Softwareerstellung „Programmer's Apprentice“ eingeführt (2-31).

Diskutieren Sie anhand folgender Fragen die Probleme, die bei der Realisierung eines Softwaresystems nach diesem Paradigma entstehen.

1. Welche Teile der Anforderungsdefinition lassen sich (z. Zt.) nur schwer formal erfassen? Betrachten Sie diese Frage auch unter Berücksichtigung der in der Vorlesung vorgestellten verschiedenen Klassen von Software.
2. Was kann man über den Detaillierungsgrad einer Anforderungsdefinition im Sinne dieses Ansatzes aussagen?
3. Wie lassen sich die in der Vorlesung vorgestellten Modellierungsprinzipien hier anwenden?
4. Für welche Klassen von Softwaresystemen oder Teilsystemen eines Gesamtsystems ist eine Vorgehensweise gemäß diesem Ansatz denkbar? Wie kann man im letzteren Falle die Vorteile des kontinuierlichen Ansatzes für Teile des Systems mit den Vorteilen diskreter Planungs- und Entwicklungsschritte für das Gesamtsystem in Einklang bringen?

Aufgabe S4 *Spiralmodell*

(8 Bonuspunkte)

In der Vorlesung wurde als alternativer Softwareentwicklungsansatz das Spiralmodell nach Boehm vorgestellt (2-32). Spielen Sie die Planungsphase eines Projekts für die Entwicklung eines *WebShop-Systems* anhand des Spiralmodells durch.

Durchlaufen Sie dabei *dreimal* den Zyklus, so dass am Ende der drei Zyklen die Entscheidungen für eine Neuentwicklung des Systems in Eigenregie unter Verwendung von PHP getroffen wurden, und für die eigentliche Entwicklung die Strategien zur Risikobeseitigung und das zu verwendende Prozessmodell festgelegt wurden. Verwenden Sie dazu das folgende Schema:

Zyklus x von 3	
Ziel	
Alternativen	(a) (b)
Randbedingung	
Evaluierung der Alternativen (Risiken)	(a) (b)
Strategien zur Risikobeseitigung	(a) (b)
Auswahl Prozessmodell	
Anwendung des Modells	
Ergebnisse	(a) (b)
Planung nächster Zyklus	

Aufgabe S5 *Bootstrapping im Compilerbau*

(6 Bonuspunkte)

Folgende Compiler stehen Ihnen zur Verfügung:

- Pascal-Compiler, geschrieben in C, mit Codegenerierung für SUN.
- ADA-Compiler, geschrieben in Pascal, mit Codegenerierung für SUN.



- Wie gehen Sie vor, um die beiden Compiler auf einer SUN-Workstation zum Laufen zu bringen? Welcher zusätzliche Compiler wird dazu benötigt, der auf der SUN-Workstation läuft?
- Portieren Sie anschließend den ADA-Compiler auf einen PC. Welche manuellen Anpassungen müssen dafür an den gegebenen Compilern vorgenommen werden?

Hinweis: Verwenden Sie zur Lösung der Aufgabe die T-Notation aus der Vorlesung, um die Transformationen darzustellen. Weitere Informationen zum Bootstrapping finden Sie auch in: *M. Nagl „Software-Technik: Methodisches Programmieren im Großen“, Kapitel 8.2.* Das Buch befindet sich im Handapparat in der Informatik-Bibliothek.

Aufgabe S6 *Dokumente des Requirements Engineering Prozesses*

(6 Bonuspunkte)

Während der Phase des Requirements Engineering werden viele verschiedene Dokumente erstellt.

- Geben Sie alle Dokumente an, die im RE-Prozess erstellt werden, und beschreiben Sie kurz deren Zweck und Inhalt.
- Welche Abhängigkeiten bestehen zwischen den verschiedenen Dokumenten?
- Welche der Dokumente gehören nicht zum Requirements Engineering im engeren Sinne? Warum müssen diese Dokumente dennoch in der Phase des Requirements Engineering erstellt werden?

Abgabe: 6.1.2009

Die Punkte dieser Sonder-Übung sind Bonuspunkte, die zusätzlich erworben werden können, um den eigenen Punktestand durch Mehrarbeit zu erhöhen. Sie werden nicht benötigt, um 100% der Übungspunkte im Semester zu erreichen. Dieses Übungsblatt kann sowohl im Team als auch einzeln bearbeitet werden.

Sie können Ihre Lösung zum obigen Termin bis spätestens 10:00 Uhr morgens am Lehrstuhl abgeben. Bitte vermerken Sie in jedem Fall die Namen und Matrikelnummern aller beteiligten Personen (maximal 3) und den Namen Ihrer Kleingruppenübung.

Aktuelle Informationen zur Vorlesung finden Sie auf den Webseiten des Lehrstuhls unter <http://se.rwth-aachen.de/swt>.