

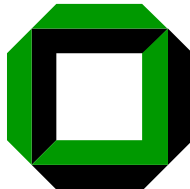
Formale Entwicklung objektorientierter Software

Praktikum im Wintersemester 2008/2009

Prof. P. H. Schmitt
Christian Engel, Benjamin Weiß

Institut für Theoretische Informatik
Universität Karlsruhe

22. Oktober 2008



Korrekte Software - zu teuer?

Software wird zunehmend in Systemen eingesetzt, bei denen Fehler teuer werden können, z.B. in

- Zugangskontrollen (security critical),



Korrekte Software - zu teuer?

Software wird zunehmend in Systemen eingesetzt, bei denen Fehler teuer werden können, z.B. in

- Zugangskontrollen (security critical),
- medizinische Apparate (safety critical),



Korrekte Software - zu teuer?

Software wird zunehmend in Systemen eingesetzt, bei denen Fehler teuer werden können, z.B. in

- Zugangskontrollen (security critical),
- medizinische Apparate (safety critical),
- Chipkarten (u.U. Austausch vieler Karten) etc.



Korrekte Software - zu teuer?

Software wird zunehmend in Systemen eingesetzt, bei denen Fehler teuer werden können, z.B. in

- Zugangskontrollen (security critical),
- medizinische Apparate (safety critical),
- Chipkarten (u.U. Austausch vieler Karten) etc.

Formale Spezifikation und Verifikation gewährleistet größtmögliche Sicherheit.



Was wird im Praktikum gemacht?

Durchlauf eines vollständigen Softwareentwicklungszyklus im Team angereichert um formale Methoden, d.h.

- Analyse, Entwurf, und Implementierung und Dokumentation von Software



Was wird im Praktikum gemacht?

Durchlauf eines vollständigen Softwareentwicklungszyklus im Team angereichert um formale Methoden, d.h.

- Analyse, Entwurf, und Implementierung und Dokumentation von Software
- Spezifikation und Verifikation der Software mit JML, ESC/Java2 und KeY



Was wird im Praktikum gemacht?

Durchlauf eines vollständigen Softwareentwicklungszyklus im Team angereichert um formale Methoden, d.h.

- Analyse, Entwurf, und Implementierung und Dokumentation von Software
- Spezifikation und Verifikation der Software mit JML, ESC/Java2 und KeY

Die notwendigen Kenntnisse werden in einzelnen Übungseinheiten vermittelt und mit Übungsblättern vertieft.



Anforderungen

- Programmierkenntnisse (Java)



Anforderungen

- Programmierkenntnisse (Java)
- *Vorlesung Formale Systeme*



Anforderungen

- Programmierkenntnisse (Java)
- *Vorlesung Formale Systeme*
- *Vorlesung Softwaretechnik* (UML, SW-Entwicklung)



Organisatorisches

Gleich



Organisatorisches

Gleich

- Einteilung in Gruppen mit 2–3 Mitgliedern



Organisatorisches

Gleich

- Einteilung in Gruppen mit 2–3 Mitgliedern
- Terminvereinbarung



Organisatorisches

Gleich

- Einteilung in Gruppen mit 2–3 Mitgliedern
- Terminvereinbarung

Scheinvergabe



Organisatorisches

Gleich

- Einteilung in Gruppen mit 2–3 Mitgliedern
- Terminvereinbarung

Scheinvergabe

- Anwesenheit bei den offiziellen Terminen



Organisatorisches

Gleich

- Einteilung in Gruppen mit 2–3 Mitgliedern
- Terminvereinbarung

Scheinvergabe

- Anwesenheit bei den offiziellen Terminen
- Bearbeitung der Übungsblätter: Alle Übungsblätter sind zu bearbeiten und abzugeben. Ein ernsthaftes Bemühen, die Aufgaben zu lösen, muss erkennbar sein.



Organisatorisches

Gleich

- Einteilung in Gruppen mit 2–3 Mitgliedern
- Terminvereinbarung

Scheinvergabe

- Anwesenheit bei den offiziellen Terminen
- Bearbeitung der Übungsblätter: Alle Übungsblätter sind zu bearbeiten und abzugeben. Ein ernsthaftes Bemühen, die Aufgaben zu lösen, muss erkennbar sein.
- Bearbeitung und Lösung des Abschlussprojekts (inklusive schriftlicher Ausarbeitung und Vortrag).

