

Spezifikation und Verifikation
Kapitel 0
Organisatorisches

Frank Heitmann
heitmann@informatik.uni-hamburg.de

2. April 2014

Aufbau der Veranstaltung

Die Vorlesung besteht aus drei “Teilen”:

Aufbau der Veranstaltung

Die Vorlesung besteht aus drei "Teilen":

Vorlesung

Einem Teil Vorlesungen.

Aufbau der Veranstaltung

Die Vorlesung besteht aus drei "Teilen":

Vorlesung

Einem Teil Vorlesungen.

Seminar

Einem seminarartigen Teil.

Aufbau der Veranstaltung

Die Vorlesung besteht aus drei "Teilen":

Vorlesung

Einem Teil Vorlesungen.

Seminar

Einem seminarartigen Teil.

Projekt

Einem projektartigen Teil.

Aufbau der Veranstaltung

Die Vorlesung besteht aus drei "Teilen" :

Vorlesung

Einem Teil Vorlesungen.

Seminar

Einem seminarartigen Teil.

Projekt

Einem projektartigen Teil.

Der blaue Block

Freizeit und "blau machen" ist nicht vorgesehen ;-)

Die Idee

Die Idee ist, dass ihr

Die Idee

Die Idee ist, dass ihr

- In der Vorlesung Grundlagen zu verschiedenen Themen präsentiert kriegt (Wissen in der Breite)

Die Idee

Die Idee ist, dass ihr

- In der Vorlesung Grundlagen zu verschiedenen Themen präsentiert kriegt (Wissen in der Breite)
- Im Seminar euch individuell spezialisiert (Wissen in die Tiefe bei einem Thema)

Die Idee

Die Idee ist, dass ihr

- In der Vorlesung Grundlagen zu verschiedenen Themen präsentiert kriegt (Wissen in der Breite)
- Im Seminar euch individuell spezialisiert (Wissen in die Tiefe bei einem Thema)
- Im Projekt dieses Wissen durch praktisches Ausprobieren/Implementieren festigt und ggf. weiter vertieft.

Der Ablauf

- Die Vorlesung ist üblicherweise Freitags (in den ersten zwei Monaten)

Der Ablauf

- Die Vorlesung ist üblicherweise Freitags (in den ersten zwei Monaten)
- Der Seminarteil ist Mittwochs (in den ersten einanhalb Monaten).

Der Ablauf

- Die Vorlesung ist üblicherweise Freitags (in den ersten zwei Monaten)
- Der Seminarteil ist Mittwochs (in den ersten einanhalb Monaten).
- Der Projektteil schließt an den Seminarteil an.

Der Ablauf

- Die Vorlesung ist üblicherweise Freitags (in den ersten zwei Monaten)
- Der Seminarteil ist Mittwochs (in den ersten einanhalb Monaten).
- Der Projektteil schließt an den Seminarteil an.

Anmerkung

Seminar- und Projektteil ist überwiegend zur Eigenarbeit vorgesehen!

Der weitere Ablauf

Der weitere Ablauf für heute und die nächsten Tage...

Der weitere Ablauf

Der weitere Ablauf für heute und die nächsten Tage...

- Ich stelle jetzt noch ein paar Themen/Paper vor.

Der weitere Ablauf

Der weitere Ablauf für heute und die nächsten Tage...

- Ich stelle jetzt noch ein paar Themen/Paper vor.
- Dann habt ihr bis **nächsten Freitag** Zeit euch die Paper zu besorgen und anzugucken.

Der weitere Ablauf

Der weitere Ablauf für heute und die nächsten Tage...

- Ich stelle jetzt noch ein paar Themen/Paper vor.
- Dann habt ihr bis **nächsten Freitag** Zeit euch die Paper zu besorgen und anzugucken.
 - **Alle** Paper ansehen - oder viele ;)
 - Überall z.B. den Abstract durchlesen, ungefähr einordnen, worum es geht.
 - Bei einzelnen, euch interessierenden Papern in die Tiefe gehen.

Der weitere Ablauf

Der weitere Ablauf für heute und die nächsten Tage...

- Ich stelle jetzt noch ein paar Themen/Paper vor.
- Dann habt ihr bis **nächsten Freitag** Zeit euch die Paper zu besorgen und anzugucken.
 - **Alle** Paper ansehen - oder viele ;)
 - Überall z.B. den Abstract durchlesen, ungefähr einordnen, worum es geht.
 - Bei einzelnen, euch interessierenden Papern in die Tiefe gehen.
- Nächsten Freitag sprechen wir dann genauer über die Paper und welche Themen euch interessieren.
- Ihr könnt dann dieses Paper zu eurem Seminar/Projekt-Schwerpunkte machen.

Literatur

- A Survey of Automated Techniques for Formal Software Verification - D'Silva, Kroening, Weissenbacher
- Deductive Software Verification - Jean-Christophe Filliatre
- Automated Deduction for Verification - Natarajan Shankar
- Software Model Checking - Jhala, Majumdar
- Software Analysis and Model Checking - Holzmann
- An Automata-Theoretic Approach to Branching-Time Model Checking - Kupferman, Vardi, Wolper
- Minimising Deterministic Bchi Automata Precisely Using SAT Solving - Ehlers

Literatur

- A Tutorial on Runtime Verification - Falcone, Havelund, Reger
- A brief account of runtime verification - Leucker, Schallhart
- *jUnit^{RV}* - Adding Runtime Verification to jUnit – Decker, Leucker, Thoma
- The software model checker BLAST - Beyer, Henzinger, Jhala, Majumdar
- Software Model Checking: The VeriSoft Approach - Godefroid

Literatur

- Designing a Theorem Prover - Paulson
- Simplify: A Theorem Prover for Program Checking - Detlefs, Nelson, Saxe

Literatur

- The Quest for Security in Mobile Ad Hoc Networks - Hubaux, Buttyan, Capkun
- Security in Mobile Ad Hoc Networks: Challenges and Solutions - Yang, Luo, Ye, Lu, Zhang
- A Secure Mobile Agent System Model Based on Extended Elementary Object System - Ma, Tsai, Murata
- Formal Modeling and Analysis of a Secure Mobile-Agent System - Ma, Tsai

Literatur

- Mobile Ambients - Cardelli, Gordon
- Anytime, Anywhere. Modal Logics for Mobile Ambients - Cardelli, Gordon
- Model Checking Mobile Ambients - Charatonik, Zilio, Gordon, Mukhopadhyay, Talbot

Projektideen

- (Mini-)Theorem Prover bauen
- (Mini-)SPIN bauen (evtl. mit mehreren Gruppen)
- RunTime Verification
- Ansätze/bestehende Tools benutzen, um die Implementation(!) (einfacher) Algorithmen zu verifizieren
- Programmiersprache programmieren
- Dinge parallelisieren oder Optimieren (sprich: bekannte Algorithmen schneller machen)

Bis...

Bis zum nächsten Mal:

Aufgabe

Paper angucken und überlegen, was ihr spannend fändet.

Bis...

Bis zum nächsten Mal:

Aufgabe

Paper angucken und überlegen, was ihr spannend fändet.

Anmerkung

Nächstes Treffen am **Freitag, 11.4!**

Und jetzt...

Und jetzt ...

auf auf!

Forscht! ;)