Analyse und Verifikation

LVA 185.276, VU 2.0, ECTS 3.0
SS 2012

Jens Knoop

Technische Universität Wien
Institut für Computersprachen

Motivation 1(4)

Programme und Programmsysteme werden

- zunehmend komplexer
- verstärkt eingesetzt auch in sicherheitskritischen Anwendungen mit unmittelbarer Verantwortung für
  - Leib und Leben (Flugüberwachungssoftware, fly-by-wire, drive-by-wire, (Industrie-) Anlagensteuerung (Kraftwerke, Chemieanlagen, etc.),...)
  - Vermögen (Finanzindustrie, Buchhaltungs-IT,...)

Konsequenz

- Testen allein nicht ausreichend
- Formale Methoden erforderlich

Motivation 2(4)

Tatsächlich wird der

- Einsatz formaler Methoden zur
  - Analyse, Verifikation und Transformation

auch in der Industrie (Airbus, Boing, Microsoft,...) zunehmend üblich.

Motivation 3(4)

Zwei aktuelle Referenzen:

- Steve P. Miller, Michael W. Whalen, Darren D. Cofer. Software Model Checking Takes Off. Communications of the ACM, Vol. 53, No. 2, Feb. 2010, 58-64.
  "Although formal methods have been used in the development of safety- and security-critical systems for years, they have not yet achieved widespread industrial use in software or systems engineering. However, two important trends are making the industrial use of formal methods practical [...]
  The second is the growing power of formal verification tools, particularly model checkers."

Motivation 4(4)

"How Coverity built a bug-finding tool, and a business, around the unlimited supply of bugs in software systems..."

- Al Bessey, Ken Block, Ben Chelf, Andy Chou, Bryan Fulton, Seth Hallem, Charles Henri-Gros, Asya Kamsky, Scott McPeak, Dawson Engler. A Few Billion Lines of Code Later: Using Static Analysis to Find Bugs in the Real World. Communications of the ACM, Vol. 53, No. 2, Feb. 2010, 66-75.

  "In 2002, Coverity commercialized a research static bug finding tool. [...] We built our tool to find generic errors (such as memory corruption and data races) and system-specific or interface-specific violations (such as violations of function-ordering constraints. The tool, like all static bug finders, leveraged the fact that programming rules often map clearly to source code; thus static inspection can find many of their violations."

Inhalt der Lehrveranstaltung

Korrektheit, Vollständigkeit, Optimalität in

- Programmverifikation
  Methode von Hoare, Korrektheit, Vollständigkeit
  stärkste Nachbedingungen, schwächste Vorbedingungen,...
- Programmanalyse
  Datenflussanalyse, abstrakte Interpretation, Model-checking,...
- Programmtransformation, speziell in der Optimierung

Ziele der Lehrveranstaltung

- Vertiefte Einsicht in fundamentale Prinzipien und Konzepte in Programmverifikation, -analyse und -transformation.
- Herausarbeiten und Verstehen von Gemeinsamkeiten, Analogien und Unterschieden zwischen Programmverifikation und -analyse.
- Erkennen, Einschätzen und Bewerten der Möglichkeiten und Grenzen insbesondere automatischer Programm-analyse, -verifikation und -optimierung.

Was Sie mitbringen sollten...

Voraussetzungen

- Grundlagen in Theoretischer Informatik, grundlegende Programmierkenntnisse
- Kenntnisse im Übersetzerbau sind hilfreich, etwa aus der LVA Übersetzerbau 185.A48 VU 4.0 (oder vormals LVA Übersetzerbau 185.311 VL 3.0, bzw. aus einer vergleichbaren Veranstaltung), aber nicht zwingend erforderlich
- Abgeschlossenes Bachelor-Studium, falls Anrechnung für Master-Studium geplant

Eine perfekte Grundlage(nergänzung)

...und Vertiefung in diesem Semester durch Mitbesuch von:

- LVA 185.A48 Übersetzerbau, VU 4.0, ECTS 6.0,
  Prof. Dr. Andreas Krall, Prof. Dr.Anton Ertl:
  `www.complang.tuwien.ac.at/ubvl/index.html`
- LVA 185.A49 Abstrakte Maschinen, UE 2.0, ECTS 3.0,
  Prof. Dr. Andreas Krall:
  `www.complang.tuwien.ac.at/ubvl/index.html`
- LVA 185.A50 Dynamische Übersetzer, VU 3.0, ECTS 3.0,
  Prof. Dr. Andreas Krall:
  `www.complang.tuwien.ac.at/ubvl/index.html`

Anrechenbarkeit

...non scholae, sed vitae discimus.

Anrechenbar für die Master-Studiengänge:

- 066 931 Computational Intelligence
- 066 937 Software Engineering & Internet Computing

Aufbau der Lehrveranstaltung

- Vorlesung
- Übung in 2er-Gruppen
- Vorlesungsprüfung

  (über Vorlesungsstoff und Übungsbeispiele)

Literatur

1. Janusz Laski, William Stanley. *Software Verification and Analysis. An Integrated, Hands-On Approach*. Springer, 2009.
2. Hanne R. Nielson, Flemming Nielson. *Semantics with Applications: An Appetizer*, Springer, 2007.
3. Krzysztof R. Apt, Ernst-Rüdiger Olderog. *Verification of Sequential and Concurrent Programs*, Springer, 1997.
4. Flemming Nielson, Hanne R. Nielson, Chris Hankin. *Principles of Program Analysis*, Springer, 1999.
5. Stephen S. Muchnick. *Advanced Compiler Design and Implementation*, Morgan Kaufmann, San Francisco, California, 1997.
6. Weitere Literatur, insbesondere Originalarbeiten, wird im Verlauf der Veranstaltung angegeben.

Anmeldung

In zwei Stufen:

- Stufe 1: *Anmeldung*
  ...jeder für sich über ein arbeitsbereichsinternes elektronisches Anmeldesystem bis Freitag, den 25.03.2012:
  `http://www.complang.tuwien.ac.at/anmeldung`
- Stufe 2: *Gruppenbildung*
  ...mithilfe desselben Anmeldesystems: eines der Gruppenmitglieder kreiert die Gruppe

Siehe auch Angaben zur Anmeldung auf der Webseite der LVA:
`www.complang.tuwien.ac.at/knoop/auv185276_ss2012.html`

Vorlesung und Übung

- Dienstags von 16:30 - 18:00 Uhr s.t., EI 3a Hörsaal (Gußhausstr. 25-29, 2. Stock, 1040 Wien.
  - 16:30 Uhr - bis ca. 17:30 Uhr: Vorlesung
  - ab ca. 17:30 Uhr - 18:00 Uhr: Übung (Abgabe und Besprechung der Übungsaufgaben der Vorwoche, Ausgabe der neuen Übungsaufgaben (im Web erhältlich))
- beginnend mit dem 20.03.2012 im Regelfall jeden Dienstag ein neues Aufgabenblatt
- insgesamt ca. 8 Abgaben

Beurteilung

- Mündliche Prüfung zu Vorlesung und Übung.

Bei Fragen und Problemen

- Webseite:
  `www.complang.tuwien.ac.at/knoop/auv185272_ss2012.html`
- Fragen und Antworten in Vorlesung und Übung
- E-Mail: `knoop@complang.tuwien.ac.at`
- Sprechstunde: mittwochs, 15 Uhr - 16 Uhr
  (bitte vorher kurz anmelden)

Interesse an einem geförderten Auslandsstudium?

Die Erasmus/LLP-Programmlinie der EU bietet eine Vielzahl lohnender Möglichkeiten, z.B.

- Universidade do Minho, Braga, Portugal
- Technical University of Lissabon, Portugal
- The University of Copenhagen, Dänemark
- Universität Halle-Wittenberg, Deutschland
- Universität Paderborn, Deutschland
- Universidad Politècnica de València, Spanien
- ...

Mehr dazu auf:
`http://www.complang.tuwien.ac.at/knoop/erasmus`

Ich wünsche Ihnen

...viel Erfolg bei dieser Lehrveranstaltung und dass Sie auch über die unmittelbare Veranstaltung hinaus davon profitieren!

Zu guter Letzt:
Vorlesung und Übung leben mit Ihnen! Ihre Rückmeldungen, Anregungen, Verbesserungsvorschläge sind willkommen!
Natürlich auch Hinweise, wenn Ihnen etwas gut gefallen hat!