Analyse und Verifikation

LVA 185.276, VU 2.0, ECTS 3.0
SS 2013

– Vorbesprechung –

(Stand: 01.03.2013)

Jens Knoop

Technische Universität Wien
Institut für Computersprachen

Ausgangspunkt

Programme und Programmsysteme sind zunehmend unverzichtbar auch zur Steuerung

- sicherheitskritischer Anwendungen und Systeme mit unmittelbaren Auswirkungen und Verantwortung für
  - Leib und Leben (Medizintechnik (Operationsroboter, Bestrahlungsgeräte), Flug- und Automobilbau (fly-by-wire, drive-by-wire, führerlose Fahrzeuge), (Industrie-) Anlagensteuerung (Kraftwerke, Chemieanlagen), Infrastruktur zur Daseinsvorsorge (Strom- und Telefonnetze), ...)
  - Hohe Sach- und Vermögenswerte (Unternehmens-IT, Finanzindustrie,...)

Gleichzeitig werden diese Systeme immer

- komplexer!

Konsequenz

- Testen allein zur Qualitätssicherung nicht ausreichend.
- Formale Methoden zur Programmanalyse und -verifikation unverzichtbar!

Einige weit bekanntgewordene Beispiele

...in Stichworten:

- Pentium-Bug
- Ariane-Absturz
- Toyota-Prius
- Mars-Sonde Pathfinder
- ...

$\rightsquigarrow$ mehr dazu siehe 1. Aufgabenblatt!

Beobachtung

Der Einsatz formaler Methoden zur

- Analyse, Verifikation und Transformation (Optimierung)

von Programmen und Programmsystemen wird zunehmenden Bereichen der Industrie (Airbus, Boing, Microsoft,...) mehr und mehr üblich und selbstverständlich.

Zwei aktuelle Referenzen zum Einstieg 1(2)

- Steve P. Miller, Michael W. Whalen, Darren D. Cofer. Software Model Checking Takes Off. Communications of the ACM 53(2):58-64, 2010.

  "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."

Zwei aktuelle Referenzen zum Einstieg 2(2)

"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 53(2):66-75, 2010.

  "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."

Inhalte und Themen der Lehrveranstaltung

Korrektheit, Vollständigkeit, Optimalität in

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

Ziele der Lehrveranstaltung

- Vertiefte Einsicht in fundamentale Prinzipien und Konzepte in Programmanalyse, -verifikation und -transformation (-optimierung).
- Herausarbeiten und Verstehen von Gemeinsamkeiten, Analogien und Unterschieden zwischen Programm-analyse und -verifikation.
- 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, etwa aus der LVA Übersetzerbau 185.A48 VU 4.0 (oder vormals LVA Übersetzerbau 185.311 VL 3.0, bzw. aus einer vergleichbaren Veranstaltung) sind hilfreich, wenn auch nicht zwingend erforderlich
- Abgeschlossenes Bachelor-Studium

Eine perfekte (Grundlagen-) Ergänzung

...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

Ablauf und Beurteilung der Lehrveranstaltung

- Vorlesungsteil (i.d.R. wöchentlich)
- Übungsteil in 2er-Gruppen (i.d.R. wöchentlich)
- Mündliche Abschlussprüfung (über Vorlesungs- und Übungsstoff)

Literaturhinweise

- Janusz Laski, William Stanley. *Software Verification and Analysis*. Springer-V., 2009.
- Hanne Riis Nielson, Flemming Nielson. *Semantics with Applications: An Appetizer*. Springer-V., 2007.
- Krzysztof R. Apt, Frank S. de Boer, Ernst-Rüdiger Olderog. *Verification of Sequential and Concurrent Programs*. 3. Auflage, Springer-V., 2009.
- Flemming Nielson, Hanne Riis Nielson, Chris Hankin. *Principles of Program Analysis*. 2. Auflage, Springer-V., 2005.
- Stephen S. Muchnick. *Advanced Compiler Design Implementation*. Morgan Kaufman Publishers, 1997.

...weitere Literaturhinweise, insbesondere auf Originalarbeiten, werden im Verlauf der Veranstaltung angegeben.

Anmeldung

Zweistufig:

- 1. Stufe: Individuelle Anmeldung
  ...jeder für sich über ein arbeitsbereichsinternes elektronisches Anmeldesystem bis Freitag, den 15.03.2013:
  www.complang.tuwien.ac.at/anmeldung
- 2. Stufe: 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_ss2013.html

Zeit und Ort für Vorlesung und Übung

- Dienstags, 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)).
- Im Regelfall jeden Dienstag, beginnend mit dem 05.03.2013, ein neues Aufgabenblatt; insgesamt 8-10 Abgaben.

Bei Fragen und Problemen

- Webseite der Lehrveranstaltung:
  `www.complang.tuwien.ac.at/knoop/auv185276_ss2013.html`
- Fragen und Antworten in Vorlesung und Übung
- Sprechstunde: Mittwochs, 15 Uhr - 16 Uhr
  (bitte vorher kurz anmelden)
- E-Mail: `knoop@complang.tuwien.ac.at`

Interesse an gefördertem Auslandsstudium?

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

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

Mehr dazu: `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!

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