

Statische Analyse

Seminar in Wintersemester 2000/2001

Lehrstuhl für Softwaretechnologie, Prof. Dr. Willem-Paul de Roever

Fehler in Programmen sind, je nach Anwendung, lästig bis nicht tolerierbar. Es gibt eine Vielzahl von Methoden, die es theoretisch erlauben, die Korrektheit von Programmen nachzuweisen. Der Pferdefuß in der Praxis ist dabei, daß sie für große, realistische Probleme immensen bis unüberwindlichen Aufwand erfordern, sodaß sie nur in Extremfällen von sicherheitskritischen Systemen und/oder Spezialanwendungen wie z.B. der Hardwareverifikation zur Anwendung kommen.

Um routinemäßig Fehler aus großen industriell entwickelten Programmen zu beseitigen, sind demgegenüber ausschließlich Methoden akzeptabel, die häufig auftretende Fehlerklassen automatisch und sehr effizient behandeln.

Die statische Analyse ist ein Oberbegriff für eine Vielzahl solcher Methoden. Beispiele für solche Problemstellungen sind das Aufspüren von Typinkonsistenzen mittels Type-Checking, Auffinden von potentiellen Null-Pointerdereferenzierungen bis hin zu Sicherheitsüberprüfungen beim Herunterladen von Java-Applets.

Der Name "Statische Analyse" erklärt sich dadurch, daß die Analyse nicht zur Laufzeit stattfindet, sondern vorab durchgeführt wird.

In dem Seminar werden ausgewählten Kapitel aus dem Buch [1] behandelt. Je nach Umfang des Stoffes werden die Kapitel an ein oder zwei Seminarteilnehmer vergeben. Als Themen bieten sich folgende an:

- Datenflussanalyse
- Interprozedurale Datenflussanalyse
- Kontrollflussanalyse
- Abstrakte Interpretation
- Typ- und Effektsysteme
- Algorithmische Fragen

Anmeldung und bei Fragen zum Seminar: Email an `ms@informatik.uni-kiel.de`. Weitere Informationen unter:

<http://www.informatik.uni-kiel.de/inf/deRoever/WS0001/Seminar/>

Literatur

- [1] Flemming Nielson, Hanne-Riis Nielson, and Chris Hankin. *Principles of Program Analysis*. Springer-Verlag, 1999.