Sobre el curso

Ma. Laura Cobo

Métodos formales para Ingeniería de Software
Departamento de Ciencias e Ingeniería de la Computación
Universidad Nacional del Sur
Argentina

Verificación formal

Las técnicas de verificación basada en modelos están basadas en:

Una descripción del comportamiento del sistema

De una manera precisa, matemática y no ambigua

Algoritmos

Exploran sistemáticamente todos los estados del modelo del sistema

Generalmente, el modelo preciso del sistema, conduce a descubrir que el mismo es:

- Incompleto
- Ambiguo
- Inconsistente con la especificación informal del sistema

La exploración de los estados del modelo proveen las bases para el rango de técnicas de verificación:

- Exhaustiva (Model Checking)
- Restringida a un conjunto de escenarios (Simulación)
- Real (Testing)

Objetivos del curso

- Comprender cómo los métodos formales ayudan a producir software de mejor calidad.
- Aprender acerca de lenguajes de modelado formales
- Aprender acerca de especificaciones formales
- Leer y comprender especificaciones de requerimientos
- Conocer las principales aproximaciones desarrolladas para verificación formal de software
- Utilizar herramientas para validar modelos y código

Temas principales

Especificación

- Diseño de alto nivel
- Diseño de sistemas y propiedades de comportamiento
- Propiedades a nivel código

Verificación

- **Encontrar modelos / chequeo**

  típicamente se trabaja con herramientas automáticas y abstactas (nivel modelo)
- **Verificación deductiva**

  típicamente se trabaja con herramientas semi-automáticas y precisas (nivel implementación, código fuente)

Diseño de alto nivel – Model checking

Utilizaremos el lenguaje **Alloy**

- Se trata de un lenguaje de modelado para diseño de software
- Ameno para realizar análisis completamente automáticos
- Resulta apropiado para expresar restricciones estructurales y de comportamiento de sistemas de software
- Herramienta de modelado basada en lógica de primer orden
- Analizador automático basado en *SAT solvers*

Objetivos

- *Diseñar y modelar sistemas en el lenguaje Alloy*
- *Chequear propiedades con el analizador Alloy*
- *Comprender qué puede y no puede ser expresado en Alloy*

Desarrollo basado en modelos

Fuera del alcance del curso

Model checkers

- Se trata de especificaciones en modelos ejecutables
- Se utiliza sobre aplicaciones concretas de la industria
- Habilidades para probar propiedades que dependen de la evolución del sistema
- Diseño automático utilizando herramientas basadas en técnicas de model checking

Objetivos

- *Diseñar y modelar sistemas en lenguaje Promela*
- *Chequear propiedades con el analizador SPIN*
- *Comprender qué puede y no puede ser expresado en Promela-SPIN*

Especificaciones a nivel código

Utilizaremos el lenguaje **JML**

- Especificación de interfaces de comportamiento para modulos JAVA
- Basado en el paradigma de diseño por contrato
- Posee la misma sintaxis y semántica formal del lenguaje de programación para el que se hace la especificación
- La especificación queda embebida en el código

Objetivos

- *Escribir especificaciones formales y contratos en JML*
- *Comprender como el código fuente y las especificaciones se representan en lógica.*
- *Verificar propiedades funcionales con KeY*