

Лекция 1: "Формални методи и формална система" - Резюме

Моделиране:

- **Модел:** Опростено представяне на реален обект или система, което позволява анализ, проектиране и проверка.
- **Ползи от моделирането:**
 1. **Абстракция** - подчертаване на важните детайли
 2. **Анализ** и познание-начин да се разбира системата
 3. **Дефиниране и проверка на взаимодействия** - яснота/негъвусмислие в писанията
 4. Проектни решения-решения, свързани с практическата реализация

Модели на софтуерни системи- някаква абстракция на системата или на част от нея: (част от) спецификация на изискванията, проект, код, тест, поведение на системата. Моделът на софтуерна система се представя чрез

използване на твърдения, формули, математическа

символика, схеми и други средства. Използват се за проектиране, тестване и проверка преди реализация.

Формални методи(ФМ)- Математически техники за разработване и анализ на софтуерни системи.

Приложение на формалните методи:

- Проектиране, разработване, тестване, експл, и поддръжка.
- Проверка за коректност, консистентност и непълноти.

Примери за успешни приложения:

- **CICS** (система за банкови трансакции).

- **B-method** (упр. на метрото в Париж, без грешки в софтуера)

Основни компоненти на Формална система (ФС):

- Формален език (азбука, граматика).
- Система за извод (аксиоми, правила за извод, теореми).

Цели на ФС:

- Доказателства за изисквания и проектиране.
- Осигуряване на съгласуваност и валидност на спецификациите.

Характеристики на формалните методи:

- **Абстракция:** Премахване на ненужни детайли за фокусиране върху същественото. Абстракциите се предоставят от модели. Съществено качество на добрата спецификация е подходящият избор на ниво абстракция.
- **Доказателство:** Повишава качество и осигурява индустриално приложение на софтуера.
- Класификация:
 1. **Полуформални методи** (езици, диаграми) – използват естествен език, диаграми, табл. и проста нотация, структурен и ОО-анализ
 2. **Формални методи** -използват математически синтаксис и семантика; примери: Z, B, VDM, LOTOS, CSP

Предимства на формалните методи:

- Намаляват грешките и неточностите.
- Осигуряват яснота и строгост в ранните етапи на разработка.

- Намаляват разходите чрез ранна проверка за коректност - намалят грешките
- Могат да проверят коректността и точността, непълнотата и неконсистентността на спецификацията

Ограничения:

- Трудност за разбиране и използване
- Трудна формализация на отделни аспекти на системата
- Трудна възвръщаемост на инвестицията, консерватизъм и нежелание

Формалните методи са основа за надеждно и ефективно проектиране на софтуерни системи, особено в критични приложения.

Model-based notations – Z

Process algebras -based notations - Communicating Sequential Processes (CSP)

Компоненти на ФС:

- Синтаксис (Syntax)- специфична нотация за представяне на спецификацията. Често се базира на синтаксиса на теорията на множествата и предикатните изчисления.
- Връзки (Relations);
- Семантика (Semantics) - помага да дефинираме обхвата от обекти, които се използват за описание на системата.

Формална система (ФС) – осн.елементи: език, правила за мислене; начин на определяне на значението на

твърденията/изказванията; начин на представяне и получаване на резултат (разсъждения и интерпретация)

Формалната система се дефинира чрез два основни елемента:

1) **формален език (синтаксис): азбука/символи/ + Синтактични правила (граматика) - мета-език**

2) **система за извод/заклучение(семантика): аксиоми + правила за извод, теореми, доказателство, умозаклучение**

Синтактични правила (граматика) – множество от правила, с които се конструират изречения/твърдения в дадена система (well-formed formulae (wff) на езика

Пропозиционна логика- Пропозиционната логика е раздел от математическата логика, който изучава логическите изрази (пропозиции) и тяхното свързване чрез логически оператори като:

"И" (конюнкция) – \wedge

"ИЛИ" (дизюнкция) – \vee

"НЕ" (отрицание) – \neg

"Ако...то..." (импликация) – \rightarrow

"Само ако..." (еквиваленция) – \leftrightarrow

Предикатната логика разширява пропозиционната логика, като въвежда **квантори** \forall , \exists , /и **предикати**/функции, които зависят от един или повече аргумента/, които описват свойства на обекти или връзки между тях.

Семантика - за да зададем значение на символите:

- избираме семантични/предметна област (domain);

- формулираме интерпретация, която обяснява как изреченията в синтактичната област придобиват значение в семантичната област.

Системи за извод (Inference Systems) - Дедуктивен апарат за манипулиране на wffs. Два компонента:

- > **Аксиоми:** wffs, които могат да бъдат записани без да се отнасяме до някои групи wffs в езика.

- > **Правила за извод (Inference rules):** правила, позволяващи да създаваме wffs като пряко следствие от гр. wffs.

Доказателство в една формална система е последователност от в асоциирания формален език, в която всяко wff е **аксиома** или пряко следствие от едно или повече предшестващи wffs /теорема/

Формална система = формален език + система за извод

Интерпретация на формална система, в която wffs са изречения, които могат да бъдат истина или лъжа е:

- > **съгласувана (консистентна/sound)**, ако всяка теорема на системата интерпретира истинно изказване;

- > **пълна (complete)**, ако всяко истинно изказване може да бъде доказано като теорема

Повечето формални системи са съгласувани, но непълни.

Получаване (извличане) във формална система е процес, при който показваме, че дадено твърдение WWW е вярно, като го докажем стъпка по стъпка въз основа на:

1. **Аксиоми** – твърдения, които приемаме за винаги верни.
2. **Предпоставки (premises)** – допълнителни твърдения, които приемаме за верни за конкретното доказателство.
3. **Логически правила** – как извличаме нови твърдения от предишните.

Пример: Ако имаме:

- **Аксиома:** "Ако вали, улицата е мокра."
- **Предпоставка:** "Вали."
Можем да **извлечем:** "Улицата е мокра."

Записваме това като: $P \vdash WP$ "Предпоставките P водят до W ".

Важно уточнение: Съществува риск да се докажат **неверни твърдения**, ако: 1/Предпоставките са неверни. 2/Логическите правила са приложени неправилно.

Затова коректността на доказателството зависи както от **предпоставките**, така и от начина на извличане.

Формалното описание ни позволява да проектираме и построяваме системи, защото: *Точно описва изискванията *Открива недостатъците в проектирането *Реализира рационални приложения

Характеристики: * Abstraction allows you to focus on the important aspects of a problem.* Mathematics allows you to reason about solutions to the problem. • Базиран се на изграждане на формални системи