G53FSP
Formal Specification

Dr. Rong Qu

Introduction to Formal Specification

http://www.cs.nott.ac.uk/~rxq/g53fsp
Background

- A specification may have many uses, in many forms
  - Management
    - Specification in English
  - Programmers
    - Written in a pseudo (or real) programming language
  - CS & Management
    - Improve quality of software systems
    - Provide proof & verifications
Background

- Need to be precise
  - Management
    - Must be able to agree what is to be implemented
Background

- Need to be precise
  - Programming
    - Written in language is probably no use
      - Expensive to create
    - Long
    - Include un-necessary details
    - Performance not effect of operation
    - Difficult to understand
Background

- Precise definition of
  - Effects of various operations rather than its performance details
  - Information to be displayed to users

- No need of
  - How the operations are to be done
  - How the data is to be stored
  - Etc

The details is no use of specify the effect of the system
Background

- Also may need a specification to
  - Prove certain properties
  - Prove that certain combinations of states never occur
  - Value of a given variable never go outside certain bounds

- Correctness of system can be proven
Purpose of Formal Specification

- To state what system should do without describing how to do it

- To reduce faults in systems
  - Invest more effort is early stage of system development
  - Requirement errors can be discovered as early as possible and resolved
Specification Parts

- A specification need to include
  - Details of the system
    - The states it can occupy
    - Invariants which will always hold
  - Dynamic aspects
    - All operations which are possible
    - The relations of inputs to outputs
    - Changes of state that can occur
Specification Parts

- A specification will include
  - Functional requirements
    - The effect of xxx will be
    - The output of command yyy will be as specified in standard zzz
    - The system will produce a report on salesman effectiveness
Specification Parts

- A specification will include
  - Non-functional requirements (properties)
    - All data access should be via company supplied subroutines
    - The system should be immune to power failures
    - The response time must be ...
Specification Parts

- A specification will include
  - Design directives
    - The system will collect data from ...
    - The VDU display will be in the form ...
    - The designer will use SSADM
Specification Parts

- A specification will include
  - Goals
    - Response times should be minimised
    - It should run in 748kb of memory
  - Data statements
    - The system must maintain the average temperature over the preceding ...
Definition – Formal Specification

- The specification will be
  - A strict mathematical definition of the effect of the required operation
    - Usually expressed in mathematical notions with precisely defined vocabulary, syntax and semantics
  - Definition is not necessarily in the form in which it can be programmed
Definition – Formal Specification

- In computer science, formal methods refers to mathematically based techniques for the specification, development and verification of software and hardware systems.

Definition – Formal Specification

- The approach is especially important in high-integrity systems, for example where safety or security is important, to help ensure that errors are not introduced into the development process.

Definition

- Formal methods are particularly effective early in development at the requirements and specification levels.

Formal Methods Parts

- Program specification
- Program verification
- Automated theorem proving
- Model checking
Program Specification

- A program specification is the definition of what a computer program is expected to do.

- It can be:
  - informal, in which case it can be considered as a blueprint or user manual from a developer point of view, or
  - formal, in which case it has a definite meaning defined in mathematical or programmatic terms.
Program Verification

- In computer science, **program verification** is the process of formally proving that a computer program does exactly what is stated in the program specification it was written to realize.

- Program verification is more specific in that it aims to verify the code itself, not only some abstract model of the program.
  - Intel, AMD: verify chips
  - BMW: automotive system
Automated Theorem Proving

Automated theorem proving is the proving of mathematical theorems by a computer program. Depending on the underlying logic, the problem of deciding the validity of a theorem varies from trivial to impossible.
Model checking

*Model checking* is a method to algorithmically verify formal systems. This is achieved by verifying if the model, often deriving from a hardware or software design, satisfies a formal specification. The specification is often written as temporal logic formulas.
Formal vs. Informal Methods

- Formal specification vs. SSADM

- SSADM
  - now government standard
  - Widely used in industry
  - Semester 5 of CSiT
Formal vs. Informal Methods

- Lots of possible inconsistencies
  - Decision tables
    To prove that, if you use “don’t care” entries, the result is completely and uniquely defined?
- Computer tools to help overcome the difficulties, but are still basic problems
Mathematics vs. Natural Language for System Specification

- Deficiencies of natural language
  - Can be vague
  - Can be ambiguous
  - Can be self-contradictory
  - Can be incomplete
  - Encourages imprecise thinking
  - Cannot easily handle abstractions
Mathematics – good features

- Easily handle abstractions
- Can be used for reasoning about, and describing a system
- Is concise
- Is non-ambiguous
Mathematics – good features

- Is applied widely to the real world
- Can approximate where exactness is unnecessary
- Changes slower than computing
Formal Notation – drawbacks

- The customer cannot easily understand the specification

- The mathematics used is unfamiliar to many of the staff

- In real world, getting user requirements document is a problem
Z

- A formal specification technique developed at Oxford
- Uses very mathematical notation to provide exact definitions of a system
- System is described in a number of small Z modules, which can cross-refer each other
- Each module is expected to have some descriptive English text to help users to understand it
Summary

- Background of formal specification
- Parts of specification
- Formal specification vs.
  - SSADM
  - Natural language
- Formal method advantages and drawbacks
- Z specification language