I. Finite-State Machines and Model Checking

Robi Malik
Office: G 2.03
Phone: 07 838 4796
e-Mail: robi@cs.waikato.ac.nz

II. Refinement in Z

Steve Reeves
Office: G 1.23
Phone: 07 838 4398
e-Mail: stever@cs.waikato.ac.nz

Assessment
- 6 Assignments
  - Assignment 1 due Monday 13 March 9:00 A.M.
  - Assignment 2 due Monday 27 March 9:00 A.M.
  - Assignment 3 due Monday 10 April 9:00 A.M.
  - Assignment 4 due Monday 8 May 9:00 A.M.
  - Assignment 5 due Monday 22 May 9:00 A.M.
  - Assignment 6 due Monday 5 June 9:00 A.M.
- Final Exam
- 5th year students: please contact lecturer

Recommended Textbook:

Some more books on model checking:

Model Checking Software
VALID Toolset
- Type /home/robi/bin/valid to run
- More instructions in class

NuSMV
- Type NuSMV to run
- Download and documentation at http://nusmv.irst.itc.it
Reactive Systems

Definition:
A “reactive system” is a computer program that continuously interacts with its environment.

Graphical User Interfaces

- Input is presented while the system is running.
- User can continuously interact with the system.
- Correctness cannot be specified by correct output.

Transformational Systems

Definition:
A “transformational system” is a computer program that accepts some input, performs computation on it, produces output, and then terminates.

Embedded Systems

Technical devices
- Household equipment
- Automotive electronics
- Industrial plants

All these are …
- often safety-critical
- reactive systems

Examples
- Compute the sum of a list of numbers.
- Read a text file, compute how often each word occurs, and display the results graphically.

Hardware

Dealing with Bugs …

In Software
- Fix the bug
- Put a patch on your website

In Hardware
- Can be very expensive!
What is the Difference?

<table>
<thead>
<tr>
<th>Transformational System</th>
<th>Reactive System</th>
</tr>
</thead>
</table>

Reactive Systems

Reactive Systems are **very hard**

- to *design* …
- to *implement* and …
- to *get right*.

Properties of Reactive Systems

*Never blow up!*

Model Checkers

**Input**
- Finite-state model of a system
- Temporal logic property

**Output**
- "true" if the property is satisfied
- Otherwise a counterexample

Properties of Reactive Systems

Letter Sorting Machine

Model Checkers

- Finite-state model of a system
- Temporal logic property

Output
- "true" if the property is satisfied
- Otherwise a counterexample