

## CS 5219 Introduction

Abhik Roychoudhury  
National University of Singapore  
abhik@comp.nus.edu.sg  
<http://www.comp.nus.edu.sg/~abhik>

CS5219 2007-08 by Abhik

1

## What is it about ?

- Techniques to help reliable software development.
- Checking program **behavior**
  - Such as checking whether desired invariants hold at program control points.
- What is the programming language ?
  - Conventional languages like C/Java

CS5219 2007-08 by Abhik

2

## The motivation

- Software Productivity Techniques
  - Methods to write reliable software more productively
  - Testing, Debugging, Programming Environments, Checking ...
  - Crucial not only for enhancing economic growth but also *productivity growth*.

CS5219 2007-08 by Abhik

3

## What kind of programs ?

- Conventional sequential programs
  - C like programs
- **Multi-threaded** software for distributed sys.
  - E.g. Multi-threaded Java
  - Many behaviors due to thread interleaving
- **Reactive** software
  - In continuous interaction with environment
  - e.g. control software in embedded sys.

CS5219 2007-08 by Abhik

4

## Conventional development

- **Collect software requirements**
  - Programmers often do not collect complete sets of requirements.
- **Write code**
  - Good programming disciplines exist e.g. modular development
- **Test and Debug**
  - Code walkthrough, Peer review, Testing
  - Again informal and/or incomplete.

CS5219 2007-08 by Abhik

5

## So are we ...

... going to look at program debugging ?

- YES
  - All our validation techniques can be used for software debugging
- NO
  - We will not only look at conventional software engineering activities like testing.

CS5219 2007-08 by Abhik

6

## Do I have the background ?

- Students in PhD program
  - Certainly OK, if you have a CS undergraduate, otherwise come and see me – it should still be OK.
- M. Comp Students
  - Not OK if you do not have CS undergrad
  - Let us talk after today's class.
- 4<sup>th</sup> year undergraduate students
  - I need to briefly interview you to find out your course background.

CS5219 2007-08 by Abhik

7

## Assessment

- Midterm : 20 % (in 7<sup>th</sup> week after recess)
- Project : 30%
  - 15 marks for results and final report, 8 for innovation, 5 for presentation and 2 for interim report.
- Final Exam : 50 %
- All Exams will be **open book**.
  - **You can bring in any material**

CS5219 2007-08 by Abhik

8

## IVLE

- Lesson Plan
  - Updated every week
  - Weekly lectures and readings available here
- Discussion Forum
  - Post messages for query, discussion.
- Workbin
  - Submissions (e.g. project reports).

CS5219 2007-08 by Abhik

9

## Module Outline

- Introduction + Software Testing
- Software Debugging
- Software Debugging + Discussion on Term Project
- Program Properties
- SPIN – a modeling & validation tool
- Model Checking

CS5219 2007-08 by Abhik

10

## Module Outline

- Midterm
- Software Abstractions
- Abstraction refinement
- Reasoning about programs
  - Computer-aided only
- Project Presentations by students
  - In different groups.

CS5219 2007-08 by Abhik

11

## Discussion on Projects

- This is the challenge problem we will tackle as the term project in this module. Every student group will solve the same problem.  
<http://sqr1.mcmaster.ca/pacemaker.htm>

CS5219 2007-08 by Abhik

12

## Project Description

- The project will involve modeling the problem in SPIN model checker and performing verification. Every group will have to do this (**no other techniques/tools can be used**) --- so that the outputs from the different groups can be compared in an objective fashion for fair assessment. This is only the first step and everyone must complete this step to get any credit in the project. A better project will also use the SPIN model as guidance to generate C code. SPIN's language is quite close to C --- so this should be possible. You will also then be able to argue how/why you could use the SPIN model as guidance and the links between requirements, model and code. Finally, an excellent project will also have one group member writing the code without going through modeling and the two codes will be compared to clarify coding errors or errors in understanding the requirements.

There already exist some recent published attempts to solve this problem, see:

- Incremental Development of a Distributed Real-Time Model of a Cardiac Pacing System Using VDM, Lecture Notes in Computer Science, ISSN 0302-9743 (Print) 1611-3349 (Online), Volume 5014/2008, Book FM 2008: Formal Methods
- Formal Specification of a Cardiac Pacing System Book Series Lecture Notes in Computer Science, Publisher Springer Berlin / Heidelberg ISSN 0302-9743 (Print) 1611-3349 (Online) Volume 5850/2009 Book FM 2009: Formal Methods

CS5219 2007-08 by Abhik

13

## Project schedule

### Class Discussion

3<sup>rd</sup>/4<sup>th</sup> week, (24 August or 31 August)

### Submission of brief project plan

28 Sep, coincides with midterm!

### Project Presentations in groups

26 Oct, 2 Nov

### Final Project Report Due

9 Nov (last class, no class on that day!)

CS5219 2007-08 by Abhik

14

## Key Announcements

- Check with me if you have questions about enrolling for the module.
  - Specially undergrad and M. Comp students
- All exams are open book.
- Midterm is in the 7<sup>th</sup> week (after recess)
- No class in 13<sup>th</sup> week (discussions today)
- **No recording of lectures will be done**
  - Kindly come to class 😊

CS5219 2007-08 by Abhik

15