MONASH University
Information Technology

FIT3013
Formal specification for software engineering

Unit Guide

Semester 1, 2010

The information contained in this unit guide is correct at time of publication. The University has the right to change any of the elements contained in this document at any time.

Last updated: 17 Feb 2010

Table of Contents

FIT3013 Formal specification for software engineering - Semester 1, 2010

Chief Examiner:

Associate Professor John Hurst
Associate Professor
Phone: +61 3 990 55192 +61 3 990 32196
Fax: +61 3 990 55159

Lecturer(s) / Leader(s):

Clayton

Associate Professor John Hurst
Associate Professor
Phone: +61 3 990 55192 +61 3 990 32196
Fax: +61 3 990 55159

Introduction

This unit is a core 6 point unit in the Bachelor of Software Engineering. In this unit you will learn how to bring some mathematical formalism to the design and specification of software projects. The software tool, Rodin, will be used to support the methodology developed by the unit, namely, the Event-B method. Students will write and prove some simple software designs to see how the formalism can be used in a practical context.

Unit synopsis

Review of set theory, the predicate calculus, relations, relational algebra and formal specification concepts; algebraic and model based specifications; the role of formal specifications in software engineering. The Event-B notation, data and algorithm design; data and operation refinement; proofs of correctness; proof obligations.

Learning outcomes

At the completion of this unit students will have -
A knowledge and understanding of:

- fundamentals of the Event-B Method;
- applications of the Event-B Method;
- Event-B specifications;
- software Testing in the discrete domain;
- the role of proof obligations and consistent specifications;
- determination of Proof Obligation;
- the role of refinement in developing formal specifications.

Developed attitudes that enable them to:

- have an appreciation of the professional need to establish formal properties of software;
- have a belief that formal specifications can improve the quality of software.

Developed the skills to:

- use the Event-B notation to develop and prove software specifications;
- install a Event-B Toolkit on a Unix/Linux/Windows platform;
- write basic Event-B specifications;
- refine and extend more advanced Event-B specifications.

Contact hours

2 hrs lectures/wk, 1 hr tutorial/wk

Workload

- 2x1 hour lectures per week
- 1x1 hour tutorial per week
- 3 hours of contact material revision per week
- 3-6 hours (average) per week practising skills with the Rodin tool suite, and developing assignment solutions.

Unit relationships

Prerequisites

FIT2004 and one of MAT1830, MTH1112 or MAT1077

Prohibitions

CSE4213

Teaching and learning method

Teaching approach

Learning resources for the unit will be made available on-line through both the unit web pages at

http://www.csse.monash.edu.au/~ajh/teaching/fit3013/2010/index.xml

and the Moodle pages

http://moodle.med.monash.edu.au/course/view.php?id=TBA

Besides the scheduled lectures and tutorials, students will be expected to use additional hours in this unit in developing their understanding of, and skills in, the use of the Rodin software suite. Manuals for the software will be available through the web pages. Students are also expected to interact dynamically through the semester via the use of the Moodle discussion pages.

In 2010, an experimental system, iCALT, will be used to gather and track student feedback on the lectures and the lecture material.

Timetable information

For information on timetabling for on-campus classes please refer to MUTTS, http://mutts.monash.edu.au/MUTTS/

Tutorial allocation

On-campus students should register for tutorials/laboratories using the Allocate+ system: http://allocate.its.monash.edu.au/

Unit Schedule

Week	Date*	Topic	Key dates
1	01/03/10	Introduction to B, Set Theory	
2	08/03/10	Abstract machines	
3	15/03/10	Using Rodin	
4	22/03/10	Modelling	Discussion 1 due
5	29/03/10	Case Study	Assignment 1 due
Mid semester break			
6	12/04/10	Proof Obligations	
7	19/04/10	Generalised Substitutions	Discussion 2 due
8	26/04/10	Structuring Specifications	
9	03/05/10	Machine Composition	Discussion 3 due
10	10/05/10	Refinement	
11	17/05/10	Refinement	Assignment 2 due
12	24/05/10	Data and Code Correctness	Discussion 4 due
13	31/05/10	Revision	

*Please note that these dates may only apply to Australian campuses of Monash University. Off-shore students need to check the dates with their unit leader.

Improvements to this unit

The main concerns expressed with the unit last year relate to the usability of the software tool, Rodin. Accordingly, it is planned to devote a few lectures to the use of this tool early in the unit, so that students become more familiar with its use before the need to use it in assignments is required. In addition, for 2010, a new exercise has been created which will use the wiki tool to build a user manual for the software. This manual will be used for the benefit of future students.

Unit Resources

Prescribed text(s) and readings

Schneider, S., The B-Method: An Introduction, first edition, Palgrave, 2001 ISBN 0-333-79284-X.

Recommended text(s) and readings

Required software and/or hardware

Software Details: the Event-B Toolkit, **Rodin**

Vendor: open source

Software Title*: **Rodin**

Version*: 1.1

OS*: Linux, Mac or Windows (see http://www.event-b.org/platform.html)

First Semester Week Required :week 2

License Details*: open source

Software Details: LaTeX

Vendor:

Software Title*: LaTeX

Version*: latest

OS*: Mac OSX, Linux or Windows

First Semester Week Required :week 2

License Details*: Freeware

Equipment and consumables required or provided

The software is available for Mac OSX, Linux or Windows. Students can download whichever version they prefer to use.

Study resources

Study resources we will provide for your study are:

Available from the moodle page

http://moodle.med.monash.edu.au/course/view.php?id=364

Assessment

Overview

Examination (2 hours): 50%; In-semester assessment: 50%

Faculty assessment policy

To pass a unit which includes an examination as part of the assessment a student must obtain:

- 40% or more in the unit's examination, and
- 40% or more in the unit's total non-examination assessment, and
- an overall unit mark of 50% or more.

If a student does not achieve 40% or more in the unit examination or the unit non-examination total assessment, and the total mark for the unit is greater than 50% then a mark of no greater than 49-N will be recorded for the unit.

Assignment tasks

Assignment coversheets

Assignment coversheets are available via "Student Forms" on the Faculty website:
http://www.infotech.monash.edu.au/resources/student/forms/
You MUST submit a completed coversheet with all assignments, ensuring that the plagiarism declaration section is signed.

Assignment submission and return procedures, and assessment criteria will be specified with each assignment.

- **Assignment task 1**

  Title:
  Discussion
  Description:
  The Moodle discussion forum will be used to support class discussion about various topics during the semester. There will be starter topics posted by the lecturer, and all students are expected to make at least one individual original response to each posting.
  Weighting:
  10%
  Due date:
  within 4 weeks of starter posting, or the end of semester, whichever comes first
  Remarks:
  This was a popular assessment in 2009, but many students lost easy marks because they failed to participate. The exercise has been scaled back this year to make room for the wiki-based assessment task

- **Assignment task 2**

Title:
Assignment 1: Specification
Description:
A proof-discharged event-B specification of a problem (exact problem to be advised)
Weighting:
20%
Due date:
end of week 5
Remarks:
Release date week 1

This assignment is administered in 2 parts. Part 1 is about defining the various parameters of the specification (in fact, a requirements analysis), and Part 2 is about coding the Event-B specification. The two parts will be offered separately, so that feedback may be returned on part 1, before students complete part 2.

- **Assignment task 3**

Title:
Assignment 2: Refinement
Description:
A proof-discharged event-B refinement of the specification begun in Assignment 1
Weighting:
20%
Due date:
end of week 11
Remarks:
Release date week 6

- **Assignment task 4**

Title:
User Manual wiki
Description:
The software used in this unit is fairly state-of-the-art, and few learning materials exist to support it. Like much of that software, it is poorly documented, there are no texts, the material it is based upon is unfamiliar to students, and the learning curve is very steep. So this exercise to write a user manual for the software, specifically designed for future students in the unit.

To do this, students will collaborate using the Moodle wiki feature. The exercise is to be collaborative, meaning that all students are expected to read and respond to other students inputs to the task, which will run over the length of the semester. The Moodle wiki will be used to monitor each student's contribution, and to determine a summative assessment. Feedback (formative assessment) will be offered through teacher interaction with the wiki.

One key concept underlying this exercise will be the need for students to reflect upon the learning experiences they themselves undergo, as they endeavour to understand the software itself, and the user interactions with it. One of the discussion exercises (task 1 above) will reinforce this reflective process.
Weighting:

10%

Due date:

end of semester

Remarks:

A new exercise for 2010

Examination

- **Weighting:** 40%
  Length: 2 hours
  Type (open/closed book): Closed book
  Remarks:

  Sample papers are available on the web pages

See Appendix for End of semester special consideration / deferred exams process.

Due dates and extensions

Please make every effort to submit work by the due dates. It is your responsibility to structure your study program around assignment deadlines, family, work and other commitments. Factors such as normal work pressures, vacations, etc. are not regarded as appropriate reasons for granting extensions. Students are advised to NOT assume that granting of an extension is a matter of course.

Students requesting an extension for any assessment during semester (eg. Assignments, tests or presentations) are required to submit a Special Consideration application form (in-semester exam/assessment task), along with original copies of supporting documentation, directly to their lecturer within two working days before the assessment submission deadline. Lecturers will provide specific outcomes directly to students via email within 2 working days. The lecturer reserves the right to refuse late applications.

A copy of the email or other written communication of an extension must be attached to the assignment submission.

Refer to the Faculty Special consideration webpage or further details and to access application forms: http://www.infotech.monash.edu.au/resources/student/equity/special-consideration.html

Late assignment

Assignments received after the due date will be subject to a penalty of 10% per day

Return dates

Students can expect assignments to be returned within two weeks of the submission date or after receipt, whichever is later.

Appendix

Please visit the following URL: http://www.infotech.monash.edu.au/units/appendix.html for further information about:

- Continuous improvement
- Unit evaluations
- Communication, participation and feedback
- Library access
- Monash University Studies Online (MUSO)
- Plagiarism, cheating and collusion
- Register of counselling about plagiarism
- Non-discriminatory language
- Students with disability
- End of semester special consideration / deferred exams