ANX-PR/CL/001-01

LEARNING GUIDE

SUBJECT

103000383 - Rigorous software development

DEGREE PROGRAMME

10AK - Master Universitario en Software y Sistemas

ACADEMIC YEAR & SEMESTER

2017/18 - Semester 1

Index

Learning guide

1. Description

1.1. Subjet details

Name of the subject	103000383 - Rigorous software development
No of credits	4 ECTS
Type	Optional
Academic year ot the programme	First year
Semester of tuition	Semester 1
Tuition period	September-January
Tuition languages	English
Degree programme	10AK - Master Universitario en Software y Sistemas
Centre	Escuela Tecnica Superior de Ingenieros Informaticos
Academic year	2017-18

2. Faculty

2.1. Faculty members with subject teaching role

Name and surname	Office/Room	Email	Tutoring hours *
Manuel Carro Li?ares (Subject coordinator)	2304	manuel.carro@upm.es	F - 15:00 - 19:00 Please send an e-mail to ensure an appointment before going to the instructor's office.
Julio Mari?o Carballo	2308	julio.marino@upm.es	Tu - 15:00 - 17:00 W - 12:30 - 13:30 Th - 15:00 - 17:00 F - 12:30 - 13:30 Please send an e-

			mail to ensure an appointment before going to the instructor's office.

* The tutoring schedule is indicative and subject to possible changes. Please check tutoring times with the faculty member in charge.

3. Prior knowledge recommended to take the subject

3.1. Recommended (passed) subjects

El plan de estudios Master Universitario en Software y Sistemas no tiene definidas asignaturas previas recomendadas para esta asignatura.

3.2. Other recommended learning outcomes

- First-order logic, formal proofs, declarative programming, reasoning about properties of algorithms.

4. Skills and learning outcomes *

4.1. Skills to be learned

CEM1 - Identificar, a partir del estado de la cuestión, la presencia de problemas de investigación relacionados con la concepción, la construcción, el uso y la evaluación de sistemas sociotécnicos complejos que hagan un uso intensivo de software

CEM5 - Aportar soluciones a aquellos problemas abiertos relacionados con el ámbito de aplicación y los métodos, técnicas y herramientas de Verificación y Validación de Software

CG12 - Comprensión amplia de las técnicas y métodos aplicables en una especialización concreta, así como de sus límites

CG13 - Apreciación de los límites del conocimiento actual y de la aplicación práctica de la tecnología más reciente.

CG14 - Conocimiento y comprensión de la informática necesaria para la creación de modelos de información, y de los sistemas y procesos complejos

CG4 - Que los estudiantes posean las habilidades de aprendizaje que les permitan continuar estudiando de un modo que habrá de ser en gran medida autodirigido o autónomo.

CG7 - Especificación y realización de tareas informáticas complejas, poco definidas o no familiares

CG8 - Planteamiento y resolución de problemas también en áreas nuevas y emergentes de su disciplina

CG9 - Aplicación de los métodos de resolución de problemas más recientes o innovadores y que puedan implicar el uso de otras disciplinas

CGI20 - Adquirir conocimientos científicos avanzados del campo de la informática que le permitan generar nuevas ideas dentro de una línea de investigación.

CGI23 - Capacidad de leer y comprender publicaciones dentro de su ámbito de estudio/investigación, así como su catalogación y valor científico

4.2. Learning outcomes

RA15 - Knowledge of techniques for proving code correctness

RA91 - Acquaintance with design requirements and implementation requirements.

RA92 - Acquaintance with various techniques for formal software development

RA93 - Knowledge of languages which ease the application of the aforementioned techniques.

RA94 - Effective use of rigorous software development techniques.

* The Learning Guides should reflect the Skills and Learning Outcomes in the same way as indicated in the Degree Verification Memory. For this reason, they have not been translated into English and appear in Spanish.

5. Brief description of the subject and syllabus

5.1. Brief description of the subject

Software is getting more and more complex and is becoming responsible for critical tasks. Therefore, any technology aimed at ensuring the reliability and quality of software will be increasingly relevant.

There are many ways to approach these goals. The *declarative* approach relies on languages and logics with a solid mathematical foundation. This includes specificacion languages (VDM, Z, B, Event-B, OBJ, Alloy, ...), functional programming languages (Haskell, Erlang, ?-calculi?), logic programming languages (Prolog, CLP, ASP,?) among others.

Some basic knowledge of logic and functional and logic programming is assumed as a prerequisite.

Some goals of the course are:

- To motivate the use of technologies in software development under the correctness-by-construction paradigm.
- To study different families of languages aimed at easing the process of building correct software.
- To understand the differences between declarative and procedural languages and the impact of these aspects in software development.
- To identify the better niches for the industrial application of declarative / correctness by construction technologies.

5.2. Syllabus

1. Introduction

    1.1. Overview, motivation, and challenges for rigorous SW development

    1.2. Review of background: formal logic, proofs...

2. Correctness by Construction

    2.1. Event-B: Theory and development methods.

    2.2. Event-B: the Rodin tool

3. Verification.

    3.1. Classical program verification

    3.2. The Dafny tool

    3.3. The Alloy tool

    3.4. Property-based testing

4. Specifications

    4.1. Algebraic specifications

    4.2. The Maude algebraic specification language

6. Schedule

6.1. Subject schedule*

Week	Face-to-face classroom activities	Face-to-face laboratory activities	Other face-to-face activities	Assessment activities
1	**Course introduction** Duration: 02:00 Lecture			
2	**Rigorous software development: a broad landscape** Duration: 01:45 Lecture			**Exercises, questions, and answers** Other assessment Continuous assessment Duration: 00:15
3	**Event-B** Duration: 01:45 Cooperative activities			**Exercises, questions, and answers** Other assessment Continuous assessment Duration: 00:15
4	**Event-B** Duration: 01:45 Cooperative activities			**Exercises, questions, and answers** Other assessment Continuous assessment Duration: 00:15
5	**Event-B** Duration: 01:45 Cooperative activities			**Exercises, questions, and answers** Other assessment Continuous assessment Duration: 00:15
6	**Event-B** Duration: 01:45 Cooperative activities			**Exercises, questions, and answers** Other assessment Continuous assessment Duration: 00:15
7	**Event-B** Duration: 01:45 Cooperative activities			**Exercises, questions, and answers** Other assessment Continuous assessment Duration: 00:15
8	**Program Verification** Duration: 01:45 Lecture			**Exercises, questions, and answers** Other assessment Continuous assessment Duration: 00:15
9	**Program Verification** Duration: 01:45 Lecture			**Exercises, questions, and answers** Other assessment Continuous assessment Duration: 00:15
10	**Program Verification** Duration: 01:45 Lecture			**Exercises, questions, and answers** Other assessment Continuous assessment Duration: 00:15
11	**Algebraic specifications** Duration: 01:45 Lecture			**Exercises, questions, and answers** Other assessment Continuous assessment Duration: 00:15

12	**Algebraic specifications** Duration: 01:45 Lecture			**Exercises, questions, and answers** Other assessment Continuous assessment Duration: 00:15
13	**Algebraic specifications** Duration: 01:45 Lecture			**Exercises, questions, and answers** Other assessment Continuous assessment Duration: 00:15
14	**Alloy** Duration: 01:45 Problem-solving class			**Exercises, questions, and answers** Other assessment Continuous assessment Duration: 00:15
15	**Exercises and problems** Duration: 02:00 Cooperative activities			
16				**Project presentation** Group work Final examination Duration: 02:00
17				**Project presentation** Group work Final examination Duration: 02:00

The independent study hours are training activities during which students should spend time on individual study or individual assignments.

Depending on the programme study plan, total values will be calculated according to the ECTS credit unit as 26/27 hours of student face-to-face contact and independent study time.

* The subject schedule is based on a previous theorical planning of the subject plan and might go to through experience some unexpected changes along throughout the academic year.

7. Activities and assessment criteria

7.1. Assessment activities

7.1.1. Continuous assessment

Week	Description	Modality	Type	Duration	Weight	Minimum grade	Evaluated skills
2	Exercises, questions, and answers	Other assessment	Face-to-face	00:15	7.69%	0 / 10	CG4 CG9 CG8 CEM1
3	Exercises, questions, and answers	Other assessment	Face-to-face	00:15	7.69%	0 / 10	CG4 CG9 CG8 CEM1
4	Exercises, questions, and answers	Other assessment	Face-to-face	00:15	7.69%	0 / 10	CG4 CG9 CG8 CEM1
5	Exercises, questions, and answers	Other assessment	Face-to-face	00:15	7.69%	0 / 10	CG4 CG9 CG8 CEM1
6	Exercises, questions, and answers	Other assessment	Face-to-face	00:15	7.69%	0 / 10	CG4 CG9 CG8 CEM1
7	Exercises, questions, and answers	Other assessment	Face-to-face	00:15	7.69%	0 / 10	CG4 CG9 CG8 CEM1
8	Exercises, questions, and answers	Other assessment	Face-to-face	00:15	7.69%	0 / 10	CG4 CG9 CG8 CEM1
9	Exercises, questions, and answers	Other assessment	Face-to-face	00:15	7.69%	0 / 10	CG4 CG9 CG8 CEM1

10	Exercises, questions, and answers	Other assessment	Face-to-face	00:15	7.69%	0 / 10	CG4 CG9 CG8 CEM1
11	Exercises, questions, and answers	Other assessment	Face-to-face	00:15	7.69%	0 / 10	CG4 CG9 CG8 CEM1
12	Exercises, questions, and answers	Other assessment	Face-to-face	00:15	7.69%	0 / 10	CG4 CG9 CG8 CEM1
13	Exercises, questions, and answers	Other assessment	Face-to-face	00:15	7.69%	0 / 10	CG4 CG9 CG8 CEM1
14	Exercises, questions, and answers	Other assessment	Face-to-face	00:15	7.69%	0 / 10	CG4 CG9 CG8 CEM1

7.1.2. Final examination

Week	Description	Modality	Type	Duration	Weight	Minimum grade	Evaluated skills
16	Project presentation	Group work	No Presential	02:00	50%	0 / 10	CG4 CG8 CG9 CEM1
17	Project presentation	Group work	No Presential	02:00	50%	0 / 10	CG4 CG8 CG9 CEM1

7.1.3. Referred (re-sit) examination

No se ha definido la evaluación extraordinaria.

7.2. Assessment criteria

Depending of the number of students, the final grade will be obtained either from:

- A suite of short, individual practical exercises periodically proposed which will be worth 50% of the final grade. The remaining 50% will come from short presentations.
- Individual practical exercises, if the number of students is too high to allow for the extra sessions needed for the presentations.

Exercises for each unit will have the same relative weight for the overall grade, although individual exercises in a given unit can have different weights.

The percentages don't add up to 100% due to rounding issues.

8. Teaching resources

8.1. Teaching resources for the subject

Name	Type	Notes
Event B development environment	Others	
Dafny	Others	
Maude	Others	
Alloy	Others	
Modeling in Event-B: System and Software Engineering. Jean-Raymond Abrial. Cambridge University Press.	Bibliography	
http://wiki.event-b.org/	Bibliography	

The Dafny web page at Microsoft RiSE: http://www.rise4fun.com/Dafny	Web resource	
All About Maude -- A High Performance Logical Framework. Clavel, M., Durán, F., Eker, S., Lincoln, P., Martí-Oliet, N., Meseguer, J., Talcott, C. Lecture Notes in Computer Science, vol. 4350.	Bibliography	
Alloy: A Lightweight Object Modelling Notation. Daniel Jackson. ACM Transactions on Software Engineering and Methodology (TOSEM'02), volume 11, issue 2, pages 256-290.	Bibliography	
Seven Myths of Formal Methods. Anthony Hall. IEEE Software, September 1990	Bibliography	
Seven More Myths of Formal Methods. Jonathan P. Bowen, Michael G. Hinchey. IEEE Software, July 1995.	Bibliography	
First Steps in the Verified Software Grand Challenge. Cliff Jones, Peter O'Hearn, Jim Woodcock. IEEE Computer, April 2006.	Bibliography	