

<b>MODULE TITLE</b>	Software Development		<b>CREDIT VALUE</b>	15
<b>MODULE CODE</b>	ECM2414		<b>MODULE CONVENER</b>	Dr Majeed Soufian (Coordinator)
<b>DURATION: TERM</b>	1	2	3	
<b>DURATION: WEEKS</b>	11	0	0	
<b>Number of Students Taking Module (anticipated)</b>	91			

#### DESCRIPTION - summary of the module content

The module will introduce you to software design and development concepts and methods, alongside intermediate and advanced constructs and concepts in the Java programming language, and the programming paradigms these relate to. This includes generic programming (and Java generics), concurrent programming (via Java threads), design patterns, networked programs and nested inner classes. We will also cover widespread tools in software development, including version control and unit testing.

Prerequisite module: ECM1410 or equivalent.

#### AIMS - intentions of the module

This module will introduce you to methods for the rigorous testing and assessment of software, and prepare you for complex programming tasks in a specific object-oriented programming language, including advanced concepts and syntax, and the use of multiple programs in parallel.

#### INTENDED LEARNING OUTCOMES (ILOs) (see assessment section below for how ILOs will be assessed)

On successful completion of this module, **you should be able to:**

##### Module Specific Skills and Knowledge:

- 1 use a software design and development method which incorporates both formal and informal techniques appropriately;
- 2 design and implement rigorous testing frameworks for software, and be aware of the inherent limitations of tests developed;
- 3 deploy advanced object-oriented language concepts and techniques;
- 4 develop multi-program software systems.

##### Discipline Specific Skills and Knowledge:

- 5 follow the phases of software development;
- 6 recognise and evaluate different development practices, and judge their appropriateness for a specific development problem.

##### Personal and Key Transferable / Employment Skills and Knowledge:

- 7 analyse and break down a problem into constituent parts;
- 8 compare and critically contrast different potential solutions to a problem;
- 9 follow the pair-programming development approach used widely in industry.

#### SYLLABUS PLAN - summary of the structure and academic content of the module

The module syllabus is based on the four themes below, each individual topic listed being related to at least one of these themes:

- software design: design patterns (builder pattern, singleton pattern, abstract factory method pattern, decorator pattern);
- software development and testing: fundamentals of software development and testing, V-model, unit testing, code coverage, corner cases, unit tests and the JUnit framework, mock objects and test suites, sophisticated integrated development environments (IDEs), version control systems;
- doing multiple things at once: concurrency, multi-tasking and threading, race conditions, atomic data, locks, deadlock, starvation, livelock, synchronisation, thread notification, data freshness, instance confinement, object publication and escape, safe construction, thread confinement, thread management, thread states, sockets, remote method invocation;
- programming styles and advanced Java syntax: generic programming, Java generics, event-driven programming, events and listeners, reflection in Java, Java collections framework.

#### LEARNING AND TEACHING

##### LEARNING ACTIVITIES AND TEACHING METHODS (given in hours of study time)

<b>Scheduled Learning &amp; Teaching Activities</b>	52.00	<b>Guided Independent Study</b>	98.00	<b>Placement / Study Abroad</b>	0.00
---	-------	---------------------------------	-------	---------------------------------	------

##### DETAILS OF LEARNING ACTIVITIES AND TEACHING METHODS

Category	Hours of study time	Description
Scheduled learning and teaching activities	22	Lectures
Scheduled learning and teaching activities	20	Workshops/tutorials
Scheduled learning and teaching activities	10	Surgeries
Guided independent study	20	Individual assessed work
Guided independent study	78	Wider reading and exam preparation

#### ASSESSMENT

##### FORMATIVE ASSESSMENT - for feedback and development purposes; does not count towards module grade

Form of Assessment	Size of Assessment (e.g. duration/length)	ILOs Assessed	Feedback Method
--------------------	---	---------------	-----------------

Form of Assessment	Size of Assessment (e.g. duration/length)	ILOs Assessed	Feedback Method
Online test	1 hour	1, 3, 5, 6, 7, 8	Oral, and in model answer review in surgeries

#### SUMMATIVE ASSESSMENT (% of credit)

Coursework	40	Written Exams	60	Practical Exams	0
------------	----	---------------	----	-----------------	---

#### DETAILS OF SUMMATIVE ASSESSMENT

Form of Assessment	% of Credit	Size of Assessment (e.g. duration/length)	ILOs Assessed	Feedback Method
Written exam	60	1-hour January Exam	All	Oral, on request
Coursework - practical programming assignment	40	40 hours	All	Written feedback submitted code/assignment and/or on feedback sheet. Oral feedback in surgeries

#### DETAILS OF RE-ASSESSMENT (where required by referral or deferral)

Original Form of Assessment	Form of Re-assessment	ILOs Re-assessed	Time Scale for Re-reassessment
Written exam	Written exam (1 hour)	All	August Ref/Def period
Coursework - practical programming assignment	Coursework - practical programming assignment	All	August Red/Def period

#### RE-ASSESSMENT NOTES

Reassessment will be by coursework and/or written exam in the failed or deferred element only. For referred candidates, the module mark will be capped at 40%. For deferred candidates, the module mark will be uncapped.

### RESOURCES

**INDICATIVE LEARNING RESOURCES - The following list is offered as an indication of the type & level of information that you are expected to consult. Further guidance will be provided by the Module Convener**

**ELE:** College to provide hyperlink to appropriate pages

#### Reading list for this module:

Type	Author	Title	Edition	Publisher	Year	ISBN	Search
Set	Bloch, Joshua	Effective Java: Programming Language Guide	2nd	Addison Wesley	2008	978-0321356680	<a href="#">[Library]</a>
Set	Darwin, Ian	Java Cookbook	2nd	O'Reilly	2004	978-0596007010	<a href="#">[Library]</a>
Set	Downey, Adam	How to think like a computer scientist	4th	http://www.greentreepress.com/thinkapjava/2004			<a href="#">[Library]</a>
Set	Freeman, E & E, Sierra, K, Bates, B	Head First Design Patterns		O'Reilly Media	2004	978-0596007126	<a href="#">[Library]</a>
Set	Langr, Jeff, Hunt, Andy & Thomas, Dave	Pragmatic Unit Testing in Java 8 with JUnit		Pragmatic Bookshelf	2015	9781941222591	<a href="#">[Library]</a>
Set	Niemeyer, Patrick and Knudsen, Jonathan	Learning Java	3rd	O'Reilly	2005		<a href="#">[Library]</a>

<b>CREDIT VALUE</b>	15	<b>ECTS VALUE</b>	7.5
---------------------	----	-------------------	-----

<b>PRE-REQUISITE MODULES</b>	ECM1410
------------------------------	---------

<b>CO-REQUISITE MODULES</b>	
-----------------------------	--

<b>NQF LEVEL (FHEQ)</b>	2 (NQF level 5)	<b>AVAILABLE AS DISTANCE LEARNING</b>	No
<b>ORIGIN DATE</b>	Tuesday 10 July 2018	<b>LAST REVISION DATE</b>	Wednesday 08 February 2023
<b>KEY WORDS SEARCH</b>	Software development; Java; concurrent programs; design patterns; software testing; networked programs.		