

Course outline for

Best Programming Practices in Java

BCPR282

Semester One, 2019

Introduction - Kōrero whakatuwhera

This outline contains important information about the delivery and assessment of this course. Read it carefully and if there is anything you do not understand please ensure you ask a staff member listed below for clarification.

Please refer to your **programme handbook** for all programme related information, for example programme structure and regulations, grade scale and assessment regulations.

Academic staff - Kā pouako

The following staff are directly involved with the delivery of this course:

Name	Role	Phone	Office	Office hours	Email address
Dr Luofeng Xu	Tutor	940 8394	N220	Friday PM in X205	Luofeng.Xu@ara.ac.nz

Timetable - Wātaka

For timetable information for this course please refer to:

- Tribal – through the student portal; or
- Moodle – look in Department of Computing Student Info > Topic 6 Timetables; or
- Noticeboards – level 2 of N-block or C-block

Course descriptor - Whakamāramataka

<i>Course Code</i>	BCPR282	<i>Contact Hours</i>	12
<i>Credits</i>	15	<i>Other Directed Hours</i>	12
<i>Level</i>	6	<i>Total Supervised Hours</i>	24
<i>Unit Standard</i>	-	<i>Self Directed Hours</i>	126
		<i>Total Learning Hours</i>	150
<i>Pre-requisites</i>	BCPR280		
<i>Restrictions</i>	BCPR212, BCPR222		

Aim

- To develop student competence in a programming language.
- To provide students with the knowledge, understanding and skills necessary to both develop and maintain commercially relevant programs, and follow them through to implementation.
- To have the student finish a sizeable software project and create a portfolio containing examples of their work.
- To compare Java with another programming language. (normally the one studied in BCSE102)

Learning Outcomes

On completion the student will be able to:

1. Describe and use common Design Patterns, Algorithms and programming language Idioms.
2. Use appropriate software life-cycle models and software construction steps.
3. Design programs.
4. Design user interfaces which conform to recognised usability criteria.
5. Code programs in the specified language using the prescribed standards.
6. Produce and execute testing strategies at the systems level using a unit testing framework.
7. Debug and test programs to the systems level.
8. Provide all appropriate systems level documentation.
9. Maintain existing programs and update documentation.
10. Compare and contrast the features and uses of different programming languages.

Assessment

No	Assessment Type	Pass Criteria	Weighting	Outcomes Assessed
1	Foundation Test	50%	25%	1, 3, 7, 10
2	Weekly Status Report		10%	1, 3, 7
3	Project Presentation Mid-Course		15%	1-9
4	Project Presentation Final	50%	50%	1-9

To pass this course, students must gain an average of at least 50% across all assessments, and gain at least 50% in the Foundation Test and the Project Presentation Final.

Learning and Teaching Strategies

Lectures, PowerPoint presentations, practical workshop laboratories, and individual assistance as required in the workshops

NZQA Level Descriptors

The following descriptors outline what is expected of students studying a course at the specified level.

	Level 4	Level 5	Level 6	Level 7
Knowledge	Broad operational and theoretical knowledge in a field of work or study	Broad operational or technical and theoretical knowledge within a specific field of work or study	Specialised technical or theoretical knowledge with depth in a field of work or study	Specialised technical or theoretical knowledge with depth in one or more fields of work or study
Skills	Select and apply solutions to familiar and sometimes unfamiliar problems Select and apply a range of standard and non-standard processes relevant to the field of work or study	Select and apply a range of solutions to familiar and sometimes unfamiliar problems Select and apply a range of standard and non-standard processes relevant to the field of work or study	Analyse and generate solutions to familiar and unfamiliar problems Select and apply a range of standard and non-standard processes relevant to the field of work or study	Analyse, generate solutions to unfamiliar and sometimes complex problems Select, adapt and apply a range of processes relevant to the field of work or study
Application [of knowledge and skills]	Self-management of learning and performance under broad guidance Some responsibility for performance of others	Complete self-management of learning and performance within defined contexts Some responsibility for the management of learning and performance of others	Complete self-management of learning and performance within dynamic contexts Responsibility for leadership within dynamic contexts	Advanced generic skills and/or specialist knowledge and skills in a professional context or field of study

Assessments - Kā Aromatawai

Assessment	Brief	Week of	Weighting
Foundation Test	Based on Java Syntax Exercises	Week 7, 8 April in class	25%
Weekly Status Report	2% tests in class based on the 5 syntax exercises "homework" per week.	Week 2, 4 March – Week 6, 1 April	10%
Project Presentation Mid-Course	Individual portfolios of work showing that <<Model>> functionality has been implemented and unit tested.	9am, 29 April (Week 8)	15%
Project Presentation Final	Individual portfolios of work showing that Android <<View>> functionality has been implemented.	9am, 1 July (Week 17)	50%

Assessment tasks - Kā tūmahi aromatawai

Teaching staff will provide you with specific details of what is required for each assessment in advance of the due date. This information may be uploaded to the appropriate course area in Moodle or be given to you in the form of a handout. Staff may also provide additional information, advice and tips regarding assessments during timetabled class sessions, so you are encouraged to attend class regularly.

Assessment criteria / Marking schedule - Kā paearu

Nearer the time of each assessment, teaching staff will provide you with information on the assessment criteria that will be applied and/or how marks will be awarded.

This will be available in the BCPR282 course area on Moodle.

Course schedule - Maramataka

Week	Commencing	Content
1	25 February	Java core syntax, architectural issues
2	4 March	Java class design 2% Assessment #2 Weekly Status Report in-class test #1
3	11 March	Java data types, assertions 2% Assessment #2 Weekly Status Report in-class test #2
4	18 March	Collections, exceptions 2% Assessment #2 Weekly Status Report in-class test #3
5	25 March	Testing framework 2% Assessment #2 Weekly Status Report in-class test #4
6	01 April	Assignment support 2% Assessment #2 Weekly Status Report in-class test #5
7	8 April	25% Assessment #1 Foundation Test in-class exam
Term break – Monday 15 April – Friday 26 April (Includes Easter Holiday 19 - 22 April and Anzac Day Thursday 25 April)		
8	29 April	Android 15% Assessment #3 Project Presentation Mid-Course due on Mon 23 April at 9am
9	06 May	Android
10	13 May	Android
11	20 May	Android
12	27 May	Android
No Classes Monday 3 June – Queen's Birthday Holiday		
13	03 June	Public holiday – no class
14	10 June	Assignment support
15	17 June	Study Week
16	24 June	Exam week
17	01 July	Exam Week 50% Assessment #4 Project Presentation Final due on Monday 25 June at 9am

Note: Students will be notified in advance if there are any changes to the course schedule.