



COMP2000

Object-Oriented Programming Practices

Session 2, In person-scheduled-weekday, North Ryde 2024

School of Computing

Contents

<u>General Information</u>	2
<u>Learning Outcomes</u>	2
<u>General Assessment Information</u>	3
<u>Assessment Tasks</u>	4
<u>Delivery and Resources</u>	5
<u>Policies and Procedures</u>	7
<u>Changes from Previous Offering</u>	8

Disclaimer

Macquarie University has taken all reasonable measures to ensure the information in this publication is accurate and up-to-date. However, the information may change or become out-dated as a result of change in University policies, procedures or rules. The University reserves the right to make changes to any information in this publication without notice. Users of this publication are advised to check the website version of this publication [or the relevant faculty or department] before acting on any information in this publication.

General Information

Unit convenor and teaching staff
Convener, Lecturer, Pattern Matcher
Matthew Roberts
matthew.roberts@mq.edu.au

Lecturer, Programming Polyglot
Damian Jurd
damian.jurd@mq.edu.au

Credit points
10

Prerequisites
COMP1010 or COMP125

Corequisites

Co-badged status

Unit description
Object-oriented programming is a key technology for modern computing. This unit bridges the gap between introductory programming and larger multi-person projects by considering the use of object-oriented techniques to produce intermediate sized software. Practical exercises emphasise the importance of programming practices such as appropriate documentation, systematic approaches to debugging and testing, and the use of software development tools. The unit is taught using Java.

Important Academic Dates

Information about important academic dates including deadlines for withdrawing from units are available at <https://www.mq.edu.au/study/calendar-of-dates>

Learning Outcomes

On successful completion of this unit, you will be able to:

- ULO1:** explain the key concepts of object oriented programming, and program proficiently in an OO programming language
- ULO2:** apply the concepts underlying software design and a working knowledge of a selection of well known design patterns
- ULO3:** demonstrate good programming practices such as testing, debugging,

documentation, version control, programming tools and interactive development environments

ULO4: apply key object oriented concepts and libraries to design and develop applications of significant complexity

ULO5: apply key concepts of concurrency theoretically and in working code

General Assessment Information

Requirements to Pass this Unit

To pass this unit you must:

1. Participate in all workshop classes. You will not be able to access exams unless you have participated in workshop classes or have special consideration for the classes.
1. Achieve a total mark equal to or greater than 50%.

Late Assessment Submission Penalty

Unless a Special Consideration request has been submitted and approved, a 5% penalty (of the total possible mark of the task) will be applied for each day a written report or presentation assessment is not submitted, up until the 7th day (including weekends). After the 7th day, a grade of '0' will be awarded even if the assessment is submitted. The submission time for all uploaded assessments is **11:55 pm**. A 1-hour grace period will be provided to students who experience a technical concern. For any late submission of time-sensitive tasks, such as scheduled tests/exams, performance assessments/presentations, and/or scheduled practical assessments/labs, please apply for [Special Consideration](#). For example, if the assignment is worth 8 marks (of the entire unit) and your submission is late by 19 hours (or 23 hours 59 minutes 59 seconds), 0.4 marks (5% of 8 marks) will be deducted. If your submission is late by 24 hours (or 47 hours 59 minutes 59 seconds), 0.8 marks (10% of 8 marks) will be deducted, and so on.

Assessments where Late Submissions will be accepted

- Major Creative Work – YES, Standard Late Penalty applies
- Examinations - NO, unless Special Consideration is Granted

Special Consideration

The [Special Consideration Policy](#) aims to support students who have been impacted by short-term circumstances or events that are serious, unavoidable and significantly disruptive, and which may affect their performance in assessment. If you experience circumstances or events that affect your ability to complete the assessments in this unit on time, please inform the convenor and submit a Special Consideration request through ask.mq.edu.au.

Assessment Tasks

Name	Weighting	Hurdle	Due
Major Creative Work	60%	No	In-class week 4, 22nd Sep, 27th Oct 11:55pm
Module Exams	40%	No	Choice of times weeks 7, 13, in class week 7 and exam period

Major Creative Work

Assessment Type [1](#): Programming Task

Indicative Time on Task [2](#): 40 hours

Due: **In-class week 4, 22nd Sep, 27th Oct 11:55pm**

Weighting: **60%**

A semester-long programming task where students put all their skills to work creating a medium sized application.

On successful completion you will be able to:

- explain the key concepts of object oriented programming, and program proficiently in an OO programming language
- apply the concepts underlying software design and a working knowledge of a selection of well known design patterns
- demonstrate good programming practices such as testing, debugging, documentation, version control, programming tools and interactive development environments
- apply key object oriented concepts and libraries to design and develop applications of significant complexity
- apply key concepts of concurrency theoretically and in working code

Module Exams

Assessment Type [1](#): Examination

Indicative Time on Task [2](#): 16 hours

Due: **Choice of times weeks 7, 13, in class week 7 and exam period**

Weighting: **40%**

The module examinations ask students to answer conceptual questions about the course

material as well as solving programming problems.

On successful completion you will be able to:

- explain the key concepts of object oriented programming, and program proficiently in an OO programming language
- apply the concepts underlying software design and a working knowledge of a selection of well known design patterns
- demonstrate good programming practices such as testing, debugging, documentation, version control, programming tools and interactive development environments
- apply key object oriented concepts and libraries to design and develop applications of significant complexity
- apply key concepts of concurrency theoretically and in working code

¹ If you need help with your assignment, please contact:

- the academic teaching staff in your unit for guidance in understanding or completing this type of assessment
- the [Writing Centre](#) for academic skills support.

² Indicative time-on-task is an estimate of the time required for completion of the assessment task and is subject to individual variation

Delivery and Resources

Classes begin in week 1 for both lectures and workshops.

CLASSES

Each week of COMP2000 has a two hour lecture and a two-hour practical class. For details of days, times and rooms, consult the University timetables webpage (<http://www.timetables.mq.edu.au>). Practical classes commence in Week 1 and are held in the 4RPD Computer Laboratories computer laboratories for on-campus classes and in zoom rooms for online classes (links published in iLearn).

In all cases students are expected to do significant preparatory work, readings and exercises *before* attending classes.

REQUIRED AND RECOMMENDED TEXTS AND/OR MATERIALS

The required texts for the unit is (available online via the Macquarie University Library, see below):

- Head First Design Patterns by Eric T Freeman, Elisabeth Robson, Bert Bates and Kathy

Sierra, O'Reilly Media, October 2004 (ISBN:978-0-596-00712-6)

- Modern Java in Action by Raoul-Gabriel Urma, Mario Fusco, Alan Mycroft

There will be no lecture notes provided, all examinable material is given in course readings and the textbook. Students are required to study this material and answer preparatory questions *before* class.

The Macquarie library contains many books on object-oriented programming in general, and on programming specifically in Java, that you may want to use to supplement the text and lecture notes.

One particularly useful service that the library provides you with is access to many Java related titles online via the Safari Books Online (<http://proquest.safaribooksonline.com/>) service. Using this service, which you can only access from a machine connected to the University network, you might like to have a look at the following Java titles:

1. Head First Design Patterns by Eric T Freeman, Elisabeth Robson, Bert Bates and Kathy Sierra, O'Reilly Media, October 2004 (ISBN:978-0-596-00712-6)
2. Learning Java, 3rd Edition by Jonathan Knudsen; Patrick Niemeyer, ISBN: 978-0-596-00873-4
3. Java in a Nutshell, 5th Edition by David Flanagan, ISBN: 978-0-596-00773-7
4. Java Examples in a Nutshell, 3rd Edition by David Flanagan, ISBN: 978-0-596-00620-4
The web itself is an ideal source of Java information, and from time to time we will be posting useful links on the COMP2000 iLearn site. Two particularly useful resources are:
5. Thinking in Java by Bruce Eckel, a free version of the 3rd edition of this pretty comprehensive book is available for download from <http://www.mindview.net/Books/TIJ/> and its 4th edition, which is updated for use with Java 5 and 6, is available for \$25 from <http://mindview.net/Books/TIJ4>.
6. The official Java Tutorial <http://download.oracle.com/javase/tutorial/> which is a comprehensive resource providing trails covering topics ranging from the basics of Java programming to more advanced subjects like GUI development, Generics, Class Reflection, Sound, Graphics, Network Programming and Concurrency

Methods of Communication

We will communicate with you via your university email and through announcements on iLearn. Queries to convenors can either be placed on the iLearn discussion board or sent to the unit convenor via the contact email on iLearn.

COVID Information

For the latest information on the University's response to COVID-19, please refer to the Coronavirus infection page on the Macquarie website: <https://www.mq.edu.au/about/coronavirus-faqs>. Remember to check this page regularly in case the information and requirements change

during semester. If there are any changes to this unit in relation to COVID, these will be communicated via iLearn.

Policies and Procedures

Macquarie University policies and procedures are accessible from [Policy Central \(https://policies.mq.edu.au\)](https://policies.mq.edu.au). Students should be aware of the following policies in particular with regard to Learning and Teaching:

- [Academic Appeals Policy](#)
- [Academic Integrity Policy](#)
- [Academic Progression Policy](#)
- [Assessment Policy](#)
- [Fitness to Practice Procedure](#)
- [Assessment Procedure](#)
- [Complaints Resolution Procedure for Students and Members of the Public](#)
- [Special Consideration Policy](#)

Students seeking more policy resources can visit [Student Policies \(https://students.mq.edu.au/support/study/policies\)](https://students.mq.edu.au/support/study/policies). It is your one-stop-shop for the key policies you need to know about throughout your undergraduate student journey.

To find other policies relating to Teaching and Learning, visit [Policy Central \(https://policies.mq.edu.au\)](https://policies.mq.edu.au) and use the [search tool](#).

Student Code of Conduct

Macquarie University students have a responsibility to be familiar with the Student Code of Conduct: <https://students.mq.edu.au/admin/other-resources/student-conduct>

Results

Results published on platform other than [eStudent](#), (eg. iLearn, Coursera etc.) or released directly by your Unit Convenor, are not confirmed as they are subject to final approval by the University. Once approved, final results will be sent to your student email address and will be made available in [eStudent](#). For more information visit ask.mq.edu.au or if you are a Global MBA student contact globalmba.support@mq.edu.au

Academic Integrity

At Macquarie, we believe [academic integrity](#) – honesty, respect, trust, responsibility, fairness and courage – is at the core of learning, teaching and research. We recognise that meeting the expectations required to complete your assessments can be challenging. So, we offer you a range of resources and services to help you reach your potential, including free [online writing and maths support](#), [academic skills development](#) and [wellbeing consultations](#).

Student Support

Macquarie University provides a range of support services for students. For details, visit <http://stu>

dents.mq.edu.au/support/

The Writing Centre

The [Writing Centre](#) provides resources to develop your English language proficiency, academic writing, and communication skills.

- [Workshops](#)
- [Chat with a WriteWISE peer writing leader](#)
- [Access StudyWISE](#)
- [Upload an assignment to Studiosity](#)
- [Complete the Academic Integrity Module](#)

The Library provides online and face to face support to help you find and use relevant information resources.

- [Subject and Research Guides](#)
- [Ask a Librarian](#)

Student Services and Support

Macquarie University offers a range of [Student Support Services](#) including:

- [IT Support](#)
- [Accessibility and disability support](#) with study
- Mental health [support](#)
- [Safety support](#) to respond to bullying, harassment, sexual harassment and sexual assault
- [Social support including information about finances, tenancy and legal issues](#)
- [Student Advocacy](#) provides independent advice on MQ policies, procedures, and processes

Student Enquiries

Got a question? Ask us via [AskMQ](#), or contact [Service Connect](#).

IT Help

For help with University computer systems and technology, visit http://www.mq.edu.au/about_us/offices_and_units/information_technology/help/.

When using the University's IT, you must adhere to the [Acceptable Use of IT Resources Policy](#). The policy applies to all who connect to the MQ network including students.

Changes from Previous Offering

This offering will make use of a new textbook for the last 5 weeks of the course.

Unit information based on version 2024.02 of the [Handbook](#)