



# COIT11134 Object Oriented Programming

## Term 1 - 2024

Profile information current as at 29/07/2024 03:15 pm

All details in this unit profile for COIT11134 have been officially approved by CQUniversity and represent a learning partnership between the University and you (our student). The information will not be changed unless absolutely necessary and any change will be clearly indicated by an approved correction included in the profile.

### General Information

#### Overview

This unit introduces object-oriented programming principles extensively. You will learn to develop interactive software applications using a modern programming language, integrated development environment (IDE), and graphical user interface (GUI) components. You will learn how to use classes, objects, methods, inheritance, polymorphism, foundation data structures and exception handling in your application development. In addition, you will learn how to apply these skills in solving practical problems.

#### Details

Career Level: *Undergraduate*

Unit Level: *Level 1*

Credit Points: 6

Student Contribution Band: 8

Fraction of Full-Time Student Load: 0.125

#### Pre-requisites or Co-requisites

Prerequisite unit COIT11222.

Important note: Students enrolled in a subsequent unit who failed their pre-requisite unit, should drop the subsequent unit before the census date or within 10 working days of Fail grade notification. Students who do not drop the unit in this timeframe cannot later drop the unit without academic and financial liability. See details in the [Assessment Policy and Procedure \(Higher Education Coursework\)](#).

#### Offerings For Term 1 - 2024

- Brisbane
- Cairns
- Melbourne
- Online
- Rockhampton
- Sydney

#### Attendance Requirements

All on-campus students are expected to attend scheduled classes – in some units, these classes are identified as a mandatory (pass/fail) component and attendance is compulsory. International students, on a student visa, must maintain a full time study load and meet both attendance and academic progress requirements in each study period (satisfactory attendance for International students is defined as maintaining at least an 80% attendance record).

#### Website

[This unit has a website, within the Moodle system, which is available two weeks before the start of term. It is important that you visit your Moodle site throughout the term. Please visit Moodle for more information.](#)

## Class and Assessment Overview

### Recommended Student Time Commitment

Each 6-credit Undergraduate unit at CQUniversity requires an overall time commitment of an average of 12.5 hours of study per week, making a total of 150 hours for the unit.

### Class Timetable

#### [Regional Campuses](#)

Bundaberg, Cairns, Emerald, Gladstone, Mackay, Rockhampton, Townsville

#### [Metropolitan Campuses](#)

Adelaide, Brisbane, Melbourne, Perth, Sydney

### Assessment Overview

#### 1. **Practical and Written Assessment**

Weighting: 20%

#### 2. **Practical and Written Assessment**

Weighting: 30%

#### 3. **Practical Assessment**

Weighting: 20%

#### 4. **Case Study**

Weighting: 30%

### Assessment Grading

This is a graded unit: your overall grade will be calculated from the marks or grades for each assessment task, based on the relative weightings shown in the table above. You must obtain an overall mark for the unit of at least 50%, or an overall grade of 'pass' in order to pass the unit. If any 'pass/fail' tasks are shown in the table above they must also be completed successfully ('pass' grade). You must also meet any minimum mark requirements specified for a particular assessment task, as detailed in the 'assessment task' section (note that in some instances, the minimum mark for a task may be greater than 50%). Consult the [University's Grades and Results Policy](#) for more details of interim results and final grades.

## CQUniversity Policies

**All University policies are available on the [CQUniversity Policy site](#).**

You may wish to view these policies:

- Grades and Results Policy
- Assessment Policy and Procedure (Higher Education Coursework)
- Review of Grade Procedure
- Student Academic Integrity Policy and Procedure
- Monitoring Academic Progress (MAP) Policy and Procedure – Domestic Students
- Monitoring Academic Progress (MAP) Policy and Procedure – International Students
- Student Refund and Credit Balance Policy and Procedure
- Student Feedback – Compliments and Complaints Policy and Procedure
- Information and Communications Technology Acceptable Use Policy and Procedure

This list is not an exhaustive list of all University policies. The full list of University policies are available on the [CQUniversity Policy site](#).

## Previous Student Feedback

### Feedback, Recommendations and Responses

Every unit is reviewed for enhancement each year. At the most recent review, the following staff and student feedback items were identified and recommendations were made.

#### Feedback from Course and Discipline Review

**Feedback**

The unit needs to be updated to suit the new pre-requisites.

**Recommendation**

An Update Unit Proposal is recommended to update the textbook, unit overview, unit learning outcomes, and assessments.

#### Feedback from Course and Discipline Review

**Feedback**

Weekly materials need to be redeveloped to suit the updated unit.

**Recommendation**

Develop weekly lecture slides, tutorials lab exercises and assessments.

#### Feedback from Discipline Reflection

**Feedback**

Improve student participation.

**Recommendation**

Add weekly lab projects and exercises into assessments.

## Unit Learning Outcomes

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

1. Explain the principles of object-oriented programming
2. Implement and test object-oriented programs using a modern programming language
3. Build interactive software applications using Graphical User Interface components
4. Apply the concept of exception handling and file data manipulation in object-oriented code
5. Apply appropriate data structure in object-oriented design and development.

The Australian Computer Society (ACS) recognises the Skills Framework for the Information Age (SFIA). SFIA is adopted by organisations, governments and individuals in many countries and provides a widely used and consistent definition of ICT skills. SFIA is increasingly being used when developing job descriptions and role profiles. ACS members can use the tool [MySFIA](#) to build a skills profile.

This unit contributes to the following workplace skills as defined by [SFIA 8](#). The SFIA code is included:

- Programming/Software development (PROG)
- Software Design (SWDN)
- Data Modelling and Design(DTAN)
- Testing (TEST)
- System Integration and Build (SINT)
- User Experience Design (HCEV)

## Alignment of Learning Outcomes, Assessment and Graduate Attributes



### Alignment of Assessment Tasks to Learning Outcomes

Assessment Tasks	Learning Outcomes				
	1	2	3	4	5
1 - Practical and Written Assessment - 20%	•	•			
2 - Practical and Written Assessment - 30%		•	•	•	
3 - Practical Assessment - 20%		•	•	•	•
4 - Case Study - 30%	•				•

### Alignment of Graduate Attributes to Learning Outcomes

Graduate Attributes	Learning Outcomes				
	1	2	3	4	5
1 - Communication	•	•	•		•
2 - Problem Solving		•	•	•	•
3 - Critical Thinking				•	•
4 - Information Literacy	•				•
5 - Team Work					
6 - Information Technology Competence		•	•	•	•
7 - Cross Cultural Competence					
8 - Ethical practice					
9 - Social Innovation					
10 - Aboriginal and Torres Strait Islander Cultures					

## Textbooks and Resources

### Textbooks

COIT11134

#### Prescribed

##### Java Programming - Tenth Edition

Edition: 10ed (2023)

Authors: Joyce Farrell

Cengage

ISBN: 978-0-357-67342-3

Binding: eBook

#### Additional Textbook Information

The link for purchasing the eTextbook will be provided on the unit website. Students will receive a discount code for the eTextbook.

### IT Resources

#### You will need access to the following IT resources:

- CQUniversity Student Email
- Internet
- Unit Website (Moodle)
- Apache NetBeans 20.0
- OpenJDK 21 LTS
- SceneBuilders 21

## Referencing Style

All submissions for this unit must use the referencing style: [Harvard \(author-date\)](#)

For further information, see the Assessment Tasks.

## Teaching Contacts

**Lily Li** Unit Coordinator

[l.li@cqu.edu.au](mailto:l.li@cqu.edu.au)

## Schedule

### Week 1 - 04 Mar 2024

Module/Topic	Chapter	Events and Submissions/Topic
Creating Java programs and Setting up Java development environment	Chapter 1 (prescribed textbook)	Download and setup software: OpenJDK 21 LTS and NetBeans 20

### Week 2 - 11 Mar 2024

Module/Topic	Chapter	Events and Submissions/Topic
Using data and methods	Chapter 2 and Chapter 3 (prescribed textbook)	

### Week 3 - 18 Mar 2024

Module/Topic	Chapter	Events and Submissions/Topic
Using classes and objects	Chapter 4 (prescribed textbook)	

### Week 4 - 25 Mar 2024

Module/Topic	Chapter	Events and Submissions/Topic
Decisions and looping	Chapter 5 and Chapter 6 (prescribed textbook)	
<b>Week 5 - 01 Apr 2024</b>		
Module/Topic	Chapter	Events and Submissions/Topic
Characters and Strings	Chapter 7 (prescribed textbook)	
<b>Vacation Week - 08 Apr 2024</b>		
Module/Topic	Chapter	Events and Submissions/Topic
<b>Week 6 - 15 Apr 2024</b>		
Module/Topic	Chapter	Events and Submissions/Topic
Inheritance and Interface	Chapter 9 (prescribed textbook)	<b>Assignment 1 - Java Console Application</b> Due: Week 6 Friday (19 Apr 2024) 11:59 pm AEST
<b>Week 7 - 22 Apr 2024</b>		
Module/Topic	Chapter	Events and Submissions/Topic
Java GUI development using JavaFX Event handling	Appendix F (prescribed textbook) CQU resources	Install and configure SceneBuilder to NetBeans
<b>Week 8 - 29 Apr 2024</b>		
Module/Topic	Chapter	Events and Submissions/Topic
Array and ArrayList	Chapter 8 and Chapter 13 (partially) (prescribed textbook)	
<b>Week 9 - 06 May 2024</b>		
Module/Topic	Chapter	Events and Submissions/Topic
Exception handling, file I/O	Chapter 10 and Chapter 11 (prescribed textbook)	
<b>Week 10 - 13 May 2024</b>		
Module/Topic	Chapter	Events and Submissions/Topic
Java Collection, generic methods and generic classes	Chapter 13 (prescribed textbook)	<b>Assignment 2 - JAVA GUI application</b> Due: Week 10 Friday (17 May 2024) 11:59 pm AEST
<b>Week 11 - 20 May 2024</b>		
Module/Topic	Chapter	Events and Submissions/Topic
Introduction to software engineering	CQU online resources	
<b>Week 12 - 27 May 2024</b>		
Module/Topic	Chapter	Events and Submissions/Topic
Case study and revision	CQU online resources	
<b>Review/Exam Week - 03 Jun 2024</b>		
Module/Topic	Chapter	Events and Submissions/Topic
		<b>Case study</b> Due: Review/Exam Week Friday (7 June 2024) 11:59 am AEST
<b>Exam Week - 10 Jun 2024</b>		
Module/Topic	Chapter	Events and Submissions/Topic

## Term Specific Information

Unit Coordinator  
Dr. Lily D Li  
Senior Lecturer, CQUniversity Brisbane Campus  
Phone: 07 3023 4253 Email: l.li@cqu.edu.au

## Assessment Tasks

### 1 Assignment 1 - Java Console Application

#### Assessment Type

Practical and Written Assessment

#### Task Description

In this assignment, you will develop a Java console application to meet the requirements as per the given specification. You will demonstrate your skills in using Java data types, decisions, looping, methods, classes and objects in the application development. You will develop the solution using an Integrated Development Environment (IDE). By completing this assignment, you will achieve the following learning outcomes:

- explain the principles of object-oriented programming and
- implement and test object-oriented programs using a modern programming language.

Apart from developing the application, a report describing how long it took to create the whole program, any problems encountered, and screenshots of the output produced, including your testing annotations, must be submitted. The full assignment specification will be available on the Moodle unit website.

#### Assessment Due Date

Week 6 Friday (19 Apr 2024) 11:59 pm AEST

#### Return Date to Students

Week 8 Friday (3 May 2024)

#### Weighting

20%

#### Assessment Criteria

Your assignment solution will be marked mainly based on the following:

Development of suitable Java classes including constructors and other relevant methods. The classes adhere to the principles of OOP and follow the best programming principles.

A running application that has the following:

- executes without logical and syntax errors
- receives correct inputs and produces appropriate outputs, and
- provides appropriate error messages.

Your written report on the required aspects regarding your practical experience of application development and testing. The detailed assessment criteria will be provided along with the assignment specification.

#### Referencing Style

- [Harvard \(author-date\)](#)

#### Submission

Online

#### Learning Outcomes Assessed

- Explain the principles of object-oriented programming
- Implement and test object-oriented programs using a modern programming language

### 2 Assignment 2 - JAVA GUI application

#### Assessment Type

Practical and Written Assessment

### Task Description

In this assignment, you are required to develop a Java GUI-based application using JavaFX technology. You will demonstrate your skills in using OO concepts such as inheritance and polymorphism, using GUI components and event handling mechanisms to build user-friendly applications.

By completing this assignment, you will achieve the following learning outcomes:

- implement and test object-oriented programs using a modern programming language,
- build interactive software applications using Graphical User Interface components, and
- apply the concept of exception handling and file data manipulation in object-oriented code.

Apart from developing the application, a report describing how long it took to create the whole program, any problems encountered, and screenshots of the output produced, including your testing annotations, must be submitted. The full assignment specification will be available on the Moodle unit website.

### Assessment Due Date

Week 10 Friday (17 May 2024) 11:59 pm AEST

### Return Date to Students

Week 12 Friday (31 May 2024)

### Weighting

30%

### Assessment Criteria

Your assignment submission will be assessed mainly on the following:

- Developing Java classes that implement inheritance and/or polymorphism,
- Using appropriate GUI controls and Listeners to handle events,
- Achieving the functionalities as per specifications, and
- Adhering to good programming practice
- Your written report on the required aspects regarding your practical experience of application testing and development.

The detailed assessment criteria will be provided along with the assignment specification.

### Referencing Style

- [Harvard \(author-date\)](#)

### Submission

Online

### Learning Outcomes Assessed

- Implement and test object-oriented programs using a modern programming language
- Build interactive software applications using Graphical User Interface components
- Apply the concept of exception handling and file data manipulation in object-oriented code

## 3 Lab hands-on practice

### Assessment Type

Practical Assessment

### Task Description

This assessment item consists of 10 weekly submissions from your lab exercises in weeks 3-12 (inclusive).

The weekly tutorial/lab exercises include the questions and the hands-on projects covered in that week. You will attend the tutorial/lab classes and **submit one lab project** for the assessment.

On-campus students: You are required to attend the tutorial/lab classes as per the timetable schedule. The tutor will guide you to complete the exercises.

Distance Education students: You will also be required to submit a specified hands-on project each week and you may be asked to meet with your tutor or the unit coordinator to discuss your work. The unit coordinator will communicate with you further about this assessment item.

Check the weekly lab sheets for the details.

### Assessment Due Date

Due in your weekly tutorial/lab class from weeks 3 -12 (inclusive).

### Return Date to Students

Marks will be returned before your next tutorial submission is due.



**Weighting**

20%

**Assessment Criteria**

Weekly tutorial/lab will test your understanding of the topics covered in that week.

Each week's submission (week 3 - 12) is worth 2% of the overall mark for the unit. The total mark of this assessment is worth 20% of the unit's overall mark.

The assessments will be marked based on your engagement and performance in the tutorial/lab exercises.

**Referencing Style**

- [Harvard \(author-date\)](#)

**Submission**

Online

**Learning Outcomes Assessed**

- Implement and test object-oriented programs using a modern programming language
- Build interactive software applications using Graphical User Interface components
- Apply the concept of exception handling and file data manipulation in object-oriented code
- Apply appropriate data structure in object-oriented design and development.

## 4 Case study

**Assessment Type**

Case Study

**Task Description**

In this assignment, you will be provided with a business case. As a software engineer, you are required to propose a software solution for the business case. The proposed solution should include functional requirements (e.g., user stories), conceptual OO design (classes, objects and main data structure), Interface design (GUI) and testing plan. You will need to submit a technical report containing the following:

- A brief description of the project
- A list of high-level user requirements, e.g., user stories, use case diagrams
- OO Design including the data structure used, e.g., class diagrams, ArrayList
- Design of user interfaces, e.g. wireframes, hand-drawn prototypes
- A brief outline of the testing plan

Plus

- A reflection on the project - problems encountered and lessons learned.

By completing this assignment, you will achieve the following learning outcomes:

- explain the principles of object-oriented programming, and
- apply appropriate data structure in object-oriented design and development.

The full specification will be available on the Moodle unit website.

Word limit: Although this assignment has no word count limit, at least 1000 words is estimated if you address all the abovementioned items.

**Assessment Due Date**

Review/Exam Week Friday (7 June 2024) 11:59 am AEST

**Return Date to Students**

The results will be returned on the certification day.

**Weighting**

30%

**Assessment Criteria**

The assignment will be marked based on the responses to the case study questions, such as:

- Requirements are clear and concise.
- OO design is reasonable and feasible.
- Data structure is appropriate for the design.

- UI is user-friendly.
- Data flow is clear and makes sense.
- The test plan is reasonable.
- Reflection report is clear.
- Professional presentation (e.g., report format, structure, cohesion).

The detailed assessment criteria will be provided along with the assignment specification.

### Referencing Style

- [Harvard \(author-date\)](#)

### Submission

Online

### Learning Outcomes Assessed

- Explain the principles of object-oriented programming
- Apply appropriate data structure in object-oriented design and development.

## Academic Integrity Statement

As a CQUniversity student you are expected to act honestly in all aspects of your academic work.

Any assessable work undertaken or submitted for review or assessment must be your own work. Assessable work is any type of work you do to meet the assessment requirements in the unit, including draft work submitted for review and feedback and final work to be assessed.

When you use the ideas, words or data of others in your assessment, you must thoroughly and clearly acknowledge the source of this information by using the correct referencing style for your unit. Using others' work without proper acknowledgement may be considered a form of intellectual dishonesty.

Participating honestly, respectfully, responsibly, and fairly in your university study ensures the CQUniversity qualification you earn will be valued as a true indication of your individual academic achievement and will continue to receive the respect and recognition it deserves.

As a student, you are responsible for reading and following CQUniversity's policies, including the [Student Academic Integrity Policy and Procedure](#). This policy sets out CQUniversity's expectations of you to act with integrity, examples of academic integrity breaches to avoid, the processes used to address alleged breaches of academic integrity, and potential penalties.

### What is a breach of academic integrity?

A breach of academic integrity includes but is not limited to plagiarism, self-plagiarism, collusion, cheating, contract cheating, and academic misconduct. The Student Academic Integrity Policy and Procedure defines what these terms mean and gives examples.

### Why is academic integrity important?

A breach of academic integrity may result in one or more penalties, including suspension or even expulsion from the University. It can also have negative implications for student visas and future enrolment at CQUniversity or elsewhere. Students who engage in contract cheating also risk being blackmailed by contract cheating services.

### Where can I get assistance?

For academic advice and guidance, the [Academic Learning Centre \(ALC\)](#) can support you in becoming confident in completing assessments with integrity and of high standard.

### What can you do to act with integrity?



**Be Honest**

If your assessment task is done by someone else, it would be dishonest of you to claim it as your own



**Seek Help**

If you are not sure about how to cite or reference in essays, reports etc, then seek help from your lecturer, the library or the Academic Learning Centre (ALC)



**Produce Original Work**

Originality comes from your ability to read widely, think critically, and apply your gained knowledge to address a question or problem