



COIT11134 Object Oriented Programming

Term 2 - 2020

Profile information current as at 07/05/2024 12:50 am

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 is designed to extend your understanding of object-oriented programming principles, complex data structures and algorithms. You will learn to develop interactive software applications using a modern programming language, integrated development environment (IDE), and graphical user interface (GUI) components. The strengths and weaknesses of the techniques are also considered. You will develop skills in applying object-oriented programming concepts and algorithm development to implement software solutions. You will apply these skills in implementing software applications to solve 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 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 2 - 2020

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

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: 10%

2. **Practical and Written Assessment**

Weighting: 15%

3. **Practical and Written Assessment**

Weighting: 20%

4. **Take Home Exam**

Weighting: 55%

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 Student Evaluation

Feedback

Weekly tutorial assessments are not sufficiently weighted

Recommendation

The assessment structure should be changed to increase the number of take-home assessments and to better reflect the effort required to complete the assessment.

Feedback from Student Evaluation

Feedback

The material is too advanced for students who lack the prerequisite knowledge

Recommendation

Review the prerequisite programming unit and provide links to review materials to increase incoming students' knowledge, skills and confidence to complete this unit.

Feedback from Student Evaluation and Unit coordinator.

Feedback

The current textbook is more suitable for professional programmers who want to learn Java. The current textbook lacks appropriate exercises. The source code provided is difficult to read without an appropriate visual (colour) mark-up.

Recommendation

Remove the requirements of the textbook and provide students with a range of free resources that support their learning.

Feedback from Student Evaluation

Feedback

Assessment feedback was insufficient or unclear

Recommendation

The assessment rubric and feedback format should be updated to make it easier to provide detailed feedback. Also, assessment answers should be discussed in tutorials after the return of assessment.

Unit Learning Outcomes

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

1. Explain the principles of object-oriented programming
2. Implement object-oriented programs using a modern programming language
3. Build interactive software applications using Graphical User Interface components
4. Apply self-designed and existing algorithms in problem solutions
5. Use complex data structures in software application development.

Australian Computer Society (ACS) recognises the Skills Framework for the Information Age (SFIA). SFIA is 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 at

<https://www.acs.org.au/professionalrecognition/mysfia-b2c.html>

This unit contributes to the following workplace skills as defined by SFIA 7. 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 - 10%	•	•			
2 - Practical and Written Assessment - 15%		•	•	•	•
3 - Practical and Written Assessment - 20%			•		•
4 - Take Home Exam - 55%	•			•	

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					

Alignment of Assessment Tasks to Graduate Attributes

Assessment Tasks	Graduate Attributes									
	1	2	3	4	5	6	7	8	9	10
1 - Practical and Written Assessment - 10%	•	•	•	•		•				

Assessment Tasks	Graduate Attributes									
	1	2	3	4	5	6	7	8	9	10
2 - Practical and Written Assessment - 15%	•	•	•	•		•		•		
3 - Practical and Written Assessment - 20%	•	•	•	•				•		
4 - Take Home Exam - 55%	•	•	•							

Textbooks and Resources

Textbooks

COIT11134

Prescribed

Core Java Volume I--Fundamentals

Edition: 11 (2018)

Authors: Cay S. Horstmann

Pearson Higher Ed USA

ISBN: 9780135166307

Binding: Paperback

Additional Textbook Information

If you prefer to study with a paper copy, they are available at the CQUni Bookshop here: <http://bookshop.cqu.edu.au> (search on the Unit code). eBooks are available at the publisher's website.

[View textbooks at the CQUniversity Bookshop](#)

IT Resources

You will need access to the following IT resources:

- CQUniversity Student Email
- Internet
- Unit Website (Moodle)
- Java Development Kit (JDK) 1.8 or later
- Apache Netbeans IDE 11.3, download from <https://netbeans.apache.org/download/nb113/nb113.html>

Referencing Style

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

For further information, see the Assessment Tasks.

Teaching Contacts

Farzad Sanati Unit Coordinator

f.sanati@cqu.edu.au

Schedule

Week 1 - 13 Jul 2020

Module/Topic	Chapter	Events and Submissions/Topic
Revision of Java Classes, Objects, and Methods	Chapter 4	

Week 2 - 20 Jul 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Inheritance	Chapter 5	
Week 3 - 27 Jul 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Polymorphism	Chapter 5	
Week 4 - 03 Aug 2020		
Module/Topic	Chapter	Events and Submissions/Topic
GUI and Event Handling	Chapters 10 and 11	
Week 5 - 10 Aug 2020		
Module/Topic	Chapter	Events and Submissions/Topic
GUI and Event Handling	Chapters 10 and 11	
Vacation Week - 17 Aug 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Week 6 - 24 Aug 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Java Exception Handling, and revisit files IO	Chapter 7 and Chapter 3	Assignment 1 Due: Week 6 Friday (28 Aug 2020) 12:00 pm AEST
Week 7 - 31 Aug 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Interfaces, Lambda expressions, and Inner Classes	Chapter 6	
Week 8 - 07 Sep 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Generic methods	Chapter 8	
Week 9 - 14 Sep 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Generic Classes	Chapter 8	
Week 10 - 21 Sep 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Collection, Linked Lists, and Algorithms	Chapter 9	
Week 11 - 28 Sep 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Stacks, Queues, and Trees	Chapter 9	Assignment 2 Due: Week 11 Friday (2 Oct 2020) 12:00 pm AEST
Week 12 - 05 Oct 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Final Revision and additional readings	Additional lecture materials and resources from the publisher/other sources.	
Review/Exam Week - 12 Oct 2020		
Module/Topic	Chapter	Events and Submissions/Topic
Exam Week - 19 Oct 2020		
Module/Topic	Chapter	Events and Submissions/Topic

Term Specific Information

Unit Coordinator: Farzad Sanati
email: f.sanati@cqu.edu.au
phone: +61 7 4726 5386

Assessment Tasks

1 Completion and submission of set weekly exercises

Assessment Type

Practical and Written Assessment

Task Description

Object-Oriented Programming is a unit with cumulative content requiring regular and sequential studies.

This assessment is to enhance student knowledge by encouraging the consistent study of theory and completion of weekly exercise.

Students will require to complete and submit 10 selective weekly tutorial exercises each worth 1% of the total available marks for this assessment.

Assessment Due Date

Submit each week commencing from Week 2

Return Date to Students

Each weekly tutorial exercise will be marked two week after submission

Weighting

10%

Assessment Criteria

Weekly tutorial exercise is completed and submitted.

Referencing Style

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

Submission

Online

Learning Outcomes Assessed

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

Graduate Attributes

- Communication
- Problem Solving
- Critical Thinking
- Information Literacy
- Information Technology Competence

2 Assignment 1

Assessment Type

Practical and Written Assessment

Task Description

In this assignment, you have to develop a Java GUI based application to meet the requirements of the given case study. By completing this assignment you will learn to:

- Use more than one Java class to implement Encapsulation, Inheritance, Aggregation and/or Polymorphism.

- Build an interactive software application using graphical user interface components.

You will develop the software solution using an Integrated Development Environment (IDE). This assignment must be submitted on-line through the Moodle assignment submission system. The full specification will be available in the unit Moodle site.

Assessment Due Date

Week 6 Friday (28 Aug 2020) 12:00 pm AEST

Return Date to Students

Week 8 Friday (11 Sept 2020)

Two weeks after submission

Weighting

15%

Assessment Criteria

The detailed assessment criteria will be provided along with the assignment specification. Your assignment will be assessed mainly on the following:

- Developing Java classes that implement Encapsulation, Inheritance, Aggregation and/or Polymorphism
- Using suitable Java Layout Managers, GUI controls and Listeners
- Adhering to good programming practice

Penalties related to late submission and plagiarism will be applied as per University policy.

Referencing Style

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

Submission

Online

Learning Outcomes Assessed

- Implement object-oriented programs using a modern programming language
- Build interactive software applications using Graphical User Interface components
- Apply self-designed and existing algorithms in problem solutions
- Use complex data structures in software application development.

Graduate Attributes

- Communication
- Problem Solving
- Critical Thinking
- Information Literacy
- Information Technology Competence
- Ethical practice

3 Assignment 2

Assessment Type

Practical and Written Assessment

Task Description

In this assignment, you have to develop a Java GUI based application that will extend the software solution, developed as part of your Assignment 1, with additional functionality as required in the given specification. By completing this assignment you will learn to:

- Implement object-oriented programs using a modern programming language.
- Build interactive software applications using Graphical User Interface components.
- Apply self-designed and existing algorithms in problem solutions.
- Use complex data structures in software application development.

You will develop the software solution using an Integrated Development Environment (IDE). This assignment must be submitted on-line through the Moodle assignment submission system. The full specification will be available in the unit Moodle site.

Assessment Due Date

Week 11 Friday (2 Oct 2020) 12:00 pm AEST

Two weeks after the due date or two weeks after submission, whichever is later.

Return Date to Students

Exam Week Monday (19 Oct 2020)

Weighting

20%

Assessment Criteria

The detailed assessment criteria will be provided along with the assignment specification. Your assignment will be assessed mainly on the following:

- Using ArrayList or LinkedList or any other data structure
- Extending Java classes and/or implementing Interfaces
- Using suitable Java Layout Managers, GUI controls and Listeners
- Developing Java classes for file reading and/or writing
- Adhering to good programming practice

Penalties related to late submission and plagiarism will be applied as per University policy.

Referencing Style

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

Submission

Online

Learning Outcomes Assessed

- Build interactive software applications using Graphical User Interface components
- Use complex data structures in software application development.

Graduate Attributes

- Communication
- Problem Solving
- Critical Thinking
- Information Literacy
- Ethical practice

4 Take Home Final Exam

Assessment Type

Take Home Exam

Task Description

Exam in this unit is a take-home exam for this term as invigilated exams are not possible during the COVID-19 pandemic. The Exam will be conducted during the exam week of the term.

Assessment Due Date

During exam week as per the University exam schedule

Return Date to Students

Marks will be released on certification date

Weighting

55%

Assessment Criteria

- Explain the principles of object-oriented programming
- Make corrections to rectify errors in each given Java code
- Apply self-designed and existing algorithms in problem solutions

Referencing Style

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

Submission

Online

Learning Outcomes Assessed

- Explain the principles of object-oriented programming
- Apply self-designed and existing algorithms in problem solutions

Graduate Attributes

- Communication
- Problem Solving
- Critical Thinking

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