



COIT20256 *Data Structures and Algorithms*

Term 3 - 2022

Profile information current as at 20/04/2024 10:49 am

All details in this unit profile for COIT20256 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

In this unit, you will study advanced data structures and algorithms for software development using an object-oriented programming language. You will learn how to design and build classes, throw exceptions, and extend a class using inheritance and polymorphism. You will practise these concepts and develop applications with front-end Graphical User Interface (GUI) components and backend databases using database programming. You will build software applications using complex data structures and Application Programming Interfaces (APIs). You will gain an understanding of basic algorithms, and learn to evaluate algorithmic performance and assess the correct use of different data structures. You will be introduced to functional programming using Lambdas and Streams. You will obtain hands-on experience using all the concepts by completing programming exercises.

Details

Career Level: *Postgraduate*

Unit Level: *Level 9*

Credit Points: 6

Student Contribution Band: 8

Fraction of Full-Time Student Load: 0.125

Pre-requisites or Co-requisites

Pre-requisite: COIT20245 Introduction to Programming

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 3 - 2022

- Brisbane
- Melbourne
- Online
- 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 Postgraduate 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 Assessment**

Weighting: 20%

2. **Practical Assessment**

Weighting: 10%

3. **Practical Assessment**

Weighting: 20%

4. **Examination**

Weighting: 50%

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 Unit Coordinator

Feedback

Some of the tutorial instructions refer to questions in the textbook making it difficult to follow.

Recommendation

Update the tutorial instructions, avoiding referring to external resources.

Unit Learning Outcomes

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

1. Design classes which use inheritance, polymorphism, and exception handling
2. Develop multi-layered software solutions, focusing on data structures and algorithms
3. Integrate data sets using complex data structures such as linked lists, stacks, and queues
4. Evaluate performance of different algorithms in problem solving
5. Investigate socially innovative practices in software development
6. Create Lambda expressions and streams using functional programming.

Australian Computer Society (ACS) recognises the Skills Framework for the Information Age (SFIA). SFIA provides a consistent definition of ICT skills. SFIA is adopted by organisations, governments, and individuals in many countries and is increasingly 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:

- Software Design (SWDN)
- System Integration and Build (SINT)
- Programming/Software Development (PROG)
- Data modelling and design (DTAN)
- Database Design (DBDS)
- Testing (TEST)
- User experience analysis (UNAN)
- 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	6
1 - Practical Assessment - 20%	•	•			•	
2 - Practical Assessment - 10%				•		

Assessment Tasks	Learning Outcomes					
	1	2	3	4	5	6
3 - Practical Assessment - 20%	•	•	•		•	•
4 - Examination - 50%			•	•		•

Alignment of Graduate Attributes to Learning Outcomes

Graduate Attributes	Learning Outcomes					
	1	2	3	4	5	6
1 - Knowledge	○	○	○	○		○
2 - Communication	○		○		○	
3 - Cognitive, technical and creative skills	○	○	○	○		○
4 - Research		○				
5 - Self-management		○				
6 - Ethical and Professional Responsibility					○	
7 - Leadership						
8 - 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
1 - Practical Assessment - 20%	○	○	○		○			
2 - Practical Assessment - 10%	○	○	○					
3 - Practical Assessment - 20%	○	○	○	○	○	○		
4 - Examination - 50%	○	○	○					

Textbooks and Resources

Textbooks

COIT20256

Prescribed

Java How to Program, Late Objects, Global Edition

11th Edition

Authors: Paul Deitel and Harvey Deitel

Pearson Higher Ed US

ISBN: 978-1292273730

Binding: eBook

Additional Textbook Information

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

[View textbooks at the CQUniversity Bookshop](#)

IT Resources

You will need access to the following IT resources:

- CQUniversity Student Email
- Internet
- Unit Website (Moodle)
- JDK 17 (available from <https://www.oracle.com/java/technologies/downloads/#java17>)
- Apache NetBeans IDE 12.5 or higher (<https://netbeans.apache.org/download/index.html>)
- MySQL Community Server 5.7 or higher (<https://dev.mysql.com/downloads/windows/installer/8.0.html>)
- Scene Builder 12 or higher (available from <https://gluonhq.com/products/scene-builder/>)
- JavaFX 17.0.4 (available from <https://gluonhq.com/products/javafx/>)

Referencing Style

All submissions for this unit must use the referencing styles below:

- [Harvard \(author-date\)](#)
- [American Psychological Association 7th Edition \(APA 7th edition\)](#)

For further information, see the Assessment Tasks.

Teaching Contacts

Mary Tom Unit Coordinator

m.tom@cqu.edu.au

Schedule

Week 1 - 07 Nov 2022

Module/Topic	Chapter	Events and Submissions/Topic
Classes and Objects: A Deeper Look	8	

Week 2 - 14 Nov 2022

Module/Topic	Chapter	Events and Submissions/Topic
Inheritance	9	

Week 3 - 21 Nov 2022

Module/Topic	Chapter	Events and Submissions/Topic
Polymorphism and Interfaces	10	Class Activity 1 Due: Week 3 Friday (25 Nov 2021) 11.45 pm AEST
Week 4 - 28 Nov 2022		
Module/Topic	Chapter	Events and Submissions/Topic
JavaFX GUI and Event-Driven Programming	12 and 13	Class Activity 2 Due: Week 4 Friday (2 Dec 2021) 11.45 pm AEST
Vacation Week - 05 Dec 2022		
Module/Topic	Chapter	Events and Submissions/Topic
Week 5 - 12 Dec 2022		
Module/Topic	Chapter	Events and Submissions/Topic
Exception Handling, Files, Streams and Serialization	11 and 15	Class Activity 3 Due: Week 5 Friday (16 Dec 2021) 11:45 pm AEST
Week 6 - 19 Dec 2022		
Module/Topic	Chapter	Events and Submissions/Topic
Recursion	18	Assignment 1 Due: Week 6 Thursday (22 Dec 2022) 11:45 pm AEST
Vacation Week - 26 Dec 2022		
Module/Topic	Chapter	Events and Submissions/Topic
Week 7 - 02 Jan 2023		
Module/Topic	Chapter	Events and Submissions/Topic
Generic Collections: Lists, Sets, Maps, and Priority Queue	16	Class Activity 4 Due: Week 7 Friday (6 Jan 2022) 11.45 pm AEST
Week 8 - 09 Jan 2023		
Module/Topic	Chapter	Events and Submissions/Topic
Lambdas and Streams	17	Class Activity 5 Due: Week 8 Friday (13 Jan 2022) 11:45 pm AEST
Week 9 - 16 Jan 2023		
Module/Topic	Chapter	Events and Submissions/Topic
Accessing Database with JDBC	24	
Week 10 - 23 Jan 2023		
Module/Topic	Chapter	Events and Submissions/Topic
Generic Classes and Methods and Algorithmic Efficiency	20, 19 Section 19.3 and lecture notes	
Week 11 - 30 Jan 2023		
Module/Topic	Chapter	Events and Submissions/Topic
Graphs and applications	Lecture Notes	Assignment 2 Due: Week 11 Tuesday (31 Jan 2023) 11:45 pm AEST
Week 12 - 06 Feb 2023		
Module/Topic	Chapter	Events and Submissions/Topic
Implementing Stacks, Queues and Binary Search Tree	21	
Exam Week - 13 Feb 2023		
Module/Topic	Chapter	Events and Submissions/Topic

Term Specific Information

Unit Coordinator

Dr Mary Tom
College of Information & Communications Technology
School of Engineering and Technology
Central Queensland University
Brisbane QLD 4000, Australia
Phone: +61 7 3295 1119
Email: m.tom@cqu.edu.au

Assessment Tasks

1 Assignment 1

Assessment Type

Practical Assessment

Task Description

In this assignment you will analyse the given problem, model and design data structures using UML class diagrams, and develop a software solution applying the Object-Oriented programming concepts of classes, inheritance, and polymorphism. You will programmatically read data and save data in a file and apply exception handling for capturing system and run-time errors. You will also design and develop a graphical user interface (GUI) for the software solution applying event-driven programming. This assessment task is to design, code, debug, and test a software application using the topics learnt in Weeks 1 - 5. Further details and marking criteria are in the Assignment 1 Specification document available from the Unit website.

Assessment Due Date

Week 6 Thursday (22 Dec 2022) 11:45 pm AEST

Return Date to Students

Week 8 Thursday (12 Jan 2023)

Weighting

20%

Assessment Criteria

1. Correct analysis of given case study, appropriate choice and design of data structures for application development.
2. Correct application of classes, inheritance, polymorphism, and exception handling.
3. Demonstration of socially innovative practices in software development.
4. Practice of data persistence by writing data to a file.
5. Implementation of Graphical User Interface (GUI) following user interface design guidelines.
6. Effective use of good coding practices.
7. Rigorous testing of software applications.

Referencing Style

- [Harvard \(author-date\)](#)
- [American Psychological Association 7th Edition \(APA 7th edition\)](#)

Submission

Online

Submission Instructions

Submit one .zip file containing the source code files (.java), GUI files (.FXML), (.css), data files (.css) and the report file (.doc or .docx).

Learning Outcomes Assessed

- Design classes which use inheritance, polymorphism, and exception handling
- Develop multi-layered software solutions, focusing on data structures and algorithms
- Investigate socially innovative practices in software development

Graduate Attributes

- Knowledge
- Communication
- Cognitive, technical and creative skills
- Self-management

2 Class Activities

Assessment Type

Practical Assessment

Task Description

This assessment task is to be completed and submitted as part of your weekly workshop sessions. It consists of a series of 5 practicals to be completed in weeks 3 – 8 (inclusive), except Week 6. You complete and submit the required tasks shown in your tutorial by end of the corresponding week before the due date given in the Weekly Schedule. Each submission has two marks allocated.

Assessment Due Date

Submit before the due date shown in the Weekly Schedule

Return Date to Students

The marked submissions will be returned to you after a week from the date of submission.

Weighting

10%

Assessment Criteria

1. Correct completion of required tutorial tasks.
2. Correct application of the theoretical aspects applicable to the task.
3. Adherence to good coding practices.
4. Timely completion and submission of activity.

Referencing Style

- [Harvard \(author-date\)](#)
- [American Psychological Association 7th Edition \(APA 7th edition\)](#)

Submission

Online

Learning Outcomes Assessed

- Evaluate performance of different algorithms in problem solving

Graduate Attributes

- Knowledge
- Communication
- Cognitive, technical and creative skills

3 Assignment 2

Assessment Type

Practical Assessment

Task Description

In this assignment, you are required to analyse the given problem, model and design the required data structures using UML class diagrams, and use complex data structures such as linked lists, queues, and streams. You will develop a software application having a three tiered architecture with a front-end interactive Graphical User Interface (GUI), the middle layer implementing the business logic, and the back-end database storing necessary data. You will practice paired programming in your software development task. You will individually prepare and submit a report documenting testing, team work experience, and theoretical aspects applied in developing the application. This assessment task includes design, document, develop code, debug, and test a software application applying topics learnt in Weeks 1 - 10. Further details and marking criteria are in the Assignment 2 specification document available from the Unit website.

Assessment Due Date

Week 11 Tuesday (31 Jan 2023) 11:45 pm AEST

Return Date to Students

Week 12 Friday (10 Feb 2023)

Weighting

20%

Assessment Criteria

1. Correct design of classes which use inheritance, polymorphism, and exception handling
2. Correct integration of data sets using complex data structures such as linked lists, stacks, and queues
3. Correct implementation of a programmatically created database.
4. Correct application of Lambda expressions and streams using functional programming.
5. Rigorous testing of software application
6. Demonstration of socially innovative practices in software development
7. Practice of paired programming.

Referencing Style

- [Harvard \(author-date\)](#)
- [American Psychological Association 7th Edition \(APA 7th edition\)](#)

Submission

Online

Submission Instructions

Submit one .zip file containing the source code files (.java), and GUI components files (.FXML), (.css) per team and the individual report file (.doc) by each member of the team.

Learning Outcomes Assessed

- Design classes which use inheritance, polymorphism, and exception handling
- Develop multi-layered software solutions, focusing on data structures and algorithms
- Integrate data sets using complex data structures such as linked lists, stacks, and queues
- Investigate socially innovative practices in software development
- Create Lambda expressions and streams using functional programming.

Graduate Attributes

- Knowledge
- Communication
- Cognitive, technical and creative skills
- Research
- Self-management
- Ethical and Professional Responsibility

Examination

Outline

Complete an invigilated examination.

Date

During the examination period at a CQUniversity examination centre.

Weighting

50%

Length

180 minutes

Exam Conditions

Open Book.

Materials

Dictionary - non-electronic, concise, direct translation only (dictionary must not contain any notes or comments).

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