COIT20256 *Data Structures and Algorithms*
Term 1 - 2020

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.

Corrections

Unit Profile Correction added on 24-04-20
The end of term examination has now been changed to an alternate form of assessment. Please see your Moodle Unit website for details of the assessment.

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 1 - 2020

- Brisbane
- 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 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 Student evaluation

Feedback

Lecture videos need to be updated and provided for all lectures.

Recommendation

Lecture videos will be updated for all lectures.

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. Analyse, design and implement multi-layered software solutions, focusing on data structures and algorithms
3. Evaluate a variety of data structures including linked lists, stacks, queues, streams, and search trees
4. Compare and contrast performance of different algorithms in problem solving
5. Investigate socially innovative practices in software development
6. Build software applications using functional programming with Lambda expressions and streams.

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:

- 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

— N/A Level	○ Introductory Level	○ Intermediate Level	● Graduate Level	○ Professional Level	○ Advanced Level

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%			●			●
3 - Practical Assessment - 20%	●	●		●	●	

Assessment Tasks	Learning Outcomes					
	1	2	3	4	5	6
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 (11e)
Global Edition (11e) (2019)
Authors: Paul Deitel and Harvey Deitel
Pearson Higher Ed US
USA
ISBN: 978-1292273730
Binding: Paperback

Additional Textbook Information
Copies can be purchased from 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)
- Java SE 11
- MySQL Server 5.7 or later
- NetBeans IDE 11
- JavaFX 11.0 or later

Referencing Style

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

- American Psychological Association 6th Edition (APA 6th edition)
- Harvard (author-date)

For further information, see the Assessment Tasks.

Teaching Contacts

Mary Tom Unit Coordinator
m.tom@cqu.edu.au

Schedule

Week 1 - 09 Mar 2020

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

Week 2 - 16 Mar 2020

Module/Topic	Chapter	Events and Submissions/Topic
Object-Oriented Programming: Inheritance	9	

Week 3 - 23 Mar 2020

Module/Topic	Chapter	Events and Submissions/Topic
Object-Oriented Programming: Polymorphism and Interfaces	10	Assessment 2: Part 1 due in Week 3 workshop

Week 4 - 30 Mar 2020

Module/Topic	Chapter	Events and Submissions/Topic
JavaFX Grphical User Interfaces	12, and 13	Assessment 2: Part 2 due in Week 4 workshop

Week 5 - 06 Apr 2020

Module/Topic	Chapter	Events and Submissions/Topic
Exception Handling, Files, Streams and Serialization	11 and 15	Assessment 2: Part 3 due in Week 5 workshop

Vacation Week - 13 Apr 2020

Module/Topic	Chapter	Events and Submissions/Topic

Week 6 - 20 Apr 2020

Module/Topic	Chapter	Events and Submissions/Topic
Object-Oriented Design	33 (online chapter)	**Assignment 1** Due: Week 6 Thursday (23 Apr 2020) 11:45 pm AEST

Week 7 - 27 Apr 2020

Module/Topic	Chapter	Events and Submissions/Topic
Generic Collections	16	Assessment 2: Part 4 due in Week 7 workshop

Week 8 - 04 May 2020

Module/Topic	Chapter	Events and Submissions/Topic
Lambdas and Streams, and Algorithmic Efficiency	17 and 19	Assessment 2: Part 5 due in Week 8 workshop

Week 9 - 11 May 2020

Module/Topic	Chapter	Events and Submissions/Topic
Accessing Database with JDBC	24	Assessment 2: Part 6 due in Week 9 workshop

Week 10 - 18 May 2020

Module/Topic	Chapter	Events and Submissions/Topic
Generic Classes and Methods	20	Assessment 2: Part 7 due in Week 9 workshop

Week 11 - 25 May 2020

Module/Topic	Chapter	Events and Submissions/Topic
Recursion	18	**Assignment 2** Due: Week 11 Thursday (28 May 2020) 11:45 pm AEST

Week 12 - 01 Jun 2020

Module/Topic	Chapter	Events and Submissions/Topic
Custom Generic Data Structures	21	

Review/Exam Week - 08 Jun 2020

Module/Topic	Chapter	Events and Submissions/Topic

Exam Week - 15 Jun 2020

Module/Topic	Chapter	Events and Submissions/Topic

Unit Coordinator
Dr Mary Tom
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 demonstrate your ability to 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 also design and develop a graphical user interface (GUI) for the software solution. This assessment task is to design, code, debug, and test a software application using the topics learnt in Weeks 1 - 5. Further details are in the Assignment 1 specification document available from the Unit website.

Assessment Due Date
Week 6 Thursday (23 Apr 2020) 11:45 pm AEST

Return Date to Students
Week 9 Monday (11 May 2020)

Weighting
20%

Assessment Criteria

1. Design and implement appropriate data structures for application development
2. Analyse, develop and implement software solutions with the focus of data structures and algorithms
3. Apply classes, inheritance, polymorphism, and exception handling
4. Appropriate use of Graphical User Interface (GUI)
5. Effective use of good coding practices
6. Rigorous testing of software applications.

Referencing Style

- [American Psychological Association 6th Edition (APA 6th edition)](#)
- [Harvard (author-date)](#)

Submission
Online

Submission Instructions
Submit one .zip file containing the source code files (.java) and the report file (.doc). Do not submit the zipped project folder or compiled binaries (.class or .jar)

Learning Outcomes Assessed

- Design classes which use inheritance, polymorphism, and exception handling
- Analyse, design and implement multi-layered software solutions, focusing on data structures and algorithms
- Compare and contrast performance of different algorithms in problem solving
- Investigate socially innovative practices in software development

Graduate Attributes

- Knowledge
- Communication
- Cognitive, technical and creative skills

- Self-management

2 Workshop Activites

Assessment Type
Practical Assessment

Task Description
This assessment item is to be developed and submitted as part of your weekly workshop sessions. It consists of a series of 7 practicals to be completed in weeks 3 – 10 (inclusive), except Week 6. For on campus students this work is due in the weekly workshop, not the due date shown in Moodle. The date given in Moodle is to cater for all scheduled workshop classes. *No marks will be awarded for work submitted outside the workshop/tutorial class or for late submissions.*
On-campus students

1. This assessment task must be completed and submitted in your weekly workshop **as and when directed** by your lecturer/tutor.
2. Marks may be deducted if your tutor is not satisfied with your progress or understanding of the work.

Online students

1. Submit your weekly work by the due date shown on the unit website for the corresponding week. **Late submissions will be awarded 0 marks.**
2. The unit coordinator is your tutor and may make arrangements to discuss your work with you.

Assessment Due Date

This task commences with an "in-class" submission in week 3 and continues with "in-class" weekly submissions until the end of week 10. There is a total of 8 weekly submissions. On-campus students must complete and submit the scheduled work in their weekly workshop.

Return Date to Students
Week 4 Friday (3 Apr 2020)
The weekly submissions will be marked and returned within 1 week of their submission.

Weighting
10%

Assessment Criteria
Practical questions included in the weekly workshop material will be used to assess your understanding of the topics taught in that week and your ability to apply those principles to the given scenario. This will include designing data structures, algorithms, or writing part of source code. You are awarded total 1.5 marks for each weekly submission except for Week 9 submission having 1 mark.

Referencing Style

- American Psychological Association 6th Edition (APA 6th edition)
- Harvard (author-date)

Submission
Online

Submission Instructions
Submit one zip file when multiple file submissions are required.

Learning Outcomes Assessed

- Evaluate a variety of data structures including linked lists, stacks, queues, streams, and search trees
- Build software applications using functional programming with Lambda expressions and streams.

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 generic data structures such as linked lists, queues, and streams, and 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. This assessment task
includes design, document, develop code, debug , and test a java application applying topics learnt in Weeks 1 - 10.
Further details are in the Assignment 2 specification document available from the Moodle Unit website .

Assessment Due Date
Week 11 Thursday (28 May 2020) 11:45 pm AEST

Return Date to Students
Review/Exam Week Thursday (11 June 2020)

Weighting
20%

Assessment Criteria

1. Efficient object-oriented program design.
2. Design and implement appropriate data structures for application development
3. Evaluate a variety of data structures and algorithmic approaches
4. Effective use of good programming practice/techniques
5. Apply classes, inheritance, polymorphism, and exception handling
6. Programmatically connect to a database and implement the database operations
7. Rigorous testing of software application
8. Work collaboratively as part of a small team.

Referencing Style

- [American Psychological Association 6th Edition (APA 6th edition)](#)
- [Harvard (author-date)](#)

Submission
Online

Submission Instructions
Submit one zip file containing the source code files (.java) per group and the individual report file (.doc)by all. Do not
submit the zipped project folder or compiled binaries(.class or .jar).

Learning Outcomes Assessed

- Design classes which use inheritance, polymorphism, and exception handling
- Analyse, design and implement multi-layered software solutions, focusing on data structures and algorithms
- Compare and contrast performance of different algorithms in problem solving
- Investigate socially innovative practices in software development

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).
Calculator - all non-communicable calculators, including scientific, programmable and graphics calculators are authorised

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