COIT20256 *Data Structures and Algorithms*
Term 1 - 2019

Profile information current as at 08/05/2024 02:49 pm

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 build classes, throw exceptions, and extend a class using inheritance and polymorphism. You will practice these concepts and build applications with front-end Graphical User Interface and back-end database using database programming. You will also learn to build software applications using built-in Java Application Programming Interfaces (APIs) for generic collections of linked lists, stacks, queues, sets and maps, and creating custom generic data structures. You will also gain understanding of sorting and searching algorithms, recursion, search trees, and learn to evaluate algorithms using Big O notation. You will be introduced to functional programming using Lambdas and Streams. You will be given hands-on experience to practice all concepts during computer lab tutorials. The object-oriented design will also be covered.

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 - 2019

- 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. **Written Assessment**
Weighting: 20%
2. **Written Assessment**
Weighting: 30%
3. **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 evaluation

Feedback

There have been some dissatisfaction about the clarity of assignment requirements.

Recommendation

The assignments are written based on authentic software development cases and this may cause some inherent vagueness. The students will be given clear instructions to design the solution with supporting assumptions taken by them.

Unit Learning Outcomes

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

1. Compare and contrast different algorithms in problem solving
2. Design and implement appropriate data structures for application development
3. Evaluate a variety of data structures and algorithmic approaches including: recursion, linked lists, stacks, queues, streams, search trees, sorting and searching
4. Analyse, develop and implement software solutions with the focus of data structures and algorithms
5. Apply classes, inheritance, polymorphism, and exception handling
6. Programmatically connect to a database and implement the database operations
7. Work collaboratively as part of a small team
8. Demonstrate socially innovative practices in software development
9. Accomplish functional programming with Lambda expressions and Streams.

Australian Computer Society (ACS) recognises the Skills Framework for the Information Age (SFIA). SFIA is in use in over 100 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 at
https://www.acs.org.au/professionalrecognition/mysfia-b2c.html
This unit contributes to the following workplace skills as defined by SFIA. The SFIA code is included:

- Systems design (DESN)
- System Integration (SINT)
- Program ming/Software Development (PROG)
- Data Analysis (DTAN)
- Database/Repository Design (DBDS)
- Testing (TEST)
- Applications Support (ASUP)

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	7	8	9

Assessment Tasks	Learning Outcomes								
	1	2	3	4	5	6	7	8	9
1 - Written Assessment - 20%		•		•	•			•	
2 - Written Assessment - 30%	•	•	•	•	•	•	•	•	
3 - Examination - 50%	•		•						•

Alignment of Graduate Attributes to Learning Outcomes

Graduate Attributes	Learning Outcomes								
	1	2	3	4	5	6	7	8	9
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 - Written Assessment - 20%	∘	∘	∘		∘			
2 - Written Assessment - 30%	∘	∘	∘	∘	∘	∘		
3 - Examination - 50%	∘	∘	∘					

Textbooks and Resources

Textbooks

COIT20256

Prescribed

Java how to program:Early Objects 11th edition (2017)
Edition: 11 (2017)
Authors: Paul Deiteil & Harvey Deitel
Pearson Higher Ed
Upper Saddle River , New Jersey , USA
ISBN: 9781292223858
Binding: Paperback

Additional Textbook Information
The examination is open book. Therefore, students can take their prescribed Textbook or any other Java Programming language book to the examination.
Copies can be purchased 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)
- Java Development Kit (JDK) 1.8 or later
- MySql Database Server version 5.7 or later
- NetBeans IDE 8 or higher

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 - 11 Mar 2019		
Module/Topic	**Chapter**	**Events and Submissions/Topic**
A Deeper Look at Classes and Objects	Chapter 8	
Week 2 - 18 Mar 2019		
Module/Topic	**Chapter**	**Events and Submissions/Topic**
Object-Oriented Programming: Inheritance	9	

Week 3 - 25 Mar 2019

Module/Topic	Chapter	Events and Submissions/Topic
Object-Oriented Programming: Polymorphism and Interfaces	10	

Week 4 - 01 Apr 2019

Module/Topic	Chapter	Events and Submissions/Topic
GUI Components	12 and 22	

Week 5 - 08 Apr 2019

Module/Topic	Chapter	Events and Submissions/Topic
Exception Handling, Files, Streams and Object Serialization	11 and 15	

Vacation Week - 15 Apr 2019

Module/Topic	Chapter	Events and Submissions/Topic

Week 6 - 22 Apr 2019

Module/Topic	Chapter	Events and Submissions/Topic
Object-Oriented Design	33 (online chapter): Object-Oriented Design with the UML	**Assignment 1** Due: Week 6 Friday (26 Apr 2019) 11:45 pm AEST

Week 7 - 29 Apr 2019

Module/Topic	Chapter	Events and Submissions/Topic
Generic Collections	16	

Week 8 - 06 May 2019

Module/Topic	Chapter	Events and Submissions/Topic
Algorithms, Algorithmic Efficiency, and Lambdas and Streams	19 and 17	

Week 9 - 13 May 2019

Module/Topic	Chapter	Events and Submissions/Topic
Accessing Database with JDBC	24	

Week 10 - 20 May 2019

Module/Topic	Chapter	Events and Submissions/Topic
Generic Classes and Methods	20	

Week 11 - 27 May 2019

Module/Topic	Chapter	Events and Submissions/Topic
Recursion	18	**Assignment 2** Due: Week 11 Thursday (30 May 2019) 11:45 pm AEST

Week 12 - 03 Jun 2019

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

Review/Exam Week - 10 Jun 2019

Module/Topic	Chapter	Events and Submissions/Topic

Exam Week - 17 Jun 2019

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
Written 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 Friday (26 Apr 2019) 11:45 pm AEST

Return Date to Students
Week 8 Friday (10 May 2019)

Weighting
20%

Assessment Criteria
1. Efficient object-oriented programming design
2. Appropriate use of Graphical User Interface (GUI)
3. Appropriate use of data structures and algorithms in problem solving
4. Appropriate use of object-oriented concepts of classes, inheritance, and polymorphism
5. Development of fault-tolerant applications by appropriate use of exception handling
6. Effective use of good coding practices
7. 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 and implement appropriate data structures for application development
- Analyse, develop and implement software solutions with the focus of data structures and algorithms
- Apply classes, inheritance, polymorphism, and exception handling
- Demonstrate socially innovative practices in software development

Graduate Attributes

- Knowledge
- Communication
- Cognitive, technical and creative skills

- Self-management

2 Assignment 2

Assessment Type
Written 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 (30 May 2019) 11:45 pm AEST

Return Date to Students
Review/Exam Week Tuesday (11 June 2019)

Weighting
30%

Assessment Criteria

1. Efficient object-oriented program design.
2. Effective use of Generic data structures and algorithms in problem solving
3. Effective evaluation of a variety of data structures and algorithmic approaches
4. Effective use of good programming practice/techniques.
5. Efficient database programming techniques.
6. Rigorous testing of software application

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

- Compare and contrast different algorithms in problem solving
- Design and implement appropriate data structures for application development
- Evaluate a variety of data structures and algorithmic approaches including: recursion, linked lists, stacks, queues, streams, search trees, sorting and searching
- Analyse, develop and implement software solutions with the focus of data structures and algorithms
- Apply classes, inheritance, polymorphism, and exception handling
- Programmatically connect to a database and implement the database operations
- Work collaboratively as part of a small team
- Demonstrate 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).

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