COIT11222 *Programming Fundamentals*
Term 3 - 2025

Profile information current as at 12/11/2026 11:52 am

All details in this unit profile for COIT11222 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 will help you become a programmer even if you have had no programming experience. You will learn and practise topics such as pseudocode, variables, constants, data types, operators, expressions, statements, classes, objects, control constructs, methods, passing parameters and arrays. In addition, you will learn how to design, implement and test programs using a modern Integrated Development Environment (IDE).

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
There are no requisites for this unit.
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 - 2025
- 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 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. Class/Tutorial Activities (Practical Assessment)
Weighting: 30%
2. Assessment 2: Programming Assessment (Practical Assessment)
Weighting: 20%
3. Examination (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
- Academic Progression Policy and Procedure
- Student Refund and Credit Balance Policy and Procedure
- 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 feedback

Feedback
Explanation of programming concepts were simple, clear, and the lecturer was extremely helpful and friendly.

Recommendation
Continue to explain concepts in a clear, approachable, and student-friendly manner.

Feedback from Student feedback

Feedback
The handwriting tool used during lectures was sometimes illegible; clearer demonstrations and more real-world examples were requested.

Recommendation
Consider replacing handwriting with programming demonstrations or step-by-step slides, and incorporate more real-world examples to improve clarity and application.

Unit Learning Outcomes

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

1. Develop clearly documented and thoroughly tested applications using an industry-standard integrated development environment (IDE)
2. Apply procedural concepts (methods, iteration, selection) and design principles (encapsulation, coupling and cohesion) to the realisation of object behaviour in applications
3. Implement standard algorithms such as searching, sorting and sequential processing for arrays and lists of objects in applications
4. Employ stream abstraction to process records contained in sequential text files
5. Apply concepts presented in this unit, including language syntax, memory models, execution models, types, scope rules, methods, parameter passing, classes, objects and algorithms.

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 9:

- Programming/Software Development (PROG)
- Functional Testing (TEST)

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
1 - Practical Assessment - 30%	•	•	•		•
2 - Practical Assessment - 20%	•	•	•	•	•
3 - Examination - 50%				•	•

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

COIT11222

Prescribed

How to Think Like a Computer Scientist: Learning with Python 3
Authors: Peter Wentworth, Jeffrey Elkner, Allen B. Downey and Chris Meyers
ISBN: 9888406787
Binding: eBook

IT Resources

You will need access to the following IT resources:

- CQUniversity Student Email
- Internet
- Unit Website (Moodle)
- Textpad 7 or higher
- Visual Studio Code (latest version)
- python 3.10 or higher
- Java 21
- Eclipse IDE for java
- Anaconda 2019.10 or Higher

Referencing Style

All submissions for this unit must use the referencing style: Harvard (author-date)
For further information, see the Assessment Tasks.

Teaching Contacts

Olatunji Omisore Unit Coordinator
o.omisore@cqu.edu.au

Schedule

Week 1 - 10 Nov 2025

Module/Topic	Chapter	Events and Submissions/Topic
Understanding programming and creating simple python programs	Python textbook Chapter 1 and online resources	Installations of dependencies

Week 2 - 17 Nov 2025

Module/Topic	Chapter	Events and Submissions/Topic
Using Data (in Python)	Python textbook, Chapter 2	

Week 3 - 24 Nov 2025

Module/Topic	Chapter	Events and Submissions/Topic
Making decisions	Python textbook Chapter 5 and online resources	Assessment item 1: part 1—due in week 3 tutorial class

Week 4 - 01 Dec 2025

Module/Topic	Chapter	Events and Submissions/Topic
Functions	Python textbook Chapter 4 and online resources	Assessment item 1: part 2—due in week 4 tutorial class

Week 5 - 08 Dec 2025

Module/Topic	Chapter	Events and Submissions/Topic

Loops	Python textbook Chapter 7 and online resources	Assessment item 1: part 3—due in week 5 tutorial class

Week 6 - 15 Dec 2025

Module/Topic	Chapter	Events and Submissions/Topic
Classes and Objects	Python textbook Chapter 15 and online resources	Assessment item 1: part 4—due in week 6 tutorial class

Vacation Week - 22 Dec 2025

Module/Topic	Chapter	Events and Submissions/Topic

Vacation Week - 29 Dec 2025

Module/Topic	Chapter	Events and Submissions/Topic

Week 7 - 05 Jan 2026

Module/Topic	Chapter	Events and Submissions/Topic
Lists and Strings	Chapter 8 and online resources	Assessment item 1: part 5- due in week 7 tutorial class

Week 8 - 12 Jan 2026

Module/Topic	Chapter	Events and Submissions/Topic
Working with files	Python textbook chapter 13	Assessment item 1: part 6 is due in week 8 tutorial class

Week 9 - 19 Jan 2026

Module/Topic	Chapter	Events and Submissions/Topic
Testing, error handling, and debugging.	Python textbook chapter 12	Assessment item 1: part 7- due in week 9 tutorial class

Week 10 - 26 Jan 2026

Module/Topic	Chapter	Events and Submissions/Topic
Python advanced concepts	Online resources	Assessment item 1: part 8—due in week 10 tutorial class Assessment Item 2: due Friday11:55pm (AEST) Assessment 2: Programming Assessment Due: Week 10 Friday (30 Jan 2026) 11:45 am AEST

Week 11 - 02 Feb 2026

Module/Topic	Chapter	Events and Submissions/Topic
Introduction to Programming in Java	Online resources	

Week 12 - 09 Feb 2026

Module/Topic	Chapter	Events and Submissions/Topic
Introduction to Design; Ethical Considerations; Review	Online resources	

Exam Week - 16 Feb 2026

Module/Topic	Chapter	Events and Submissions/Topic

Term Specific Information

Unit Coordinator: Dr Olatunji Omisore
o.omisore@cqu.edu.au

Assessment Tasks

1 Practical Assessment

Assessment Title

Class/Tutorial Activities

Task Description
This assessment item is to be developed and submitted as part of your weekly tutorial/practical sessions.
Note:

1. This assessment is exempted from the 72-hour submission grace period and must be completed by the stated submission date/time. Thus,
    - For on-campus students each weekly tutorial must be submitted in your tutorial.
    - For distance education students each weekly tutorial must be submitted by the due date and time specified for the weekly tutorial in Moodle.
2. Re-attempts are not allowed for this assessment item.

A. For on-campus students
This assessment item will be based on weekly tutorial/practical work in weeks 3 -11 (inclusive). The assessment will be based on your performance/engagement in the class as well as the sample of work your tutor will ask you to submit during the class. However, note that the class is a tutorial/practical class, not a test. It is a normal tutorial where your tutor will guide you through the activities and help you if you have any questions. It is important to prepare for the class by attending and/or viewing the week's lecture content before your tutorial/practical class. For on-campus students (and offshore students), the submission of the sample of work is due in the weekly tutorial/practical class, NOT the due date shown in Moodle. The date given in Moodle is to cater for all scheduled workshop classes.
Note:
1. Unless there are special circumstances (approved by the unit coordinator), this assessment task must be developed and submitted on Moodle in your weekly tutorial/practical class. If you have special circumstances that prevent you from attending a specific class, please contact your tutor and unit coordinator as soon as you are aware of an issue.
2. Marks may be deducted if your tutor is not satisfied with your progress or understanding of the work. Marks are not only awarded for the sample of work submitted during the class. Satisfactory participation in all tutorial/practical class activities is also required to be awarded marks for this assessment item.
B. Distance Education students
You are highly encouraged to attend your online tutorial sessions to support your understanding of the assessment and overall learning. However, attendance is not mandatory. The unit coordinator will communicate with you more about this assessment item at the start of the term. However, note the following:

1. You will be required to submit the sample of your weekly work requested by your tutor by the due date shown on the unit website for the corresponding week. Late submissions will normally be awarded 0 marks.
2. All the worksheet questions are important to help your understanding of the work covered each week, so you should attempt all questions. However, just as is the case for on-campus students, you will not be required to submit all the questions in the weekly worksheet. You will be told which questions are to be submitted as a sample of your work each week.
3. As with on-campus students, it is a tutorial, not a test. If you are having any difficulty with the tutorial/practical work please do not hesitate to get in touch with the unit coordinator.
4. The unit coordinator is normally the tutor for distance education students.
5. The unit coordinator may make arrangements to discuss aspects of the weekly work with you before marks can be awarded.
6. If you have any special circumstances that may impact your ability to complete the weekly assessment task please contact the unit coordinator to discuss this as soon as you are aware of any issue.

AI ASSESSMENT SCALE - AI PLANNING
You may use AI for planning, idea development, and research. Your final submission should show how you have developed and refined these ideas.
Note: For programming assignments you can get examples and ideas from AI, but you must not include code in your program that was generated by AI. You must write the code in your application yourself. If asked, you must be able to explain any of the code submitted and you must be capable of writing similar code under invigilated, test conditions if required to do so. In addition, you must only use the language features and techniques covered in the unit (and course) so far.

Assessment Due Date
Due in your weekly tutorial/practical class

Return Date to Students
Within 2 weeks of the due date or within 2 weeks of submission (whichever is the later)

Weighting
30%

Assessment Criteria
Tasks in the weekly tutorial/practical will test your understanding of the topics covered in that week.
On-campus students: You are required to participate in all activities in the tutorial and submit the sample of work requested when you are instructed to do so during the class. Your tutor will monitor your progress in class. Be prepared to ask questions if anything is unclear and to explain your work to your tutor in class.
Distance Education students: You will also be required to submit a specified sample of work each week and you may be asked to meet with your tutor to discuss your work (e.g. over Zoom). The tutor will communicate with you further about

this assessment item.

Referencing Style
Harvard (author-date)

Submission
Online

Learning Outcomes Assessed
- Develop clearly documented and thoroughly tested applications using an industry-standard integrated development environment (IDE)
- Apply procedural concepts (methods, iteration, selection) and design principles (encapsulation, coupling and cohesion) to the realisation of object behaviour in applications
- Implement standard algorithms such as searching, sorting and sequential processing for arrays and lists of objects in applications
- Apply concepts presented in this unit, including language syntax, memory models, execution models, types, scope rules, methods, parameter passing, classes, objects and algorithms.

2 Practical Assessment

Assessment Title
Assessment 2: Programming Assessment

Task Description
This assessment requires the development of programming applications.
Importantly, this assessment is intended to assess your ability to apply the programming concepts studied at class for developing, testing, and documenting software applications. The assessments require you to apply the concepts about variables, constants, data types, arithmetic expressions, selection statements, repetition statements, input/output from/to the standard input/output devices, classes, and compound data types, searching, sorting, exceptions, *etc.* as covered in the unit. You must only use the language features and techniques covered in the unit (and course) so far.
Further details about the requirements for software application can be found in their detailed specification document on the unit website.
AI ASSESSMENT SCALE - AI PLANNING
You may use AI for planning, idea development, and research. Your final submission should show how you have developed and refined these ideas.
Notes:
You can get examples and ideas from AI for your programming assignments. *We may run sessions on working without GenAI, and assistive/collaborative/wholly code development with GenAI.* Thus, you are required to indicate how GenAI is used e.g. reference, documentation, code versioning, etc, in all your submissions.
Importantly, assessment will be not be based on final codes where AI is wholly/majorly used to develop a program or application. Therefore, you are required to write the code in your assessment tasks by yourself. You must be able to explain any of the code submitted when asked, and you must be capable of writing similar code under invigilated, test conditions when required to do so.

Assessment Due Date
Week 10 Friday (30 Jan 2026) 11:45 am AEST

Return Date to Students
Within 2 weeks of the due date or within 2 weeks of submission (whichever is the later)

Weighting
20%

Assessment Criteria
The submissions will be assessed based on the following criteria:
- Achieves the requirements outlined in the specification (including functionality and design specified)
- Appropriate use of programming constructs/concepts
- Appropriate use of objects, classes, and methods
- Good programming practices (see Coding Guidelines document on the website)
- Compilation and execution of the program using a modern IDE
- Appropriate and complete code documentation
- Appropriate test planning and thorough testing of the code

A more detailed breakdown of the assessment criteria can be found in the specifications on the Moodle unit website.

Referencing Style
Harvard (author-date)

Submission
Online

Learning Outcomes Assessed

- Develop clearly documented and thoroughly tested applications using an industry-standard integrated development environment (IDE)
- Apply procedural concepts (methods, iteration, selection) and design principles (encapsulation, coupling and cohesion) to the realisation of object behaviour in applications
- Implement standard algorithms such as searching, sorting and sequential processing for arrays and lists of objects in applications
- Employ stream abstraction to process records contained in sequential text files
- Apply concepts presented in this unit, including language syntax, memory models, execution models, types, scope rules, methods, parameter passing, classes, objects and algorithms.

Examination

Outline
Complete an invigilated examination.

Date
During the examination period at a CQUniversity examination centre.

Weighting
50%

Length
180 minutes

Minimum mark or grade
40%

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