COIT11222 *Programming Fundamentals*
Term 2 - 2024

Profile information current as at 02/07/2025 11:02 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 2 - 2024

- Brisbane
- Cairns
- 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 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 Assessment**
Weighting: 35%
2. **Practical Assessment**
Weighting: 15%
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 Teacher Evaluation (Term 2, 2024)

Feedback
The lecture content lacks sufficient interactive elements, potentially limiting student engagement and participation.

Recommendation
To make the class more interactive, consider incorporating more real-world examples to create a more engaging learning environment.

Feedback from Teacher Evaluation (Term 2, 2024).

Feedback
Students found the current number of examples illustrating Python custom classes to be inadequate for a comprehensive understanding of the concept.

Recommendation
Increase the number of examples illustrating Python custom classes in the lecture to enhance student comprehension and application skills.

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 8:

- Programming/Software Development (PROG)
- 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 - 35%	•	•	•		•
2 - Practical Assessment - 15%	•	•	•	•	•
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

Referencing Style

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

Teaching Contacts

Umair Ullah Tariq Unit Coordinator
u.tariq@cqu.edu.au

Schedule

Week 1 - 08 Jul 2024

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

Week 2 - 15 Jul 2024

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

Week 3 - 22 Jul 2024

Module/Topic	Chapter	Events and Submissions/Topic
Making decisions	Python textbook Chapter 5 and online resources	**Assessment item 2: part 1** - due in week 3 tutorial class

Week 4 - 29 Jul 2024

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

Week 5 - 05 Aug 2024

Module/Topic	Chapter	Events and Submissions/Topic
Looping	Python textbook Chapter 7 and online resources	**Assessment item 2: part 3** - due in week 5 tutorial class

Vacation Week - 12 Aug 2024

Module/Topic	Chapter	Events and Submissions/Topic

Week 6 - 19 Aug 2024

Module/Topic	Chapter	Events and Submissions/Topic
Classes and Objects	Python textbook Chapter 15 and online resources	**Assessment item 2: part 4** - due in week 6 tutorial class

Week 7 - 26 Aug 2024

Module/Topic	Chapter	Events and Submissions/Topic
Lists and Strings	Chapter 8 and online resources	**Assessment Item 1: Part A (15%) due** Friday 11:55pm (AEST) **Assessment item 2: part 5**- due in week 7 tutorial class

Week 8 - 02 Sep 2024

Module/Topic	Chapter	Events and Submissions/Topic
Working with files	Python textbook chapter 13	**Assessment item 2: part 6**- due in week 8 tutorial class

Week 9 - 09 Sep 2024

Module/Topic	Chapter	Events and Submissions/Topic
Testing, Error handling, and debugging.		**Assessment item 2: part 7**- due in week 9 tutorial class

Week 10 - 16 Sep 2024

Module/Topic	Chapter	Events and Submissions/Topic
Introduction to Programming in Java Part 1 (primitive types, reference types, and decision making)		**Assessment item 2: part 8** - due in week 10 tutorial class

Week 11 - 23 Sep 2024

Module/Topic	Chapter	Events and Submissions/Topic
Introduction to Programming in Java Part 2 (Looping, Methods, and Arrays)		**Assessment Item 1: Part B (20%) due** Friday11:55pm (AEST) **Assessment item 2: part 9** - due in week 11 tutorial class

Week 12 - 30 Sep 2024

Module/Topic	Chapter	Events and Submissions/Topic
Introduction to Design; Ethical Considerations; Review	Online resources	**Assessment item 2: part 10** - due in week 12 tutorial class

Review/Exam Week - 07 Oct 2024

Module/Topic	Chapter	Events and Submissions/Topic

Exam Week - 14 Oct 2024

Module/Topic	Chapter	Events and Submissions/Topic
		Examination (50%) - scheduled during examination block

Term Specific Information

Contact information for Dr. Umair Ullah Tariq:
Email: u.tariq@cqu.edu.au
Office: Level 2 Room 2.08, 400 Kent Street, Sydney NSW 2000
P +61 2 9324 5096
If you have any queries, please email me and I will get back to you within one to two business days. For an individual discussion, please email to discuss a time for a meeting.

Assessment Tasks

1 Assessment 1: Programming Assessment

Assessment Type
Practical Assessment

Task Description
This assessment requires the development of programming applications. It is intended to assess your ability to apply the programming concepts studied to the development, testing, and documentation of software applications. The applications are called Assessment Item 1: Part A and Assessment Item 1: Part B.

1. **Part A (15%):** requires you to apply the concepts covered in the first part of the term (variables, constants, data types, arithmetic expressions, selection statements, repetition statements, input/output from/to the standard input/output devices, classes and compound data types). It is **due at the end of week 7 (Friday).**
2. **Part B (20%):** requires you to apply the concepts covered in the first part of the term as well as concepts covered in the second part of the term (e.g. looping, compound data types, searching, sorting, exceptions, etc). It is **due at the end of week 11 (Friday).**

Further details about the requirements for both software applications can be found in their detailed specification documents on the unit website.

Assessment Due Date

Part A is due on Friday of week 7. Part B is due on Friday of week 11.

Return Date to Students

2 weeks after the due date.

Weighting
35%

Assessment Criteria
The submissions will be assessed based on the following criteria

1. Achieves the requirements outlined in the specification (including functionality **and design specified**) (Part A and B)
2. Appropriate use of programming constructs/concepts: variables, constants, types, operators, expressions, selection statements, standard input, and output), compound data types (Part A and B); Exceptions, searching, sorting, and working with files (Part B)
3. Appropriate use of objects, classes, and methods (Part A and B)
4. Good programming practices (see Coding Guidelines document on the website) (Part A and B)
5. Compilation and execution of the program using a modern IDE (Part A and B)
6. Appropriate and complete code documentation (Part A and B)
7. Appropriate test planning and thorough testing of the code (Part A and B)

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
- Apply concepts presented in this unit, including language syntax, memory models, execution models, types, scope rules, methods, parameter passing, classes, objects and algorithms.

2 Assessment 2: Class/Tutorial Activities

Assessment Type
Practical Assessment

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

For on-campus students

This assessment item will be based on weekly tutorial/practical work in weeks 3 -12 (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.

Distance Education students

As distance education students you are not required to attend a tutorial/practical class. 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 (and check all the solutions when they become available). 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.

Each week (weeks 3 - 10) is worth 1.5% of your overall mark for the unit. (This may appear as a mark out of 3 in the assessment area in Moodle. Moodle does not allow fractional maximum assessment item marks. In that case, the final marks in this assessment item will be divided by 2 when included in the calculation for your final grade.)

Assessment Due Date

Due in your weekly tutorial/practical class from weeks 3 -12 (inclusive).

Return Date to Students

Marks will be returned before your next tutorial submission is due.

Weighting
15%

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 or the unit coordinator to discuss your work (e.g. over Zoom). The unit coordinator will communicate with you further about this assessment item.
Each week is worth 1.5 marks.

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
50%

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