



COIT20245 *Introduction to Programming*

Term 1 - 2017

Profile information current as at 27/04/2024 08:34 am

All details in this unit profile for COIT20245 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 is designed for students who have had little or no programming experience. The unit aims to teach students principles, design and development of object-oriented programs. It covers topics such as Integrated Development Environment (IDE), variables, fields, constants, data types, operators, expressions, loops, classes, objects, methods, arrays, file processing, graphical user interfaces and the principles of human-computer interaction. Students will learn how to design, implement and test programs using a modern IDE. Note: If you have completed unit COIT29222 then you cannot take this unit.

Details

Career Level: *Postgraduate*

Unit Level: *Level 8*

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 1 - 2017

- Brisbane
- Distance
- Melbourne
- 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 and Written Assessment**

Weighting: 20%

2. **Practical and 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 Last year's ACER

Feedback

This course is less focus on GUI . it provide less information of graphical things such as button, jpanel, etc. for example, this course have not provide any information like how to code behind buttons. Only teaches little bit about JOptionPane.

Recommendation

In the future offer, the topics covering on human computer interaction could be merged into one week from current two weeks. thus one week lecture topic on building a simple GUI could be added.

Action

HCI is covered in multiple units, so a broader discussion has been initiated prior to any change being made.

Feedback from have your say

Feedback

More practical examples and more resources for FLEX and novice programmers

Recommendation

Create practical tutorial videos which demonstrate coding examples of the weekly topics

Action

As part of the moodle site restructuring, all resources and examples were reviewed. Some were deleted and many were modified. Also new resources (e.g. a NetBeans FAQ), and code examples were added.

Feedback from have your say, hearsay and self reflection

Feedback

Some students believe the course is too easy to pass without learning the fundamentals of programming

Recommendation

Place a hurdle of 40-45% on the exam or reduce the assignment marks and increase the exam marks

Action

No change was made to assessment weightings / pass marks. Rather, the exam questions were modified to reflect a stronger focus on fundamental programming concepts such as scope.

Unit Learning Outcomes

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

1. Explain the principles of object-oriented programming.
2. Demonstrate the use of an integrated development environment (IDE).
3. Develop programs using various data types, operators, expressions, loops, classes, objects and methods.
4. Develop programs using arrays, files and streams for storing, searching and sorting data.
5. Explain the principles of human-computer interaction.
6. Design, analyse and apply techniques to produce quality programs.

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 Integration (SINT)
- Program ming/Software Development (PROG)
- Data Analysis (DTAN)
- Testing (TEST)
- Applications Support (ASUP)

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 and Written Assessment - 20%	•	•	•			•
2 - Practical and Written Assessment - 30%	•	•	•	•		•
3 - 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						

Graduate Attributes	Learning Outcomes					
	1	2	3	4	5	6
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 and Written Assessment - 20%	○		○		○			
2 - Practical and Written Assessment - 30%	○	○	○		○			
3 - Examination - 50%	○		○					

Textbooks and Resources

Textbooks

COIT20245

Prescribed

Java How to Program : Early Objects Edition

Edition: 10th edn Global (2014)

Authors: Paul Deitel and Harvey Deitel

Pearson Education

Upper Saddle River , NJ , USA

ISBN: 9781292018195

Binding: Paperback

Additional Textbook Information

The final examination is open book where students can take the Textbook with them for the examination. Therefore, it is better for students to get a hard copy of the Textbook even though the rest of the study can be managed with an e-Book. Paper copies can be purchased through the CQUni Bookshop here: <http://bookshop.cqu.edu.au>

[View textbooks at the CQUniversity Bookshop](http://bookshop.cqu.edu.au)

IT Resources

You will need access to the following IT resources:

- CQUniversity Student Email
- Internet
- Unit Website (Moodle)
- JDK, download from <http://www.oracle.com/technetwork/java/javase/downloads/index.html>
- NetBeans IDE, download from <http://www.oracle.com/technetwork/java/javase/downloads/index.html>
- Textpad, download from <https://www.textpad.com/download/index.html>

Referencing Style

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

Teaching Contacts

Dennis Jarvis Unit Coordinator
d.jarvis@cqu.edu.au

Schedule

Week 1 - 06 Mar 2017

Module/Topic	Chapter	Events and Submissions/Topic
Introduction to Object Technology and Java	Chapter 1	

Week 2 - 13 Mar 2017

Module/Topic	Chapter	Events and Submissions/Topic
Variables, Types and Operators	Chapter 2	

Week 3 - 20 Mar 2017

Module/Topic	Chapter	Events and Submissions/Topic
Classes, Objects and Methods	Chapter 3	

Week 4 - 27 Mar 2017

Module/Topic	Chapter	Events and Submissions/Topic
Problem Solving and Control Statements	Chapter 4	

Week 5 - 03 Apr 2017

Module/Topic	Chapter	Events and Submissions/Topic
Control Statements and Logical Operators	Chapter 5	

Vacation Week - 10 Apr 2017

Module/Topic	Chapter	Events and Submissions/Topic

Week 6 - 17 Apr 2017

Module/Topic	Chapter	Events and Submissions/Topic
Program Design: Modules and Reusability	Chapter 6	Assignment 1 Due: Week 6 Friday (21 Apr 2017) 11:45 pm AEST

Week 7 - 24 Apr 2017

Module/Topic	Chapter	Events and Submissions/Topic
Arrays and ArrayLists	Chapter 7	

Week 8 - 01 May 2017

Module/Topic	Chapter	Events and Submissions/Topic
Array Searching and Sorting	Chapter 19	

Week 9 - 08 May 2017

Module/Topic	Chapter	Events and Submissions/Topic
Human Computer Interaction: Usability of Interactive systems, Guidelines and Principles	Lecture Notes Week 9	

Week 10 - 15 May 2017

Module/Topic	Chapter	Events and Submissions/Topic
Input Data Validation: Characters, Strings and Regular Expressions	Chapter 16	

Week 11 - 22 May 2017

Module/Topic	Chapter	Events and Submissions/Topic
Files and Streams	Chapter 17	Assignment 2 Due: Week 11 Friday (26 May 2017) 11:45 pm AEST

Week 12 - 29 May 2017

Module/Topic	Chapter	Events and Submissions/Topic
Managing and Evaluating Interface Designs	Lecture Notes Week 12	

Review/Exam Week - 05 Jun 2017

Module/Topic	Chapter	Events and Submissions/Topic
--------------	---------	------------------------------

Exam Week - 12 Jun 2017

Module/Topic	Chapter	Events and Submissions/Topic
--------------	---------	------------------------------

Assessment Tasks

1 Assignment 1

Assessment Type

Practical and Written Assessment

Task Description

This assessment item is designed to test your understanding of topics such as variables, constants, types, operators, standard input/output, loops, if statements, classes, UML design, objects and methods. The assessment task is to design, write, compile and execute a java program using the above mentioned topics. Further details will be available on the unit website.

Assessment Due Date

Week 6 Friday (21 Apr 2017) 11:45 pm AEST

Return Date to Students

Two weeks after due date

Weighting

20%

Assessment Criteria

1. Efficient object-oriented program design.
2. Appropriate use of variables, constants, types, operators, expressions, statements and loops.
3. Appropriate use of objects, classes and methods.
4. Effective use of good programming practice/techniques.
5. Rigorous testing of the program for logic, runtime and other errors.
6. Compilation and execution of the program using a modern IDE.

Referencing Style

- [Harvard \(author-date\)](#)

Submission

Online

Learning Outcomes Assessed

- Explain the principles of object-oriented programming.

- Demonstrate the use of an integrated development environment (IDE).
- Develop programs using various data types, operators, expressions, loops, classes, objects and methods.
- Design, analyse and apply techniques to produce quality programs.

Graduate Attributes

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

2 Assignment 2

Assessment Type

Practical and Written Assessment

Task Description

This assessment item is designed to test your understanding of topics such as array/arrayLists, methods with parameters, sorting, searching and input validation. The assessment task is to design, write, compile, test, and execute a java program using the above mentioned topics. Further details will be available on the unit website.

Assessment Due Date

Week 11 Friday (26 May 2017) 11:45 pm AEST

Return Date to Students

Two weeks after due date

Weighting

30%

Assessment Criteria

1. Efficient object-oriented program design.
2. Appropriate use of variables, constants, types, operators, expressions, statements and loops.
3. Appropriate use of classes, objects, and methods.
4. Effective use of good programming practice/techniques.
5. Rigorous testing of the program for logic and runtime errors, data validation and reuse of code.
6. Compilation and execution of the program using a modern IDE.
7. Efficient use of arrays/arrayLists, searching algorithms and sorting algorithms.

Referencing Style

- [Harvard \(author-date\)](#)

Submission

Online

Learning Outcomes Assessed

- Explain the principles of object-oriented programming.
- Demonstrate the use of an integrated development environment (IDE).
- Develop programs using various data types, operators, expressions, loops, classes, objects and methods.
- Develop programs using arrays, files and streams for storing, searching and sorting data.
- Design, analyse and apply techniques to produce quality programs.

Graduate Attributes

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

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

No calculators permitted

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