MONASH University
Information Technology

Unit Guide

FIT1008
Introduction to computer science

Semester 2, 2017

The information contained in this unit guide is correct at time of publication. The University has the right to change any of the elements contained in this document at any time.

Last updated: 03 Jul 2017

Status: Approved

Table of contents

Unit handbook information

Synopsis

This unit introduces students to core problem-solving, analytical skills, and methodologies useful for developing flexible, robust, and maintainable software. In doing this it covers a range of conceptual levels, from high-level algorithms and data-structures, down to the machine models and simple assembly language programming. Topics include data types; data structures; algorithms; algorithmic complexity; recursion; and translation to assembly language.

Mode of delivery

Clayton (Day)
Malaysia (Day)

Workload requirements

Minimum total expected workload equals 12 hours per week comprising:

(a.) Contact hours for on-campus students:
- Three hours of lectures
- One 1-hour tutorial
- One 3-hour laboratory

(b.) Additional requirements (all students):
- A minimum of 5 hours of personal study per week in order to satisfy the reading and assignment expectations.

Unit relationships

Prerequisites

((One of FIT1040, ECE2071, FIT1002) and FIT1029) or FIT1045
Students beginning FIT1008 are assumed to be able to: Identify the main components of an algorithm (variables, operators, expressions, etc), and write the algorithm associated to the specification of a simple problem. Be able to translate a simple algorithm into a program containing variable declarations, selection, repetition, and lists and/or arrays.

Prohibitions

CSE1303, CSC1030, FIT1015, FIT2085

Co-requisites

None

Chief Examiner

Dr Julian Garcia

Campus Lecturer(s)

Clayton

Name: Dr Julian Garcia
Email: Julian.Garcia@monash.edu
Consultation hours: available via Moodle

Name: Mr Phillip Abramson
Email: Phillip.Abramson@monash.edu
Consultation hours: available via Moodle

Malaysia

Name: Dr Muhammad Fermi Pasha
Email: Muhammad.FermiPasha@monash.edu
Consultation hours: will be available via Moodle

Academic overview

Learning outcomes

At the completion of this unit, students should be able to:

1. translate simple problem statements into algorithms, implement them in a high level programming language and test them;
2. summarise and compare the properties of basic abstract data types such as stacks, queues, lists, trees, priority queues, heaps and hash tables;
3. evaluate different algorithms and implementations of basic abstract data types;
4. analyse algorithms by determining their best case and worst case big O time complexity;
5. deconstruct simple high-level code into assembly code such as MIPS R2000.

Teaching approach

Lecture and/or tutorials or problem classes

This teaching and learning approach helps students first encounter the information at lectures, discuss and explore them at length during tutorials, tests them via quizzes, and enables them to practice in a hands-on environment during labs.

Peer assisted learning

We use peer assisted learning during the lectures by asking questions about the material and holding short discussions among peers. For this purpose we rely on the use of the MARS software, which may require students to use a device connected to the internet.

Assessment summary

Examination (2 hours plus 30 minutes reading and noting time): 60%; In-semester assessment: 40%

Assessment task	Value	Due date
Mid-semester Test (1 hour)	10%	Week 7 during lecture.
Interview Pracs	20%	Interview pracs are due on weeks 3 & 4, 8 & 9 and 11& 12.
Code review pracs	5%	Code review pracs due on weeks 1, 2, 5, 6 and 10
Student participation (Quiz)	5%	Every week
Examination 1	60%	To be advised

Unit schedule

Week	Activities	Assessment
0	Register for tutorials, pracs and lectures. Review basic Python if needed.	No formal assessment or activities are undertaken in week 0.
1	Lectures on Intro to the Unit, MIPS Architecture and MIPS simple programs. Revision Tutorial. Prac 1 on Python and Algorithmics.	Online quiz. Code review during the Prac.
2	Lectures on Decisions in MIPS, Arrays in MIPS and Iteration in MIPS. Tutorial on MIPS simple programs. Prac 2 on MIPS Architecture and MIPS simulator.	Online quiz. Code review during the Prac.
3	Lectures on MIPS Memory and MIPS Functions. Tutorial on week 2 material. Prac 3 on MIPS.	Online quiz. Prac 3 checkpoint.
4	Lectures on Complexity, Sorting, Exceptions and Assertions. Tutorial on week 3 material. Prac 3 on MIPS.	Online quiz. Prac 3 assessment.
5	Lectures on Abstract Data Types; Classes and Objects and Variables and Scoping. Tutorial on week 4 material & Prac 4 on Complexity.	Online quiz. Code review during the Prac.
6	Lectures on Array-based Lists, Sorted Lists and Stack Arrays. Tutorial on week 5 material & Prac 5 on Classes, Objects and Testing.	Online quiz. Code review during the Prac.
7	Lectures on Array-based Queues and Linked Stacks. Tutorial on week 6 material. No Pracs this week.	Online quiz. **Mid-Semester Test.** No prac this week due to test.
8	Lectures on Linked Queues, Linked List, Iterators. Tutorial on week 7 material. Prac 6 on Containers.	Online quiz. Prac 6 checkpoint.
9	Lectures on Hashing, Collision Resolution I and II. Tutorial on week 8 material. Prac 6 on Containers Assessment.	Online quiz. Prac 6 to be assessed.
10	Lectures on Recursion, Recursive Sorts and Complexity, and Recursion vs Iteration. Tutorial on week 9 material. Prac 7 on Hash Tables.	Online quiz. Prac 7 checkpoint.
11	Lectures on Binary Trees and Binary Search Trees. Tutorial on week 8 material. Prac 7 on Hash Tables.	Online quiz. Prac 7 to be assessed.
12	Lectures on Heaps and Priority Queues. Tutorial on week 11 material & Prac 8 on Binary Trees.	Online quiz. Code review during the Prac.

	SWOT VAC	No formal assessment is undertaken in SWOT VAC
	Examination period	LINK to Assessment Policy:http://policy.monash.edu.au/policy-bank/academic/education/assessment/assessment-in-coursework-policy.html

Assessment requirements

Faculty Unit Assessment Hurdles Policy

To pass a unit which includes an examination as part of the assessment, a student must obtain, unless otherwise approved and published:

- 40% or more in the unit's examination, and
- 40% or more in the unit's total non-examination assessment, and
- an overall unit mark of 50% or more.

If a student does not achieve 40% or more in the unit examination or the unit non-examination total assessment, and the total mark for the unit is:

- equal to or greater than 50%, then a mark of 49-N will be recorded for the unit.
- less than 50% then the actual mark for the unit will be recorded.

Assessment tasks

Assessment title: Mid-semester Test (1 hour)

Learning outcomes: Learning outcome 1, 2, 3, 4 and 5.

Details of task: This test is performed during the class and covers exam-like questions for the first part of the course. It is intended to give students an idea of how they would perform in the final exam, given their current progress.

Value: 10%

Hurdle requirements (Where applicable): N/A

Individual Assessment in Group Tasks (Where applicable): N/A

Criteria for marking: This test will evaluate your understanding of the material provided during the first few weeks of semester, your capability to code simple algorithms given a clear specification,

and to analyse the behaviour and complexity of simple fragments of code. It will also test your ability to translate high level code into MIPS.

Due date: Week 7 during lecture.

Remarks (where applicable): N/A

Assessment title: Interview Pracs

Learning outcomes: Learning outcome 1, 2, 3, 4 and 5.

Details of task: There is a total of 8 pracs, 3 of which will be assessed with an interview. Each interview prac will span over two weeks, including a hurdle checkpoint due on the first week of each prac, and the interview due on the second week - check **due date** for assessment week. Prac assignments are long (they are designed to take about 4 hours). This means that you must have a significant proportion of the prac completed before attending the scheduled computer lab.

Value: 20%

Hurdle requirements (Where applicable): N/A

Individual Assessment in Group Tasks (Where applicable): N/A

Criteria for marking: Every prac sheet contains the assessment criteria used to assess that prac. In addition, demonstrators carry with them a marking guide prepared by the lecturer which indicates how exactly to mark each prac question. You can request the demonstrator to show you the marking guide after he/she has marked your prac. Marking rubrics will be provided for all interview pracs.

Due date: Interview pracs are due on weeks 3 & 4, 8 & 9 and 11& 12.

Remarks (where applicable): N/A

Assessment title: Student participation (Quiz)

Learning outcomes: Learning outcome 1, 2, 3, 4 and 5.

Details of task: Student participation will take place during lectures using peer discussion. Assessment will be carried out on a weekly basis via Moodle quizzes.

Value: 5%

Hurdle requirements (Where applicable): N/A

Individual Assessment in Group Tasks (Where applicable): N/A

Criteria for marking: A grade will be based on a student's quizzes via Moodle.

Due date: Every week

Remarks (where applicable): N/A

Assessment title: Code review pracs

Learning outcomes: Learning outcome 1, 2, 3, 4 and 5.

Details of task: The remaining 5 pracs will be peer assessed via code review. This review must happen in person at the end of each prac. Prac assignments are long (they are designed to take about 4 hours). This means that you must have a significant proportion of the prac completed before attending the scheduled computer lab.

Value: 5%

Hurdle requirements (Where applicable): N/A

Individual Assessment in Group Tasks (Where applicable): N/A

Criteria for marking: Evidence of critical thinking and reflection as assessed by demonstrator.

Due date: Code review pracs due on weeks 1, 2, 5, 6 and 10

Remarks (where applicable): N/A

Title : Examination 1
Value : 60%
Length : 2 hours
Type (open/closed book) : Closed book
Hurdle requirements (where applicable) : N/A
Electronic devices allowed : None
Learning outcomes assessed : Learning outcomes 1, 2, 3, 4 and 5.
Remarks (where applicable) : None

Extensions and penalties

Submission must be made by the due date otherwise penalties will be enforced.

You must negotiate any extensions formally with your campus unit lecturer via the in-semester special consideration process: http://www.monash.edu.au/exams/special-consideration.html

Late submissions are not allowed.

Returning assignments

Students can expect assignments to be returned within two weeks of the submission date or after receipt, whichever is later.

Referencing requirements

To build your skills in citing and referencing, and using different referencing styles, see the online tutorial Academic Integrity: Demystifying Citing and Referencing at http://www.lib.monash.edu/tutorials/citing/

Assignment submission

It is a University requirement (http://www.policy.monash.edu/policy-bank/academic/education/conduct/student-academic-integrity-managing-plagiarism-collusion-procedures.html) for students to submit an assignment coversheet for each assessment item. Faculty Assignment coversheets can be found at http://www.infotech.monash.edu.au/resources/student/forms/. Please check with your Lecturer on the submission method for your assignment coversheet (e.g. attach a file to the online assignment submission, hand-in a hard copy, or use an electronic submission). **Please note that it is your responsibility to retain copies of your assessments.**

Online submission: If Electronic Submission has been approved for your unit, please submit your work via the learning system for this unit, which you can access via links in the my.monash portal.

Please keep a copy of tasks completed for your records.

Feedback to you

☑ Informal feedback on progress in labs/tutes
☑ Graded assignments without comments
☑ Test results and feedback
☑ Other
Detailed solutions to tutes

Learning resources

READING LIST:

1) MIPS Assembly Language Programming
Author: Britton

2)Problem Solving with algorithms and data structures using Python
Author: Miller & Ranum
[also freely available online under CC license: https://interactivepython.org/runestone/static/pythonds/index.html]

and:

3) Data Structures & Algorithms in Python
Authors: Goodrich, Tamassia & Goldwasser

Monash Library Unit Reading List (if applicable to this unit):
http://readinglists.lib.monash.edu/index.html

Required resources

Students generally must be able to complete the requirements of their course without the imposition of fees that are additional to the student contribution amount or tuition fees. However, students may be charged certain incidental fees or be expected to make certain purchases to support their study. For more information about this, refer to the Higher Education Administrative Information for Providers, Chapter 18, Incidental Fees at http://education.gov.au/help-resources-providers.

Please check with your lecturer before purchasing any required resources. Limited copies of prescribed texts are available for you to borrow in the library, and prescribed software is available in student labs.

Your feedback to us

One of the formal ways students have to provide feedback on teaching and their learning experience is through the Student Evaluation of Teaching and Units (SETU) survey. The feedback is anonymous and provides the Faculty with evidence of aspects that students are satisfied with and areas for improvement.

Previous student evaluations of this unit

In response to previous SETU results of this unit, the following changes have been made:

Pracs will be assessed individually. Pracs are assessed every other week. Participation is marked based on quizzes online. Lectures involved a large component of coding live.

If you wish to view how previous students rated this unit, please go to;
http://www.monash.edu/ups/setu/setu-results/unit-evaluation-reports

Other information

Policies

Monash has educational policies, procedures and guidelines, which are designed to ensure that staff and students are aware of the University's academic standards, and to provide advice on how they might uphold them. You can find Monash's Education Policies at: http://www.policy.monash.edu/policy-bank/academic/education/index.html

Graduate Attributes Policy

http://www.monash.edu/policy-bank/academic/education/course-governance-and-design/course-design-policy

Student Charter

http://www.monash.edu/students/policies/student-charter.html

Student Services

The University provides many different kinds of services to help you gain the most from your studies. Contact your tutor if you need advice and see the range of services available at http://www.monash.edu/students.

For Malaysia see http://www.monash.edu.my/Student-services, and for South Africa see http://www.monash.ac.za/current/.

Monash University Library

The Monash University Library provides a range of services, resources and programs that enable you to save time and be more effective in your learning and research.
Go to http://www.monash.edu/library or the library tab in my.monash portal for more information.

At Malaysia visit the Library and Learning Commons at http://www.lib.monash.edu.my/.

At South Africa visit http://www.lib.monash.ac.za/.

Disability Support Services

Students who have a disability, ongoing medical or mental health condition are welcome to contact Disability Support Services.

Disability Support Services also support students who are carers of a person who is aged and frail or has a disability, medical condition or mental health condition.

Disability Advisers visit all Victorian campuses on a regular basis.

- Website: monash.edu/disability
- Telephone: 03 9905 5704 to book an appointment with an Adviser, or contact the Student Advisor, Student Community Services at 03 55146018 at Malaysia
- Email: disabilitysupportservices@monash.edu

- Drop In: Level 1, Western Annexe, 21 Chancellors Walk (Campus Centre) Clayton Campus, or Student Community Services Department, Level 2, Building 2, Monash University, Malaysia Campus