FIT1008
Introduction to computer science

Unit Guide

Semester 1, 2014

Last updated: 21 Feb 2014

Table of Contents

FIT1008 Introduction to computer science - Semester 1, 2014

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)

Workload Requirements

Minimum total expected workload equals 12 hours per week comprising:

(a.) Contact hours for on-campus students:

- Three 1-hour 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.

Additional workload requirements

The One 3-hour laboratory will be a Computer Lab Prac that includes 1.5 hours per week (requiring advance preparation) followed by an extra 1.5 hours for marking.

Unit Relationships

Prohibitions

CSE1303, CSC1030, FIT1015

Prerequisites

(FIT1040 or ECE2071 or FIT1002) and FIT1029
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.

Chief Examiner

Dr David Albrecht

Campus Lecturer

Clayton

Dr David Albrecht

Your feedback to Us

Monash is committed to excellence in education and regularly seeks feedback from students, employers and staff. One of the key formal ways students have to provide feedback is through the Student Evaluation of Teaching and Units (SETU) survey. The University's student evaluation policy requires that every unit is evaluated each year. Students are strongly encouraged to complete the surveys. The feedback is anonymous and provides the Faculty with evidence of aspects that students are satisfied and areas for improvement.

For more information on Monash's educational strategy, see:

www.monash.edu.au/about/monash-directions/ and on student evaluations, see:
www.policy.monash.edu/policy-bank/academic/education/quality/student-evaluation-policy.html

Previous Student Evaluations of this Unit

As a result of student feedback, we have done the following:

- Added programs with the code used in all lectures. This code gives you a good idea of what we want in terms of commenting and testing.
- Added quizzes to each week to help students practice the concepts being taught.

If you wish to view how previous students rated this unit, please go to
https://emuapps.monash.edu.au/unitevaluations/index.jsp

Academic Overview

Learning Outcomes

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

- implement and modify common data types such as stacks, queues, lists, trees, priority queues, heaps and hash tables using a variety of data structures such as arrays and linked nodes. Implement simple algorithms that manipulate these data types. Construct new basic data types;
- compare and evaluate different implementations of a basic data type and evaluate their implications regarding time complexity, functionality, and memory usage;
- design and implement simple recursive algorithms and data structures, including those manipulating lists, trees and heaps. Assess the relationship between recursive and iterative algorithms, their advantages and disadvantages;
- calculate the best case and worst case big O time complexity of simple iterative and recursive algorithms (including all those studied in the unit);
- explain the differences between compilation, interpretation and hybrid compilation methods. Identify different compilation targets, including abstract machine code, assembly language, object code, and machine code. Manually translate simple high-level code containing if-then-elses, loops, arithmetic and function calls into the assembly code used by a particular computer architecture such as MIPS R2000.

Unit Schedule

Week	Activities	Assessment
0	Register for tutorials, pracs and lectures	No formal assessment or activities are undertaken in week 0
1	Lectures on Python Basics, Sorting Revision and Big O. Tute & Prac.	Prac 1. Peer Instruction Participation occurs in every lecture
2	Lectures on Analysis of sorting, Assembly and MIPS Archecture. Tute & Prac.	Prac 2
3	Lectures on Assembly Programming. Tute & Prac.	Prac 3
4	Lectures on Function Calling/Returning in Assembly. Tute & Prac	Prac 4
5	Lectures on Data Structures, Classes & Objects, and Exception Handling. Tute & Prac	Prac 5
6	Lectures on Lists, Stacks and Queues. Tute & Prac.	Prac 6
7	Test, Basic Searching algorithms & Linked Structures. Tutes & No Prac.	Mid-Semester Test in Monday lecture
8	Lectures on Linked Stacks, Linked Queues and Linked Lists. Tute & Prac.	Prac 7
9	Lectures on Iterators and Recursion. Tute & Prac.	Prac 8
10	Lectures on Recursive Sorts and Binary Trees. Tute & Prac.	Prac 9
11	Lectures on Binary Search Trees, Priority Queues and Heaps. Tute & Prac.	Prac 10
12	Lectures on Hash Tables. Tute & Prac.	Prac 11
	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

*Unit Schedule details will be maintained and communicated to you via your learning system.

Teaching Approach

Lecture and 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, and practice them in a hands-on environment during labs.

Assessment Summary

Examination (3 hours): 70%; In-semester assessment: 30%

Assessment Task	Value	Due Date

Unit Schedule

Mid-semester Test (1 hour)	5%	Week 7 in Monday lecture
Pracs (1.5 hours each)	20%	Weekly except in Week 7
Peer Instruction Participation in Lecture Sessions	5%	Every lecture
Examination 1	70%	To be advised

Assessment Requirements

Assessment Policy

Faculty Policy - Unit Assessment Hurdles
(http://intranet.monash.edu.au/infotech/resources/staff/edgov/policies/assessment-examinations/assessment-hurd

Academic Integrity - Please see resources and tutorials at
http://www.monash.edu/library/skills/resources/tutorials/academic-integrity/

Assessment Tasks

Hurdle Requirements

There are four hurdles in this unit:

- First, students must actively participate in at least 7 out of 12 tutorials. Active participation includes contributing opinions to a discussion, providing an answer to some question/exercise, or posing a unit-related question.
- Second, students must achieve at least 50% of the total prac marks.
- Third, students must achieve at least 50% of the exam mark.
- Fourth, students must achieve at least 50% of the total marks.

Students who do not meet all these hurdles can get a maximum of 49N for the unit.

Participation

Students must actively participate in at least 7 out of the 12 tutorials. Active participation includes contributing opinions to a discussion, providing an answer to some question/exercise, or posing a unit-related question.

Peer Instruction Participation occurs in every lecture.

- **Assessment task 1**

  Title:
  Mid-semester Test (1 hour)
  Description:
  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.
  Weighting:
  5%
  Criteria for assessment:
  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.
  Due date:
  Week 7 in Monday lecture

- **Assessment task 2**

    Title:
    Pracs (1.5 hours each)
    Description:
    Each week you will need to complete a prac assignment. In some of the pracs, you will need to work together with another student, but will you will be marked individually. 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. The aim of the 1.5 hour computer lab practical is to iron out any bugs, ask any questions about the prac you have not been able to solve on your own, and improve the parts that your demonstrator points out as lacking (including comments, algorithms, etc). If you do nothing before the 1.5 hours scheduled, you will soon realise that you do not have enough time to complete it. The prac sheets will be released every Thursday morning and made available on the unit's web page.
    Weighting:
    20%
    Criteria for assessment:
    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.
    Due date:
    Weekly except in Week 7

- **Assessment task 3**

    Title:
    Peer Instruction Participation in Lecture Sessions
    Description:
    Student participation through the response gathering system will be recorded.
    Weighting:
    5%
    Criteria for assessment:
    Student answers during the peer instruction session will not be graded based on correctly answering questions. The grade will be based on participation.
    Due date:
    Every lecture
    Remarks:
    On Campus students are required to purchase a Turning Point clicker from the Campus Bookstore or directly from the Australian Distributor.

Examinations

- **Examination 1**

    Weighting:
    70%
    Length:
    3 hours
    Type (open/closed book):
    Closed book
    Electronic devices allowed in the exam:

Assessment Requirements

      None

Learning resources

Reading list

For algorithms and data structures:

- D. Reed and J. Zelle - Data Structures and Algorithms using Python and C++

For MIPS

- R.L. Britton - MIPS Assembly Language Programming
- D. A. Patterson and J.L. Hennessy - Computer Organisation & Design: The Hardware/Software Interface

For learning Python 3:

- L. Perkovic - Introduction to Computing using Python: An Application Development Focus

For more Python 3 (for more experienced programmers):

- Dive into Python 3: http://getpython3.com/diveintopython3/
- Python tutorial: http://docs.python.org/3/tutorial/

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

Faculty of Information Technology Style Guide

Feedback to you

Examination/other end-of-semester assessment feedback may take the form of feedback classes, provision of sample answers or other group feedback after official results have been published. Please check with your lecturer on the feedback provided and take advantage of this prior to requesting individual consultations with staff. If your unit has an examination, you may request to view your examination script booklet, see
http://intranet.monash.edu.au/infotech/resources/students/procedures/request-to-view-exam-scripts.html

Types of feedback you can expect to receive in this unit are:

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

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 leader via the in-semester special consideration process: http://www.monash.edu.au/exams/special-consideration.html

Returning assignments

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

Assignment submission

It is a University requirement (http://www.policy.monash.edu/policy-bank/academic/education/conduct/student-academic-integrity-managing-pla 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 online quiz). Please note that it is your responsibility to retain copies of your assessments.

Online submission

Please submit your work via Moodle at the time in which you are marked (and, thus, **before** you leave the lab). You can access Moodle via links in the my.monash portal.

Required Resources

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.

Python 3: This will be our programming language. Please make sure you get version 3.3

PyCharm: A good Python IDE. (http://www.jetbrains.com/pycharm/)

MARS (MIPS Assembler and Runtime Simulator): An IDE for MIPS Assembly Language Programming. (http://courses.missouristate.edu/kenvollmar/mars/)

All the above are included as part of the Standard Operating Environment used in Faculty Computer Labs.

Additional subject costs

On Campus students are required to purchase a Turning Point clicker from the Campus Bookstore or directly from the Australian Distributor.

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:
www.policy.monash.edu.au/policy-bank/academic/education/index.html

Key educational policies include:

- Student Academic Integrity Policy and Student Academic Integrity: Managing Plagiarism and Collusion Procedures ;
  http://www.policy.monash.edu/policy-bank/academic/education/conduct/student-academic-integrity-policy.h
- Assessment in Coursework Programs;
  http://www.policy.monash.edu/policy-bank/academic/education/assessment/assessment-in-coursework-po
- Special Consideration;
  http://www.policy.monash.edu/policy-bank/academic/education/assessment/special-consideration-policy.ht
- Grading Scale;
  http://www.policy.monash.edu/policy-bank/academic/education/assessment/grading-scale-policy.html
- Discipline: Student Policy;
  http://www.policy.monash.edu/policy-bank/academic/education/conduct/student-discipline-policy.html
- Academic Calendar and Semesters; http://www.monash.edu.au/students/dates/
- Orientation and Transition; http://intranet.monash.edu.au/infotech/resources/students/orientation/
- Academic and Administrative Complaints and Grievances Policy;
  http://www.policy.monash.edu/policy-bank/academic/education/management/complaints-grievance-policy.h

Faculty resources and policies

Important student resources including Faculty policies are located at
http://intranet.monash.edu.au/infotech/resources/students/

Graduate Attributes Policy

http://www.policy.monash.edu/policy-bank/academic/education/management/monash-graduate-attributes-policy.h

Student Charter

www.opq.monash.edu.au/ep/student-charter/monash-university-student-charter.html

Student services

The University provides many different kinds of support services for you. Contact your tutor if you need advice and see the range of services available at http://www.monash.edu.au/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 www.lib.monash.edu.au 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 Liaison Unit

Students who have a disability or medical condition are welcome to contact the Disability Liaison Unit to discuss academic support services. Disability Liaison Officers (DLOs) visit all Victorian campuses on a regular basis.

- Website: http://www.monash.edu/equity-diversity/disability/index.html
- Telephone: 03 9905 5704 to book an appointment with a DLO; or contact the Student Advisor, Student Commuity Services at 03 55146018 at Malaysia
- Email: dlu@monash.edu
- Drop In: Equity and Diversity Centre, Level 1, Building 55, Clayton Campus, or Student Community Services Department, Level 2, Building 2, Monash University, Malaysia Campus