MONASH University
Information Technology

FIT2004
Algorithms and data structures

Unit Guide

Semester 2, 2015

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: 23 Jul 2015

Table of Contents

FIT2004 Algorithms and data structures - Semester 2, 2015

This unit introduces students to problem solving concepts and techniques fundamental to the science of programming. In doing this it covers problem specification, algorithmic design, analysis and implementation. Detailed topics include analysis of best, average and worst-case time and space complexity; introduction to numerical algorithms; recursion; advanced data structures such as heaps and B-trees; hashing; sorting algorithms; searching algorithms; graph algorithms; and numerical computing.

Mode of Delivery

Clayton (Day)

Workload Requirements

Minimum total expected workload equals 12 hours per week comprising:

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

- Two hours of lectures
- One 3-hour laboratory/tutorial

(b.) Additional requirements (all students):

- A minimum of 7 hours of independent study per week for lab preparation and reading.

See also Unit timetable information

Additional workload requirements

Students will be expected to spend a total of 12 hours per week during semester on this unit as follows:

- Lectures: 2 hours per week

- One 3-hour laboratory per week (note, every **alternate week** starting Week 3, the 3 hour slot will be split into a 1-hour tutorial followed by a 2 hour lab)

- Reading: 3.5 hours per week
- Lab Preparation: 3.5 hours per week

Unit Relationships

Prohibitions

CSE2304, FIT2009

Prerequisites

One of FIT1008, FIT1015, FIT2085 or CSE1303 and 6 points of Level 1 mathematics.

Chief Examiner

Dr Arun Konagurthu (Semester 1)
Dr Reza Haffari (Semester 2)

Campus Lecturer

Clayton

Reza Haffari

Consultation hours: TBA in week 1 S1/2015

Tutors

Clayton

To be advised

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

Student feedback has informed a substantive revision to the unit that includes reorienting the content to be supported by a single principal text book and reorganisation to improve the incremental progression of unit content.

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 should be able to:

1. analyse general problem solving strategies and algorithmic paradigms, and apply them to solving new problems;
2. prove correctness of programs, analyse their space and time complexities;
3. compare and contrast various abstract data types and use them appropriately;
4. develop and implement algorithms to solve computational problems.

Unit Schedule

Week	Activities	Assessment
0	Revise concepts learnt in FIT1029 (Dynamic Programming, Divide-Conquer, sorting, Recursion, Invariants, Trees etc.)	No formal assessment or activities are undertaken in week 0
1	Introduction: Proof Techniques, Mathematics, Recursion, Algorithm Analysis (Correctness Proof, Complexity)	
2	Analysis of Sorting Algorithms, Divide & Conquer, Dictionary ADT, Hash Tables	Non assessed 3hr lab
3	Dictionary ADT: Binary Search Trees, AVL Trees, Red-Black Trees	Non assessed 1hr tute + 2hr lab
4	Self-balancing Search Trees (B-Trees), Heaps (Build Heap, Heap Sort, Leftist Heaps)	Assignment 1 due (Assessed 3hr lab)
5	Disjoint Sets ADT, Amortized Analysis	Non assessed 1hr tute + 2hr lab
6	String Algs and Data Structures: Pattern Matching and Suffix Arrays/Tries	Assignment 2 due (Assessed 3hr lab)
7	Graphs: Representations and Shortest Path Problem	Non assessed 1hr tute + 2hr lab
8	Graphs: Shortest Path Problem and Maximum Flow Problem	Assignment 3 due (Assessed 3hr lab)
9	Minimum Spanning Tree in Graphs, Intro to Computational Complexity and Intractibility	Non assessed 1hr tute + 2hr lab
10	Alg Design Paradigms: Divide & Conquer, Dynamic Programming	Assignment 4 due (Assessed 3hr lab)
11	Alg Design Paradigms: Greedy, Randomised, Backtrack	Non assessed 1hr tute + 2hr lab
12	Numerical Algs: Root Finding, Integration	Assignment 5 due (Assessed 3hr lab)
	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 provides facilitated learning, practical exploration and peer learning.

Assessment Summary

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

Assessment Task	Value	Due Date
Assignment 1	6%	Week 4
Assignment 2	6%	Week 6
Assignment 3	6%	Week 8
Assignment 4	6%	Week 10
Assignment 5	6%	Week 12
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

Participation

- **Assessment task 1**

  Title:
  Assignment 1
  Description:
  Practical problems arising from the material covered in lectures and tutorials.
  Weighting:
  6%
  Criteria for assessment:

  - ♦ Demonstrate code where applicable
  - ♦ Explain your solutions

  Due date:
  Week 4

- **Assessment task 2**

  Title:
  Assignment 2
  Description:
  Practical problems arising from the material covered in lectures and tutorials.
  Weighting:
  6%
  Criteria for assessment:

  - ♦ Demonstrate code where applicable
  - ♦ Explain your solutions

  Due date:
  Week 6

- **Assessment task 3**

  Title:
  Assignment 3
  Description:
  Practical problems arising from the material covered in lectures and tutorials.
  Weighting:

6%

Criteria for assessment:

- ♦ Demonstrate code where applicable
- ♦ Explain your solutions

Due date:

Week 8

- **Assessment task 4**

Title:

Assignment 4

Description:

Practical problems arising from the material covered in lectures and tutorials.

Weighting:

6%

Criteria for assessment:

- ♦ Demonstrate code where applicable
- ♦ Explain your solutions

Due date:

Week 10

- **Assessment task 5**

Title:

Assignment 5

Description:

Practical problems arising from the material covered in lectures and tutorials.

Weighting:

6%

Criteria for assessment:

- ♦ Demonstrate code where applicable
- ♦ Explain your solutions

Due date:

Week 12

Examinations

- **Examination 1**

Weighting:

70%

Length:

3 hours

Type (open/closed book):

Closed book

Electronic devices allowed in the exam:

None

Learning resources

Reading list

Main reading:

- M. A. Weiss. Data Structures and Algorithm Analysis in Java

Additional reading (chapters to be specified clearly in the lecture slides as the course progresses):

- Introduction to Algorithms/ Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein (3rd Edition), McGraw Hill
- Data Structures and Algorithms in Python, Michael T. Goodrich, Roberto Tamassia, Michael Goldwasser

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

Feedback to you

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

- Informal feedback on progress in labs/tutes
- Graded assignments without comments
- Solutions to tutes, labs and assignments

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 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.

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 (latest version) installed in the labs, you can download a free copy from Python.org

Prescribed text(s)

Limited copies of prescribed texts are available for you to borrow in the library.

Mark Allen Weiss. (2012). *Data Structures and Algorithm Analysis in Java*. (3rd Edition) Pearson (ISBN: 9780132576277).

Recommended text(s)

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein. (2009). *Introduction to Algorithms*. (3rd Edition) MIT Press.

Donald Knuth. (1997). *The Art of Computer Programming*. () Addison Wesley.

Examination material or equipment

Closed book. No calculators.

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

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

Other

Engineers Australia Stage 1 competencies

This unit is a core unit in the Bachelor of Software Engineering accredited by Engineers Australia. Engineers Australia Accreditation Policy of Professional Engineering Programs requires that programs demonstrate how engineering graduates are prepared for entry to the profession and achieve Stage 1 competencies. The following information describes how this unit contributes to the development of these competencies for the Bachelor of Software Engineering. (Note: not all competencies may be emphasised in this unit).

Stage 1 competency	**How the compency is developed in this unit**
1. Knowledge and Skills base	
1.1. **Comprehension, theory based understanding** of the underpinning natural and physical sciences and the engineering fundamentals applicable to the engineering discipline.	This unit covers data structures, algorithms and complexity, which fall under required foundation knowledge for software engineering (SWEBOK). This element of competency is covered by lecture notes, practical exercises and assignments.
1.2. **Conceptual understanding** of the mathematics, numerical analysis, statistics, and computer and information sciences, which underpin the engineering discipline.	The unit deals with developing algorithms using mathematical knowledge and numerical analysis to develop software.
1.3. **In-depth understanding** of specialist bodies of knowledge within the engineering discipline.	The unit addresses one aspect of SWEBOK (Software Engineering Body of Knowledge) – Computing Foundations for software engineers.
1.4. **Discernment** of knowledge development and research directions within th engineering discipline.	Not covered in this unit.
1.5. **Knowledge** of engineering design practice and contextual factors impacting the engineering discipline.	Not covered in this unit.
1.6. **Understanding** of the scope, principles, norms, accountabilities and bounds of sustainable engineering practice in the specific discipline.	Not covered in this unit.
2. Engineering application ability	
2.1. **Application** of established engineering methods to complex engineering problem solving.	Lectures, practical exercises and assignments applies problem solving strategies to resolve software problems,
2.2 **Fluent application** of engineering techniques, tools and resources.	Students learn to use mathematical tools and methods to solve problems in their assignments.
2.3. **Application** of systematic engineering synthesis and design processes.	Not covered in this unit.
2.4. **Application** of systematic approaches to the conduct and management of engineering projects.	Not covered in this unit.
3. Professional and personal attributes	
3.1. **Ethical** conduct and professional accountability.	Not covered in this unit.
3.2. **Effective** oral and written communication in professional and lay domains.	Precise languages and notations are employed in lectures and assignments. Students need also to write up reports as part of their assignment assessment.
3.3. **Creative**, innovative and proactive demeanour.	

	Developing algorithms is inherently a creative endeavour.
3.4. **Professional** use and management of information.	Not covered in this unit.
3.5. **Orderly** management of self, and professional conduct.	Not covered in this unit.
3.6. **Effective** team membership and team leadership.	Not covered in this unit.

Relationship between Unit Learning Outcomes and BSE Course Outcomes

No.	CO 1	CO 2	CO 3	CO 4	CO 5	CO 6	CO 7	C0 8	CO 9	CO 10	CO 11	CO 12	CO 13
1	X	X				X		X					
2				X		X		X					
3	X			X			X	X					
4	X			X			X	X					

Relationship between Unit Learning Outcomes and Assessments

No.	Assignments	Tests	Practical Exercises	Exam
1	X		X	X
2	X			X
3	X		X	X
4	X		X	