FIT2004
Algorithms and data structures

Unit Guide

Semester 1, 2011

Last updated: 27 Feb 2011

Table of Contents

FIT2004 Algorithms and data structures - Semester 1, 2011

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)
- Sunway (Day)

Contact Hours

2 hrs lectures/wk, 3 hr laboratory/fortnight, 1 hr tutorial/fortnight

Workload

2 hours of lecture

1 hour of tutorial (fortnightly)

3 hours of laboratory (fortnightly)

4 hours reading

4 hours laboratory preparation

Unit Relationships

Prohibitions

CSE2304, FIT2009, FIT2071

Prerequisites

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

Chief Examiner

Geoff Webb

Campus Lecturer

Clayton

Geoff Webb

Sunway

Farhad Keissarian

Learning Objectives

At the completion of this unit students will have:

- understanding of a formal specification;
- ability to create a formal specification for an informal problem;
- knowledge and understanding of algorithmic properties such as correctness, termination and complexity;
- ability to, given a non-trivial algorithm, formally prove certain properties, such as correctness and termination;
- ability, given a non-trivial algorithm, to determine its best, average and worst-case, time and space-complexity;
- knowledge and understanding of reasonably complex data structures such as minimum spanning trees, and Directed and Undirected, Weighted and Unweighted Graphs;
- ability to design and implement new non-trivial algorithms using complex data structures;
- knowledge of and ability to use algorithmic paradigms such as divide and conquer, greedy, dynamic programming and so on;
- ability to identify these paradigms in diverse algorithms;
- knowledge and understanding of the issues involved in implementing a non-trivial algorithm efficiently.

Developed attitudes that enable them to:

- carefully design and/or analyse the algorithms they are using in order to verify important properties such as correctness, termination, and complexity;
- identify the key features of a brief informal problem description and abstract the underlying formal problem.

Developed the skills to:

- create their own data structures.
- create a new algorithm to solve a new problem.
- make a formal argument about desirable properties of the solution.
- adapt an existing algorithm and/or data-structure where that is possible and appropriate.
- implement a non-trivial algorithm efficiently.

Demonstrated the communication skills necessary to:

- make a formal argument that an algorithm and/or data-structure has a given property, such as correctness, termination or complexity.

Graduate Attributes

Monash prepares its graduates to be:

1. responsible and effective global citizens who:

a. engage in an internationalised world
b. exhibit cross-cultural competence
c. demonstrate ethical values

critical and creative scholars who:

a. produce innovative solutions to problems
b. apply research skills to a range of challenges
c. communicate perceptively and effectively

Assessment Summary

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

Assessment Task	Value	Due Date
ADTs, Proofs & Induction	6 %	18 March 2011
Complexity Analysis	6 %	1 April 2011
Dynamic Programming	6 %	15 April 2001
Trees	6 %	6 May 2011
Graphs & Graph Algorithms	6 %	27 May 2011
Examination 1	70%	To be advised

Teaching Approach

The approach to teaching and learning includes two weekly one-hour lectures, a fortnightly one-hour tutorial and a fortnightly three-hour laboratory. Additionally, each student should spend a minimum of 8 to 12 hours for personal study every week and should allocate up to 5 hours per week in some weeks for use of a computer, including time for newsgroup and discussion.

Feedback

Our 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

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 SETU, Student Evaluation of Teacher and Unit. 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, and on student evaluations, see:
http://www.monash.edu.au/about/monash-directions/directions.html
http://www.policy.monash.edu/policy-bank/academic/education/quality/student-evaluation-policy.html

Previous Student Evaluations of this unit

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

Required Resources

Java (latest version) installed in the labs, you can download a free copy from Sun Microsystems.

Examination material or equipment

None

Unit Schedule

Week	Date*	Activities	Assessment
0	21/02/11		No formal assessment or activities are undertaken in week 0
1	28/02/11	Specification & Abstract Data Types	
2	07/03/11	Proofs & Induction	
3	14/03/11	Complexity Analysis I	
4	21/03/11	Complexity Analysis II	Assignment 1 due 8 March 2011
5	28/03/11	Pattern Matching	
6	04/04/11	Dynamic Programming	Assignment 2 due 1 April 2011
7	11/04/11	Dynamic & Balanced Trees	
8	18/04/11	Amortized Analysis	Assignment 3 due 15 April 2011
		Mid semester break	
9	02/05/11	Multi-way Trees	
10	09/05/11	Graphs	Assignment 4 due 6 May 2011

11	16/05/11	Path Problems	
12	23/05/11	Flow Problems	Assignment 5 due 27 May 2011
	30/05/11	SWOT VAC	No formal assessment is undertaken SWOT VAC

*Please note that these dates may only apply to Australian campuses of Monash University. Off-shore students need to check the dates with their unit leader.

Assessment Policy

To pass a unit which includes an examination as part of the assessment a student must obtain:

- 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 greater than 50% then a mark of no greater than 49-N will be recorded for the unit

Assessment Tasks

Participation

- **Assessment task 1**

  Title:
  ADTs, Proofs & Induction
  Description:
  specific tasks and marking criteria will be distributed at the appropriate time during the semester
  Weighting:
  6 %
  Criteria for assessment:
  specific tasks and marking criteria will be distributed at the appropriate time during the semester
  Due date:
  18 March 2011

- **Assessment task 2**

  Title:
  Complexity Analysis
  Description:
  specific tasks and marking criteria will be distributed at the appropriate time during the semester
  Weighting:
  6 %

Criteria for assessment:

> specific tasks and marking criteria will be distributed at the appropriate time during the semester

Due date:

> 1 April 2011

- ## Assessment task 3

    Title:

    > Dynamic Programming

    Description:

    > specific tasks and marking criteria will be distributed at the appropriate time during the semester

    Weighting:

    > 6 %

    Criteria for assessment:

    > specific tasks and marking criteria will be distributed at the appropriate time during the semester

    Due date:

    > 15 April 2001

- ## Assessment task 4

    Title:

    > Trees

    Description:

    > specific tasks and marking criteria will be distributed at the appropriate time during the semester

    Weighting:

    > 6 %

    Criteria for assessment:

    > specific tasks and marking criteria will be distributed at the appropriate time during the semester

    Due date:

    > 6 May 2011

- ## Assessment task 5

    Title:

    > Graphs & Graph Algorithms

    Description:

    > specific tasks and marking criteria will be distributed at the appropriate time during the semester

    Weighting:

    > 6 %

    Criteria for assessment:

    > specific tasks and marking criteria will be distributed at the appropriate time during the semester

    Due date:

    > 27 May 2011

Examinations

- **Examination 1**

  Weighting:
  > 70%
  Length:
  > 3 hours
  Type (open/closed book):
  > Closed book
  Electronic devices allowed in the exam:
  > None

Assignment submission

Assignment coversheets are available via "Student Forms" on the Faculty website:
http://www.infotech.monash.edu.au/resources/student/forms/
You MUST submit a completed coversheet with all assignments, ensuring that the plagiarism declaration section is signed.

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.infotech.monash.edu.au/resources/student/equity/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

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

Key educational policies include:

- Plagiarism
  (http://www.policy.monash.edu/policy-bank/academic/education/conduct/plagiarism-policy.html)
- Assessment
  (http://www.policy.monash.edu/policy-bank/academic/education/assessment/assessment-in-coursework-p
- Special Consideration
  (http://www.policy.monash.edu/policy-bank/academic/education/assessment/special-consideration-policy.h
- 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/key-dates/);
- Orientation and Transition (http://www.infotech.monash.edu.au/resources/student/orientation/);
  and
- Academic and Administrative Complaints and Grievances Policy
  (http://www.policy.monash.edu/policy-bank/academic/education/management/complaints-grievance-policy.

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 www.monash.edu.au/students The Monash University Library provides a range of services and resources that enable you to save time and be more effective in your learning and research. Go to http://www.lib.monash.edu.au or the library tab in my.monash portal for more information. 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://adm.monash.edu/sss/equity-diversity/disability-liaison/index.html;
- Telephone: 03 9905 5704 to book an appointment with a DLO;
- Email: dlu@monash.edu
- Drop In: Equity and Diversity Centre, Level 1 Gallery Building (Building 55), Monash University, Clayton Campus.

READING LIST

Mark Allen Weiss:
Data Structures and Algorithm Analysis in Java. 2nd ed
Addison-Wesley.

Michael Goodrich and Roberto Tamassia.
Data Structures and Algorithms in Java, 3rd ed
John Wiley.

Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein. Introduction to Algorithms, 2nd Edition, EHT MIT Press & McGraw Hill

 Duane Bailey. Java Structures: Data Structures in Java for the Principled Programmer. International Edition, May 1999, Mc Graw Hill. This book is available for free online at
http://www.cs.williams.edu/~bailey/JavaStructures/Welcome.html