



MONASH University
Information Technology

FIT2009
Data structures and algorithms

Unit Guide

Semester 2, 2011

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: 22 Aug 2011

Table of Contents

| | |
|-------------------------------------------------------------------------------|-----------|
| <u>FIT2009 Data structures and algorithms - Semester 2, 2011</u> | 1 |
| <u>Mode of Delivery</u> | 1 |
| <u>Contact Hours</u> | 1 |
| <u>Workload</u> | 1 |
| <u>Unit Relationships</u> | 1 |
| <u>Prohibitions</u> | 1 |
| <u>Prerequisites</u> | 1 |
| <u>Chief Examiner</u> | 1 |
| <u>Campus Lecturer</u> | 2 |
| <u>Gippsland</u> | 2 |
| <u>South Africa</u> | 2 |
| <u>Academic Overview</u> | 3 |
| <u>Learning Objectives</u> | 3 |
| <u>Graduate Attributes</u> | 3 |
| <u>Assessment Summary</u> | 3 |
| <u>Teaching Approach</u> | 4 |
| <u>Feedback</u> | 4 |
| <u>Our feedback to You</u> | 4 |
| <u>Your feedback to Us</u> | 4 |
| <u>Previous Student Evaluations of this unit</u> | 4 |
| <u>Required Resources</u> | 4 |
| <u>Recommended Resources</u> | 5 |
| <u>Unit Schedule</u> | 6 |
| <u>Assessment Requirements</u> | 7 |
| <u>Assessment Policy</u> | 7 |
| <u>Assessment Tasks</u> | 7 |
| <u>Participation</u> | 7 |
| <u>Examinations</u> | 8 |
| <u>Examination 1</u> | 8 |
| <u>Assignment submission</u> | 8 |
| <u>Extensions and penalties</u> | 8 |
| <u>Returning assignments</u> | 8 |
| <u>Resubmission of assignments</u> | 9 |
| <u>Other Information</u> | 10 |
| <u>Policies</u> | 10 |
| <u>Student services</u> | 10 |

FIT2009 Data structures and algorithms - Semester 2, 2011

Algorithm analysis. Application and implementation of some common data structures: stacks, queues, lists, priority queues, tables, sets and collections. Data representations including: arrays, linked lists, heaps, trees (including balanced trees) and hashing. Design of application programs making use of common data structures. Design and implementation of new data structures. Study of advanced algorithms in areas such as: graph theory, pattern searching and data compression. Access to the University's computer systems through an Internet service provider is compulsory for off-campus students.

Mode of Delivery

- Gippsland (Day)
- Gippsland (Off-campus)
- South Africa (Day)

Contact Hours

2 hrs lectures/wk, 2 hrs laboratories/wk

Workload

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

For on-campus students:

Lectures: 2 hours per week

Tutorials/Lab Sessions: 2 hours per week per tutorial

and up to an additional 8 hours in some weeks for completing lab and project work, private study and revision.

Off-campus students generally do not attend lecture and tutorial sessions, however, you should plan to spend equivalent time working through the relevant resources and participating in discussion groups each week.

Unit Relationships

Prohibitions

FIT2004, FIT2071, FIT9015, GCO2817, GCO3512, GCO9807

Prerequisites

FIT1007 or GCO1812 or GCO9808 or FIT2034

Chief Examiner

Associate Professor Manzur Murshed

Campus Lecturer

Gippsland

Manzur Murshed

South Africa

Neil Manson

Academic Overview

Learning Objectives

At the completion of this unit students will have -

- the ability to analyse simple algorithms to work out an order of magnitude estimate of running time and space;
- familiarity with some of the most common data structures: stacks, queues, lists, priority queues, tables, sets, collections;
- the ability to implement these data structures using various common data representations: arrays, linked lists, heaps, trees (including balanced trees), hashing;
- the ability to evaluate which implementation would be most appropriate for a given data structure and application;
- the ability to apply the same principles used in implementing the common data structures to implement other data structures;
- ability to design and implement new data structures;
- an understanding of some more advanced algorithms in areas such as: graph theory (shortest path etc), pattern searching, data compression (precise selection of advanced algorithms will vary from year to year);
- the ability to design new algorithms to solve new problems;
- an enjoyment of programming as an intellectual exercise;
- an appreciation of the elegance of certain data structures and algorithms as a form of art;
- an interest in understanding how data structures and algorithms are implemented rather than merely using other peoples implementations (and consequently a preference for open source software.

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): 60%; In-semester assessment: 40%

| Assessment Task | Value | Due Date |
|------------------------|--------------|---------------------|
| Assignment 1 | 20% | 5 September 2011 |

Academic Overview

| | | |
|---------------|-----|--------------------|
| Assignment 2 | 20% | 10 October 2011 |
| Examination 1 | 60% | To be advised |

Teaching Approach

Lecture and tutorials or problem classes

This teaching and learning approach provides facilitated learning, practical exploration and peer learning.

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 with 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

Textbook

Mark Allen Weiss, *Data Structures & Problem Solving using Java*, 4th Edition, Addison Wesley, 2010, ISBN: 0-321-54622-9.

Text books are available from the Monash University Book Shops. Availability from other suppliers cannot be assured. The Bookshop orders texts in specifically for this unit. You are advised to purchase your text book early.

Recommended Resources

Required software

Java SE JDK version 1.5 (also known as version 5) or later.

This software is included in the GSIT Unit Software CD-ROM, which will be sent to all students.

The software may also be downloaded free from <http://java.sun.com>

Unit Schedule

| Week | Activities | Assessment |
|------|-------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 0 | | No formal assessment or activities are undertaken in week 0 |
| 1 | Generic Data Structures (Study Guide 1) | |
| 2 | Algorithm Analysis (Study Guide 2) | |
| 3 | Developing Algorithms (Study Guide 3) | |
| 4 | Sorting Algorithms (Study Guide 4) | |
| 5 | Lists (Study Guide 5) | |
| 6 | Stacks and Queues (Study Guide 6) | |
| 7 | Graphs and Trees (Study Guide 7) | Assignment 1 due 5 September 2011 |
| 8 | Binary Search Trees (Study Guide 8) | |
| 9 | Hashing (Study Guide 9) | |
| 10 | Heaps (Study Guide 10) | |
| 11 | Some Applications of Data Structures (Study Guide 11) | Assignment 2 due 10 October 2011 |
| 12 | Revision | |
| | 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 MUSO (Blackboard or Moodle) learning system.

Assessment Requirements

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:

Assignment 1

Description:

Students will be required to perform a number of tasks involving both analytical and practical (computer programming) skills from the syllabus covered in Study Guides 1-4.

Weighting:

20%

Criteria for assessment:

The assignment requires individual submission via MUSO. The specification and marking criteria will be released in MUSO four teaching weeks in advance of the due date. Solutions will be released after the cut-off date, which is one week after the due date.

Broad criterial for assessment:

1. How well underlying principles and theories are demonstrated in the student's answer
2. The degree to which programs meet the problem specification
3. The quality of the student's argument

Due date:

5 September 2011

• Assessment task 2

Title:

Assignment 2

Description:

Students will be required to perform a number of tasks involving both analytical and practical (computer programming) skills from the syllabus covered in Study Guides 5-8.

Weighting:

20%

Criteria for assessment:

Assessment Requirements

The assignment requires individual submission via MUSO. The specification and marking criteria will be released in MUSO four teaching weeks in advance of the due date. Solutions will be released after the cut-off date, which is one week after the due date.

Broad criteria for assessment:

1. How well underlying principles and theories are demonstrated in the student's answer
2. The degree to which programs meet the problem specification
3. The quality of the student's argument

Due date:

10 October 2011

Examinations

• Examination 1

Weighting:

60%

Length:

3 hours

Type (open/closed book):

Closed book

Electronic devices allowed in the exam:

None

Assignment submission

It is a University requirement

(<http://www.policy.monash.edu/policy-bank/academic/education/conduct/plagiarism-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 online quiz).

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

Resubmission of assignments

- Late penalties will be enforced if an assignment is submitted after the due date.
- Students may resubmit an assignment any time before the cut-off date, which is usually a week after the due date.
- No assignment can be submitted after the cut-off date.

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://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>)
- Codes of Practice for Teaching and Learning
(<http://www.policy.monash.edu.au/policy-bank/academic/education/conduct/suppdocs/code-of-practice-tea>)

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.

Recommended text(s) and readings

1. Ford, W. H. and Topp, W. R., Data Structures with Java, 2005, Pearson Education International, ISBN:0-131-29337-0.
2. Lafore, R., Data Structures & Algorithms in Java, 2nd Edition, 2002, SAMS, ISBN: 0-672-32453-9.
3. Gray, S., Data Structures in Java, 2007, Pearson Education Inc., ISBN: 0-321-39279-5.

Other Information

Study resources

Study resources we will provide for your study are:

- A Unit Book containing 11 study guides at MUSO.
- This Unit Information, which outlines the administrative information for the unit.
- A CD-ROM (possibly sent at the start of the year if you were enrolled in a first semester unit that required it) with software required for all units (this includes all the software required to complete this unit).
- A unit web page at MUSO where lecture slides, weekly tutorial requirements, assignment specifications, sample solutions and supplementary material will be posted.
- Discussion forums at MUSO.