MONASH University
Information Technology

FIT1029
Algorithmic problem solving

Unit Guide

Semester 1, 2013

Last updated: 04 Mar 2013

Table of Contents

FIT1029 Algorithmic problem solving - Semester 1, 2013

Algorithms are recipes for solving a problem. They are fundamental to computer science and software engineering. Algorithms are the formal foundation of computer programming but also exist independently of computers as systematic problem-solving procedures. This unit introduces algorithmics, the study of algorithms. It is not about programming and coding but rather about understanding and analysing algorithms and about algorithmic problem-solving, i.e. the design of systematic problem-solving procedures. The unit will not require any knowledge of a programming language and is very hands-on. Students will develop algorithms to solve a wide variety of different problems, working individually as well as together in groups and as a class.

Topics include: what is a computational problem and what is an algorithm; basic control structures; basic data structures; modular algorithm structure; recursion; problem-solving strategies for algorithm development; arguing correctness of an algorithm; arguing termination of an algorithm; understanding the efficiency of an algorithm; and limitations of algorithms.

Mode of Delivery

- Clayton (Day)
- Sunway (Day)

Contact Hours

2 hrs lectures/wk, 2 hrs tutorials/wk

Workload requirements

Students will be expected to spend 12 hours per week on various activities including:

- two-hour lecture and
- two-hour tutorial (requiring advance preparation)
- a minimum of 2-3 hours of personal study per one hour of contact time in order to satisfy the reading and assignment expectations.
- You will need to allocate up to 2 hours per week in some weeks, for use of a computer, including time for newsgroups/discussion groups.

Chief Examiner

Dr David Albrecht

Campus Lecturer

Clayton

Dr. Peter Tischer

Sunway

Dr. Simon Egerton

Academic Overview

Learning Outcomes

At the completion of this unit students will have -A knowledge and understanding of:

- the difference between algorithms and processes;
- basic ways to structure algorithms: basic data structures (simple variables, collections structure, specifically vectors, lists, sets, and tables); basic control structures (sequence, choice, iteration);
- recursion;
- modular algorithm structures;
- the equivalence of recursion and iteration;
- problem solving strategies suitable for algorithm development including top-down design and bottom-up design;
- simple standard patterns for algorithms (eg traversal, search);
- what makes a good algorithm
- limitations of algorithms (high level).

Developed the skills to:

- develop simple iterative and recursive algorithms
- argue the correctness of simple algorithms
- judge the efficiency of simple algorithms, and

Developed attitudes that enable them to:

- value clear specification of problems;
- understand the relation between algorithms and programs;
- appreciate the value of designing abstract algorithms before starting to code a program;
- have confidence that they can develop algorithms to solve computational problems;
- appreciate that seemingly difficult problems can have very simple elegant algorithmic solutions (and vice versa);
- value correctness arguments for algorithms; and
- value the importance of simplicity and efficiency.

Demonstrated the communication skills necessary to:

- solve a problem by discussing possible approaches and solutions as a team; and
- clearly communicate (the specification of) a computational problem, its algorithmic solution and arguments for correctness and efficiency.

Unit Schedule

Week	Activities	Assessment
0		No formal assessment or activities are undertaken in week 0
1	Introduction to the unit and the type of problems	
2	Understanding and modelling the problem	
3	Invariants in problems and data	
4	Decomposition of problems and applying Brute Force to solve problems	Assignment 1 due
5	Using abstraction, symmetry, heuristics and divide and conquer to simplify problems	
6	Recursion	Test during lecture
7	Backtracking	
8	Dynamic Programming	
9	Fundamentals	
10	Abstract Data Types and Correctness	Assignment 2 due
11	Complexity	
12	Limitations of algorithms	
	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.

Assessment Summary

Examination (3 hours): 60%; In-semester assessment: 40%

Assessment Task	Value	Due Date
Assignment 1	10%	Week 4
Test	15%	Week 6
Assignment 2	15%	Week 10
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.

Assessment Requirements

Assessment Policy

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

Academic Integrity - Please see the Demystifying Citing and Referencing tutorial at
http://lib.monash.edu/tutorials/citing/

Assessment Tasks

Participation

- **Assessment task 1**

  Title:
  Assignment 1
  Description:
  This assignment will aim to help you understand how to go about finding algorithms to solve problems.
  Weighting:
  10%
  Criteria for assessment:
  Detailed assessment criteria will be issued along with the assignment.

  1. All assumptions should be stated.
  2. All algorithms must meet the problem specification.
  3. Students should be able to answer questions about their own work.

  Due date:
  Week 4

- **Assessment task 2**

  Title:
  Test
  Description:
  This test will be based on the first 4 weeks of material.
  Weighting:
  15%
  Criteria for assessment:
  Detailed assessment criteria will be issued along with the assignment.

  1. All assumptions should be stated.
  2. All algorithms must meet the problem specification.
  3. Students should be able to answer questions about their own work.

  Due date:
  Week 6

- **Assessment task 3**

Title:
Assignment 2
Description:
This assignment will help you understand different search techniques. It will also help you to communicate and reason about algorithms.
Weighting:
15%
Criteria for assessment:
Detailed assessment criteria will be issued along with the assignment.

1. All assumptions should be stated.
2. All algorithms must meet the problem specification.
3. Students should be able to answer questions about their own work.

Due date:
Week 10

Examinations

- **Examination 1**

Weighting:
60%
Length:
3 hours
Type (open/closed book):
Closed book
Electronic devices allowed in the exam:
None

Learning resources

Reading list

1. Levitin, A., *Introduction to the Design and Analysis of Algorithms* (2nd Edition), Addison-Wesley, 2006
2. Harel, D. with Y. Feldman, *Algorithmics: The Spirit of Computing (*3rd Edition), Pearson Education Limited, 2004.
3. Michalewicz, Z. and M. Michalewicz, *Puzzle-Based Learning: An introduction to critical thinking, mathematics, and problem solving,* Hybrid Publishers, 2008.
4. Polya, G., *How to solve it; a new aspect of mathematical method* (2nd Edition), Garden City, N.Y., Doubleday, 1957
5. Bentley, J., *Programming Pearls,* Addison-Wesley, 1986
6. Bentley, J., *More Programming Pearls: confessions of a coder,* Addison-Wesley, 1988
7. Skiena, S., *The Algorithm Design Manual, TELOS--the Electronic Library of Science,* 1998
8. Cormen, T., C.E. Leiserson, R.L. Rivest, and C. Stein, *Introduction to Algorithms*, The MIT Press, 1990

Monash Library Unit Reading List
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 with comments
- Test results and feedback

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.

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).

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.

Recommended text(s)

Levitin, A. (2006). *Introduction to the Design and Analysis of Algorithms*. (2nd Edition) Addison-Wesley.

Harel, D. with Feldman, Y. (2004). *Algorithmics: The Spirit of Computing*. (3rd Edition) Pearson Education Limited.

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:

- Plagiarism; http://www.policy.monash.edu/policy-bank/academic/education/conduct/plagiarism-policy.html
- 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
- Code of Practice for Teaching and Learning; http://www.policy.monash.edu.au/policy-bank/academic/education/conduct/suppdocs/code-of-practice-teac

Graduate Attributes Policy

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

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 Sunway 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 Sunway, 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.htmlTelephone: 03 9905 5704 to book an appointment with a DLO; or contact the Student Advisor, Student Commuity Services at 03 55146018 at SunwayEmail: dlu@monash.eduDrop In: Equity and Diversity Centre, Level 1, Building 55, Clayton Campus, or Student Community Services Department, Level 2, Building 2, Monash University, Sunway Campus

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

This unit was first offered in 2010 and in Semester 1, 2012, some of those students were taking third year units. Evidence gathered over the years indicates that the introduction of FIT1029 has helped to improve the students' programming skills and improved the pass rates in core BCS and BSE units.

Feedback from students who have taken FIT1029 and have gone on to later year studies in the BCS and the BSE indicates that students feel that FIT1029 has helped them gain a better appreciation of algorithms, and to regard thinking about algorithms as an activity that should be kept separated from coding algorithms into a particular computer programming language.

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