MONASH University
Information Technology

FIT3042
System tools and programming languages

Unit Guide

Semester 1, 2015

Last updated: 24 Feb 2015

Table of Contents

FIT3042 System tools and programming languages - Semester 1, 2015

This unit provides students with an introduction to UNIX tools for managing processes; searching, editing and modifying files and data streams; and command interpreters and shell scripts. In addition, students will learn about a typical system call interface and its use for systems programming in a language like C.

Mode of Delivery

- Clayton (Day)
- Malaysia (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 2-hour laboratory

(b.) Additional requirements (all students):

- A minimum of 2-3 hours of personal study per one hour of contact time in order to satisfy the reading and assignment expectations.

See also Unit timetable information

Unit Relationships

Prohibitions

CSE2391, CSE3391

Prerequisites

One of FIT1008, FIT1015, FIT2085, CSE1303

Chief Examiner

Dr Peter Tischer

Campus Lecturer

Clayton

Peter Tischer

Consultation hours: Mondays 1.00pm - 2.00pm, appointments can also be arranged via email

Malaysia

Dr Ahmed Elmesiry

Consultation hours: Will be notified at the lecture.

Tutors

Malaysia

Dr Ahmed Elmesiry

Consultation hours: Will be notified at the lecture.

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

The exam format will more closely reflect the language-oriented content of the unit.

Material relating to Makefiles has been expanded. Lab sheets have been restructured.

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. describe the relationship between Unix, Unix shells, and C language;
2. write programs using C's dynamic memory allocation to create standard container structures;
3. construct programs in C that use system calls to interact with the underlying operating environment;
4. use the Make utility to build programs written in C;
5. construct shell scripts that use standard POSIX filters in pipelines to achieve common tasks;
6. use regular expressions to perform text-processing tasks in shell and Perl on unstructured and semi-structured text;
7. build Perl scripts using Perl features such as hashes and references.

Unit Schedule

Week	Activities	Assessment
0		No formal assessment or activities are undertaken in week 0
1	Unit Introduction	Laboratory Exercises are assessed at the end of each lab session
2	Introduction to C, Make	
3	C programming: Pointers & Data structures	
4	Unix C Programming	Assignment 1 handed out
5	Inter-process communication, third-party libraries	
6	Introduction to shell programming	
7	Shell filters	
8	Shell programming	Assignment 1 due Monday 27 April 2015
9	Regular expressions	Assignment 2 handed out
10	Perl 1: scalars & arrays	
11	Perl 2: Perl regexes	
12	Perl 3: Perl modules, Perl 6.	Assignment 2 due Monday 25 May 2015
	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

The teaching and learning approach provides facilitated learning, practical exploration and peer learning, equipping you with the ability to apply skills upon completion.

Assessment Summary

Examination (3 hours): 50%; In-semester assessment: 50%

Assessment Task	Value	Due Date
Assignment 1	20%	Monday 27 April 2015
Assignment 2	20%	Monday 25 May 2015
Laboratory Exercises	10%	At the end of each lab session
Examination 1	50%	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

Students are expected to attend at least 8 of the 11 labs.

- **Assessment task 1**

  Title:
  Assignment 1
  Description:
  C/Unix programming assignment
  Weighting:
  20%
  Criteria for assessment:

  - Correctness
  - Efficiency
  - Quality of solution
  - Documentation

  Due date:
  Monday 27 April 2015

- **Assessment task 2**

  Title:
  Assignment 2
  Description:
  Shell/Perl programming assignment
  Weighting:
  20%
  Criteria for assessment:

  - Correctness
  - Efficiency
  - Quality of solution
  - Documentation

  Due date:
  Monday 25 May 2015

- **Assessment task 3**

Title:
Laboratory Exercises
Description:
Exercises held during laboratory sessions.
Weighting:
10%
Criteria for assessment:
Lab exercises are assessed during the scheduled laboratory session. Marks are awarded for successful completion of the laboratory exercises.
Due date:
At the end of each lab session

Examinations

- **Examination 1**

Weighting:
50%
Length:
3 hours
Type (open/closed book):
Open book
Electronic devices allowed in the exam:
None

Learning resources

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 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.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.

Resubmission of assignments

Multiple assignment submission may be enabled in Moodle. However, only the last version will be assessed. Resubmission after the due date will only be permitted after special consideration is granted through the regular faculty processes, or, occasionally, in other exceptional circumstances with lecturer permission. Penalties may apply in such circumstances.

Referencing requirements

Any written work must use appropriate referencing methods, according to the Library Guides for citing and referencing http://guides.lib.monash.edu/content.php?pid=88267&sid=656564

Generally, code submitted in your assignments should be your own original work. However, where code uses ideas from specific sources, they should be cited in comments.

Specific assignments may provide additional direction on referencing and reuse of third-party code.

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

Most assignments will be submitted via the Moodle electronic learning system, which is accessed through the subject web page. The assignment coversheets will also be made availble through Moodle.

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.

Students will be provided with a Linux-based virtual machine environment. The virtual machine will be made available in labs and can also be installed, using free software, on any PC.

As a virtual machine, it will run under the VMWare Player software (freely downloadable) as any other application under Windows, Mac, or another version of Linux.

Students may choose to use another Linux distribution if they wish, however no support will be provided for this.

Recommended text(s)

Mark G. Sobell. (2009). *A Practical Guide to Linux Commands, Editors, and Shell Programming*. (2nd Edition) Prentice Hall (ISBN: 978-0131367364).

Brian W. Kernighan and Dennis M. Ritchie. (1988). *The C Programming Language*. (2nd Edition) Prentice Hall (ISBN: 978-0131103627).

Michael Kerrisk. (2010). *The Linux Programming Interface*. (1st Edition) No Starch Press (ISBN: 978-159372-200-3).

Larry Wall, Tom Christiansen, Jon Orwant. (2000). *Programming Perl*. (3rd Edition) O'Reilly Media (ISBN: 978-0-596-00027-1).

K. N. King. (2008). *C Programming: A Modern Approach*. (2nd Edition) W. W. Norton & Company (ISBN: 978-0-393-97950-3).

Field trips

No field trips.

Additional subject costs

No additional costs.

Examination material or equipment

Exam details, including permitted equipment, will be announced on the unit website during the semester.

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.ht

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.	The unit covers many theoretical aspects of computing which underpins the foundation of software engineering.
1.2. **Conceptual understanding** of the mathematics, numerical analysis, statistics, and computer and information sciences, which underpin the engineering discipline.	The Unix operating system and C language are two of the major exemplars of operating systems, and system programming languages. The students are also exposed to the concept of shell scripting, a fundamental concept in computing.
1.3. **In-depth understanding** of specialist bodies of knowledge within the engineering discipline.	Students are taught the in-depth theory and practice of the Unix system call interface, the C, shell, and Perl languages, and regular expressions. These are covered in lectures, labs, and assignments.
1.4. **Discernment** of knowledge development and research directions within th engineering discipline.	Not a focus in this unit.
1.5. **Knowledge** of engineering design practice and contextual factors impacting the engineering discipline.	Lecture material discusses the historical context of Unix and C.
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.	Scripting is introduced as a standard method to solve "glue" problems in computer systems. Regular expressions are used as a text-processing tool.
2.2 **Fluent application** of engineering techniques, tools and resources.	Students are taught C, Perl and Shell in the Unix environment, one of the standard programming environments for server computing.
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.	Not covered in this unit.
3.3. **Creative**, innovative and proactive demeanour.	Not specifically covered in this unit.
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							
2	X												
3	X												
4	X												
5	X												
6	X												
7	X												

Relationship between Unit Learning Outcomes and Assessments

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