Unit Outline: KXC251 Algorithms and Metrics

September 2004 - January 2005
Fuzhou, China

Prerequisites
KXC154

Corequisites
None

Unit Weight
12.5% of one academic year

Unit Coordinator
Robert Ollington

Lecturing Staff
Ge Xu (Lecturer and Tutor)
Email: xugeear@hotmail.com

Details of Teaching Arrangements
Lectures: 3 hr/wk
Tutorials: 1 hr/wk (from week 2)

Unit Website
The unit website is accessed from http://webct.utas.edu.au:8900/. You will need to use your University of Tasmania email pop account username and password to log on to the WebCT system. Once authenticated by the system your personalised MyWebCT area will be displayed. It contains links to the websites that you have permission to access - including the website for this unit.
This unit is Web Dependent: communication. This means that you will need to use the Web for this unit. The unit website contains unit information and resources.
If you are not able to access the unit website, please contact the IEN-Start Institute administration office.

University Website
Information and Resources for 'Current Students' are available on the university website at:
http://www.utas.edu.au/students/

Provider
School of Computing - Faculty of Science, Engineering, and Technology.
http://www.comp.utas.edu.au/ienstart

OVERVIEW

Introduction
The primary focus of the unit is on standard algorithms and data structures for solving computing problems, and the analysis of the corresponding run-time and space complexity. The unit starts with an introduction to programming in C (presupposing a knowledge of Java) and ends with some software engineering theory.

Prior Learning
The student is assumed to have a knowledge of programming (in Java), of elementary algorithms and data structures, and of software engineering, as covered in the prerequisite unit, Software Process, KXA154.

Learning Outcomes
On successful completion of this unit, you will be able to:

1. Develop C programs using iteration, recursion, arrays, structs, pointers, dynamically allocated memory and linked data structures, as appropriate.
2. Analyse the run-time and space complexity of algorithms and their associated data structures, using O() and related notation if appropriate.
3. Apply standard algorithms and data structures for storing and searching, for sorting, and for solving graph problems, and demonstrate understanding of the methods’ run-time and space complexity.
4. Demonstrate understanding of standard algorithm design techniques.
5. Demonstrate understanding of the use in software engineering of measurement, cost and schedule prediction, and risk analysis.

Unit Content
Introduction:
- Programming in C
- Algorithm Analysis: O() and related notations
- Lists, Stacks and Queues in C
Storing, Searching and Sorting:
- Trees: Binary Search Trees, AVL trees
- Hashing and Heaps
- Sorting Algorithms

Graphs:
- Graph Searching and Shortest Paths
- Minimum Spanning Trees
- Critical Path Analysis and Network Flow

Algorithm Design Techniques

Software Engineering:
- Measurement, Cost and Schedule Prediction, and Risk Analysis

For more information see the section titled 'Content' on the unit website.

Generic Skills

The university has defined a set of generic graduate attributes expected in its graduates. You can learn more about these attributes at the following link: [http://www.utas.edu.au/policy/subject.html#graduates](http://www.utas.edu.au/policy/subject.html#graduates). Your course is designed to enable you to develop generic skills that are valued in, and expected of, graduates. These are skills that you will need to develop over time. Hence you are encouraged to look for opportunities, as you study each unit, to reflect on and improve these skills.

LEARNING AND TEACHING

Approach to Learning
You are expected to spend about 130 hrs studying in this unit - this includes attendance at scheduled teaching sessions. (For a 13 week semester this is, on average, 10 hr/wk.) This is the amount of study time that the 'typical' student will need to reach the level of competence and understanding required to fulfil the unit objectives.

You are expected to:
- attend all scheduled lectures and tutorials, unless otherwise notified by the unit coordinator
- prepare for, and actively participate in lectures and tutorials
- complete the assigned learning tasks
- review what has been learnt
- complete assessment items and submit them on time
- access and be familiar with the information and resources available on the unit website
- seek help from teaching staff if you have any questions or difficulties in studying this unit

You are encouraged to read the university's Code of Conduct for Teaching and Learning. Part A describes the 'Responsibility of the University to Students' and part B describes the 'Responsibilities of Students to the University'. [http://www.utas.edu.au/tl/policies/codes.html](http://www.utas.edu.au/tl/policies/codes.html)

Schedule
See the 'Schedule' section on the unit website.

Teaching and Support Staff

Teaching Staff

Unit Coordinator:

Robert Ollington
E-Mail: Robert.Ollington@utas.edu.au

Lecturing Staff

Ge Xu (Lecturer and Tutor)
Email: xugeear@hotmail.com

School Help Desk

Contact the IEN-Start Institute administration office if you have any queries or problems with accessing, using, or printing from the computers in the Computer labs.

University Services and Support

The University has staff available to assist you, such as the:
- Learning Development Advisor
- Student Counselor
- Careers Advisor
- Disability Officer
For more information and contact details see the Services and Support section on the University 'Current Students' web page. http://www.utas.edu.au/students/

**Resources**

**Unit Website**

The unit website contains unit information and resources.

**Prescribed Text**


**Readings**

The classic text on the C language is:


Possible sources for some of the Software Engineering material are:


**Software**

The software that you will need to access the unit website and to study this unit, including general purpose software such as word processors, is provided on the computers in the computing labs. If you intend to use software on other computers please check that the versions are compatible.

In this unit, you will use a C compiler with an ANSI-C option.

---

### ASSESSMENT

**Assessment Items**

**Item 1**

**Title:** First Assignment  
**Type:** In-Semester - individual assignment  
**Weighting:** 15%  
**Due:** To be finalised.

This will be a programming assignment. Your submission will have to meet the requirements in the specification. These will include requirements relating to the input and output, and to the compiler and machine on which your submission will be tested.

**Item 2**

**Title:** Second Assignment  
**Type:** In-Semester - individual assignment  
**Weighting:** 15%  
**Due:** To be finalised.

This will be a programming assignment. Your submission will have to meet the requirements in the specification. These will include requirements relating to the input and output, and to the compiler and (School) machine on which your submission will be tested.

**Item 3**

**Title:** 3 hr Examination  
**Type:** Formal Examination  
**Weighting:** 70%  
**Due:** University Examination Period

It is expected that this year’s exam will be similar to last year’s exam in respect of the style of questions, and assessment balance amongst major topics. Further information regarding this year's exam will be given towards the end of the unit.

See the ‘Assessment’ section in unit website for more detailed information about assessment items.
In-Semester Assessment

Unless specifically stated in the specification of the assessment item provided on the unit website, it is required that:

- work submitted by a student is the work of that student alone OR
- where the assessment item is to be completed by a group of students, the work submitted by the group of students is the work of that group of students alone.

Plagiarism

Plagiarism is a form of cheating. It is taking and using someone else's thoughts, writings or inventions and representing them as your own, for example:

- using an author's words without putting them in quotation marks and citing the source;
- using an author's ideas without proper acknowledgment and citation; or
- copying another student's work.

If you have any doubts about how to refer to the work of others in your assignments, please consult your lecturer or tutor for relevant referencing guidelines, and the academic integrity resources on the web at http://www.utas.edu.au/tl/supporting/academicintegrity/index.html.

The intentional copying of someone else's work as one's own is a serious offence punishable by penalties that may range from a fine or deduction/cancellation of marks and, in the most serious of cases, to exclusion from a unit, a course or the University. Details of penalties that can be imposed are available in the Ordinance of Student Discipline – Part 3 Academic Misconduct, see http://www.utas.edu.au/policy/subject.html#students.

The University reserves the right to submit assignments to plagiarism detection software, and might then retain a copy of the assignment on its database for the purpose of future plagiarism checking.

Referencing

The university document on plagiarism contains information about referencing the work or ideas of others. (See http://www.utas.edu.au/plagiarism/.) The preferred text referencing systems for the School is the Harvard system (also referred to as the author-date system).

Submissions

The details of the submission method (paper, electronic or other) for each assignment will be supplied in a separate assignment specification sheet. All in-semester assignment submissions (including electronic submissions) are to include an Assignment Cover Sheet which includes a statement confirming that the submission is your own work. If this undertaking is not signed, the assignment will not be marked. The Assignment Cover Sheet is available on the School's web site http://www.comp.utas.edu.au/app/studyresources.jsp.

Extensions and Penalties

Assessment items will not be accepted after the due date except under the conditions stated in the school policy on late assessment. http://www.comp.utas.edu.au/app/late_assess.jsp

Formal Examination

The formal examination will be held at the IEN-Start Institute, and is conducted by the University Registrar.

Final Grade

Overall assessment will be based on the student's performance throughout the semester as well as in a formal examination. In order to achieve a pass (or better) result, a student must obtain:

1. at least 40% of the total mark for in-semester assessment items
2. at least 40% of the mark for the formal examination
3. at least 50% of the overall mark

Passing grades will be awarded based on the AVCC guidelines:

- PP at least 50% of the overall mark but less than 60%
- CR at least 60% of the overall mark but less than 70%
- DN at least 70% of the overall mark but less than 80%
- HD at least 80% of the overall mark

The maximum mark awarded to a student who fails the unit will be 44.

For more information, including other grades such as Supplementary and Terminating grades, see the School of Computing's guidelines for assessment - available at: http://www.comp.utas.edu.au/app/assess.jsp