**Unit Outline: KXA151 Programming and Problem Solving**

**Semester 2, 2005**

**Sandy Bay Campus, Hobart**

**Newnham Campus, Launceston**

**Prerequisites**
None

**Corequisites**
None

**Unit Weight**
12.5% of one academic year

**Unit Coordinator**
Dr. Julian Dermoudy

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

The Unit Timetable can be accessed from the Study Resources section of the School website. ([http://www.comp.utas.edu.au/app/studyresources.jsp](http://www.comp.utas.edu.au/app/studyresources.jsp)). Lectures for Hobart students will be via video conference.

**Unit Website**
The unit website is accessed from [http://www.utas.edu.au/coursesonline/](http://www.utas.edu.au/coursesonline/). You will need to use your university 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 University IT help desk:
- Entrance Level, Morris Miller Library, Sandy Bay Campus;
- Entrance Level, Launceston Campus Library, Newnham Campus.
- Telephone: 6226 1818 and 1300 304 903.
- Email: servicedesk@utas.edu.au

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

**Provider**
School of Computing - Faculty of Science, Engineering, and Technology.

---

**OVERVIEW**

**Introduction**
Students learn to use a high level language such as Java to write programs which solve problems defined by a program specification. They master fundamental concepts relating to imperative, object-based programming and are introduced to concepts relating to graphical user interfaces and event driven programs. Students are required to demonstrate syntactic, logical and strategic knowledge of the programming constructs introduced in the unit. They are expected to use systematic processes to plan, document, debug and test their programs. Programming exercises are introduced in the context of small problems.

**Warning on Over-confidence**: Some students who have done a considerable amount of home or school computing may think that they are already expert computer programmers. This is extremely unlikely, as most self-taught or uncorrected programmers have picked up bad habits which are inappropriate in professional programming, and may have major gaps in their understanding of concepts. Please bear in mind that practising computing at a professional level is very different from practising it as a hobby. Experience has shown that very few students who have studied computing at school are so good that they can treat programming units lightly.

**Prior Learning**
This unit does have any formal prerequisites. However, students need to navigate the web site for the unit and use a keyboard to write their programs. Students who do not have the basic skills required to "surf" the web or to use a computer keyboard (at about the level required to use a simple word processor) will need to spend extra time early in the semester learning these (simple) skills.

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

1. Write, compile, and run Java programs that contain statements of the types introduced in the unit (see unit content).
   *This demonstrates syntactic knowledge of programming constructs.*

2. Understand the effect of Java statements of the types introduced in the unit (see unit content).
   *This demonstrates conceptual knowledge of programming constructs.*

3. Analyse a problem specification and plan and produce a program which is a solution to the problem and uses Java statements of the types introduced in the unit (see unit content).
   *This demonstrates strategic knowledge of programming constructs.*

4. Use standard techniques to document work. This will include:
   - Appropriate documentation of the programs written during the semester.
   - Formal recording of aspects of activities throughout the process of software development.
   - Production of a record book which documents the activities undertaken in this unit during the semester and is a suitable aide memoire for use in the formal examination.

Unit Content

Introduction:
- unit introduction
- programming terms & tools
- computing tools & terms
- solving problems with computers

Data Storage:
- primitive types
- objects

Objects of prewritten classes:
- object methods
- class methods

Flow of control:
- branches
- planning and implementing branches
- multiway branching
- loops
- implementing loop algorithms
- nesting flow of control

Extending existing classes:
- writing methods
- testing methods
- method parameters and return values
- drawing a GUI

Creating new classes:
- planning
- implementation

Documenting programs:
- purpose of documentation
- internal and external documentation

Structured data - arrays:
- declaring & filling arrays
- using arrays
- arrays - sorting algorithms
- arrays searching algorithms

Graphical User Interfaces (GUI):
- adding components to a GUI
- making a GUI respond to events

Types of errors in programs:
- run time errors - exceptions
- handling exceptions

Recursion:
- concepts
- implementation

Revision:
- OO Concepts summarised
- practical skills
- exam techniques
For more information see the section titled 'Content' on the unit website.

The university has defined a set of generic graduate attributes expected in its 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.

In this unit these skills are specifically targeted:

Knowledge: Students will have the opportunity to begin the acquisition of the knowledge and understanding of computer programming which is a fundamental requirement for all professionals in information technology.

Problem-solving skills: Students learn and practise the fundamental skills needed when attempting to write a computer program that correctly solves a problem that has been set.

Social Responsibility: Students come to understand that assessment tasks that require individual work, must be completed without copying from other students (or other sources).

**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

**Schedule**

See the 'Schedule' section on the unit website.

**Teaching Staff**

**Unit Coordinator:**

Dr. Julian Dermoudy  
E-Mail: Julian.Dermoudy@utas.edu.au  
Phone: (03) 6226 2933  
Room: 351, Sandy Bay Campus, Hobart

**School Help Desk**

Contact the School Help Desk if you have any queries or problems with accessing, using, or printing from the computers in the School of Computing labs.

- **Hobart:** the Help Desk is located near the School's reception desk and is open from 10am - 4pm Monday-Friday. The phone number is 6226 2960.
- **Launceston:** the Help Desk is located near the entrance to the computing labs and is open in the morning from 10am - 12pm, and in the afternoon from 2pm - 4:30pm, Monday-Thursday. On Fridays it is open from 10am - 12pm in the morning and 2pm - 4pm in the afternoon. The phone number is 6324 3447.
- **Burnie:** the computer labs at the NWC are maintained by ITS. Please contact the University Help Desk for assistance. The 6 Macs are maintained by the School of Computing. If you have a query or problem that is specific to the School of Computing please phone the School of Computing Help Desk in Launceston.

**University Services and Support**

The University has staff available to assist you, such as the:

- Learning Development Advisor
- Student Counselor
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/](http://www.utas.edu.au/students/)

**Resources**

**Unit Website**

The unit website contains unit information and resources. Electronic versions of the printed material provided in classes are available on the unit website, as well as administrative information, lecture overheads, tutorial exercises and solutions, and other resources.

Video recordings of the lectures delivered to Burnie in semester 1 will be available on the unit website.

This unit is classified as 'Web dependent: communication' because students will be obliged to use the website for the following things:
- Filling in and submitting participation reports
- Completing practical tests
- Viewing notices about the unit eg. administrative matters, assignment clarifications or hints

**Prescribed Text**


The 2nd edition of this book can still be used in 2005. Do not purchase the 4th edition; it is less suitable than the 3rd, 2nd, or 1st.

**Readings**

Printed copies of the powerpoint slides and the programs discussed in lectures will be available from the on-campus uniprint shopfront in Hobart and Launceston.

A charge is made for these notes to recover the costs of printing. The same material will be available on the unit website.

**NOTE:** These notes are intended as a resource to assist learning in lectures. They WILL NOT work as a substitute for attendance at lectures.

For Hobart students: learning by video conference is rather different from face to face teaching. The leaflet 'The student's guide to video conferencing' provides helpful hints. This is available for download from the site [http://www.utas.edu.au/itr/videoconf/resources.htm](http://www.utas.edu.au/itr/videoconf/resources.htm). Copies of the document are available in the University video conference venues.

Students are not required (and are unlikely to need) to use any resources other than those provided in the text book and the unit materials. If students wish to read more about program development using the Java programming language, there are many standard texts and freely available web sites with relevant information. Students using such resources should be aware that there are many approaches to introducing learners to programming in Java, it is possible that reading a book or web site that takes a different approach from the one used in this unit may increase rather than decrease confusion.

**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 School's computing labs. If you intend to use software on other computers please check that the versions are compatible.

The programming language for this unit is Java 2 (the version used is JDK 1.4.2).

Students who want to work at home will need (at a minimum) this version of Java, a simple text editor, and the files for the packages of Java classes provided especially for this unit. A CD is available (for a small cost) from the Help desk (Hobart and Launceston) which contains Java 1.4.2 and the Java packages required, along with information about how these can be installed on most types of home computer.

**NOTE:**

- Students are not required to have their own computer. There is 24 hour access to suitable computers on campus.
- The School of Computing is not able to provide any technical support for students working on their home computers.
- All work that is submitted for assessment must be on (and work correctly on) the platform provided by the School.

**Computing Facilities**
The School has PC labs (Windows XP), Mac labs (Mac OS-X 10.3), and Networking labs at the Newnham and Sandy Bay campuses. It also maintains 6 Macs (Mac OS-X 10.3) at the NW Centre. Unix accounts can be accessed from all Macs and PCs.

If you have not used these facilities before please contact the School Help Desk to organise your account details. If you would like to access the facilities at the Newnham and Sandy Bay campuses after hours please contact the School Help Desk.

Please contact the School Help Desk if you have difficulty accessing or using these facilities.

All students enrolled in this unit will be issued an account to access the Java software and to store their programs.

The technical information about platforms is as follows:

- Hobart students will use PCs running Java 2 (JDK1.4.2)
- Launceston students will use Macintosh machines running Java 2 (JDK1.4.2) under MacOS X

There is 24 hour access to these machines in Launceston and Hobart. In Burnie, there is 24 hour access to different (but suitable) machines, more information will be provided with the unit materials.

Use of Facilities

Use of computing facilities provided by the School is subject to the School’s Ethics Guidelines - http://www.comp.utas.edu.au/app/ethics.jsp. Copies of the guidelines are also available in all School labs. The School’s facilities may only be used for study-related purposes, and may not be used for personal gain. The playing of games is strictly prohibited in all labs at all times. Before being granted access to the School’s facilities, you will be required to sign a declaration that you have read and understand these guidelines, and that you will abide by them. Disciplinary action may be taken against students who violate the guidelines.

Occupational Health and Safety

The university is committed to providing a safe and secure teaching and learning environment. For more information see http://www.admin.utas.edu.au/hr/ohs/pol_proc/.

ASSESSMENT

<table>
<thead>
<tr>
<th>Assessment Items</th>
<th>Item 1</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td><strong>Title:</strong> Prac Tests</td>
</tr>
<tr>
<td></td>
<td><strong>Type:</strong> In-Semester - test</td>
</tr>
<tr>
<td></td>
<td><strong>Weighting:</strong> 6%</td>
</tr>
<tr>
<td></td>
<td><strong>Due:</strong> At specified times during semester</td>
</tr>
</tbody>
</table>

There will be 3 of these to be assessed during tutorial sessions in designated weeks (see unit schedule for more information). Each prac test will consist of:

- Multiple Choice Questions (MCQ): These will be completed in the tutorial session (a tutor will need to enter a password before the MCQ part of the test can be done).
- A small practical programming task. The program should be written before the tutorial. To mark the program a tutor will run the program, check the form of the code, and ask some questions about the program and the student's record book.

<table>
<thead>
<tr>
<th>Item 2</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Title:</strong> Assignment 1</td>
</tr>
<tr>
<td><strong>Type:</strong> In-Semester - individual assignment</td>
</tr>
<tr>
<td><strong>Weighting:</strong> 9%</td>
</tr>
<tr>
<td><strong>Due:</strong> 3:00pm, Friday 20 August, 2005 (end of week 6)</td>
</tr>
</tbody>
</table>

This will require students to:

- write a small Java program with a single class, all the code will be in the main() method.
- provide some specified items of documentation for the program and the process they followed in producing the program

Students can expect to have 2 - 3 weeks to complete this assignment.
Item 3

**Title:** Assignment 2  
**Type:** In-Semester - individual assignment  
**Weighting:** 15%  
**Due:** 3:00pm, Friday 8 Oct, 2005 (end of week 12)

This will require students to:

- Write code that will form part of a Java program which consists of several interacting classes. The code that students write will be expected to
  - perform correctly when integrated with prewritten code provided as part of the program specification.
  - consist of several methods
  - show good programming style
  - conform with the programming standards and naming conventions expected in this unit
- Provide some specified items of documentation for the program and the process they followed in producing the program

Students can expect to have approximately 6 weeks to complete this assignment.

Item 4

**Title:** Final Exam  
**Type:** Formal Examination  
**Weighting:** 70%  
**Due:** University Examination Period

This will consist of 2 sections.

- **Section A** - Carries 1/6 of the marks and consists of multiple choice questions.
- **Section B** - Carries 5/6 of the marks. Students will be required to answer 5 "long" questions. Each question will require the student to demonstrate their ability to complete some part of a programming and or problem solving exercise.

**NOTE:** The only materials that students will be permitted to take into the formal examination will be the Record book which they have produced during the course of the semester.

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.

Programs that students submit for individual assignments may be compared using "plagiarism detection" programs before they are marked. Students who have assignments that show marked similarity to the assignments of other students may be asked to explain, or in extreme cases may be reported to the Head of School for possible disciplinary action. (See information below on the University policy on plagiarism.)

**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](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](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 from the School Help Desk in Launceston and Hobart, and 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 is conducted by the University Registrar. The 'Current Students' section on the university website contains information about the conduct of, and timetable for, formal examinations.

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