



MONASH University

**FIT9013**  
**Computer programming 2**

**Unit guide**

**Semester 2, 2008**

*Last updated : 18 Jun 2008*

# Table of Contents

<u>FIT9013 Computer programming 2 - Semester 2 , 2008</u> .....	1
<u>Unit leader</u> :.....	1
<u>Lecturer(s)</u> :.....	1
<u>Gippsland</u> .....	1
<u>Introduction</u> .....	2
<u>Unit synopsis</u> .....	2
<u>Learning outcomes</u> .....	2
<u>Workload</u> .....	3
<u>Unit relationships</u> .....	3
<u>Prerequisites</u> .....	3
<u>Relationships</u> .....	4
<u>Continuous improvement</u> .....	5
<u>Student Evaluations</u> .....	5
<u>Unit staff - contact details</u> .....	6
<u>Unit leader</u> .....	6
<u>Lecturer(s)</u> :.....	6
<u>Additional communication information</u> .....	6
<u>Teaching and learning method</u> .....	7
<u>Off-campus distributed learning or flexible delivery</u> .....	7
<u>Communication, participation and feedback</u> .....	7
<u>Unit Schedule</u> .....	7
<u>Unit Resources</u> .....	9
<u>Prescribed text(s) and readings</u> .....	9
<u>Recommended text(s) and readings</u> .....	9
<u>Required software and/or hardware</u> .....	9
<u>Equipment and consumables required or provided</u> .....	9
<u>Study resources</u> .....	10
<u>Library access</u> .....	10
<u>Monash University Studies Online (MUSO)</u> .....	10
<u>Assessment</u> .....	11
<u>Unit assessment policy</u> .....	11
<u>Assignment tasks</u> .....	11
<u>Examinations</u> .....	12
<u>Assignment submission</u> .....	12
<u>Assignment coversheets</u> .....	13
<u>University and Faculty policy on assessment</u> .....	14
<u>Due dates and extensions</u> .....	14
<u>Late assignment</u> .....	14
<u>Return dates</u> .....	14
<u>Plagiarism, cheating and collusion</u> .....	14
<u>Register of counselling about plagiarism</u> .....	15
<u>Non-discriminatory language</u> .....	15
<u>Students with disabilities</u> .....	15
<u>Deferred assessment and special consideration</u> .....	15

# **FIT9013 Computer programming 2 - Semester 2 , 2008**

## **Unit leader :**

Shane Moore

## **Lecturer(s) :**

### **Gippsland**

- Shane Moore

## Introduction

Welcome to FIT9013 Computer Programming 2 for Semester 2, 2008. This 6 point unit is a foundation unit for MAIT and Graduate Certificate students. Following on from FIT9008, this unit is intended to supplement the first programming unit to provide you with an understanding of more advanced concepts in object-oriented programming using the Java programming. This in turn prepares students to go on to the unit GCO3512/FIT9015, which provides a more in-depth study on data structures and algorithms.

## Unit synopsis

This unit introduces more advanced object-oriented programming topics and techniques than its predecessor, and gives students a deeper understanding of programming and data structures and more practical skills in designing, building and testing computer programs.

## Learning outcomes

### Knowledge and Understanding

At the completion of this unit, students will have an understanding of:

- K1. Object-oriented concepts such as inheritance, polymorphism, and abstract classes.
- K2. The implementation in Java of object-oriented concepts such as multiple inheritance.
- K3. How to test a program consisting of many interacting classes.
- K4. The collection classes in the Java API.
- K5. Design principles for building an object-oriented program.
- K6. Problem-solving techniques for debugging an object-oriented program.
- K7. The concept of recursion in a computer program.
- K8. Dynamic data structures.

### Attitudes, Values and Beliefs

At the completion of this unit, students will have attitudes that will allow them to:

- A1. Write programs that conform to programming standards
- A2. Use good design principles when constructing systems
- A3. Take a patient and thorough approach to testing
- A4. Acknowledge any assistance they have received in writing a program
- A5. Search for supplementary Java class-related information in appropriate places when necessary

### Practical Skills

At the completion of this unit, students will be able to:

- P1. Design an object-oriented program consisting of many interacting classes with association, generalization and aggregation relationships.
- P2. Construct a test harness for testing a multiple class object-oriented program.
- P3. Write code to implement a multiple class object-oriented design in Java including association, generalization and aggregation relationships.
- P4. Debug and modify an existing program consisting of many interacting classes.
- P5. Use the Java API classes as part of their programs.
- P6. Use the Java collection classes to store and retrieve data appropriately.
- P7. Use recursion to solve new problems

- P8. Read data from keyboard and files, and write data to screen and files.
- P9. Use exception-handling techniques in programs.
- P10. Use UML to design an object-oriented program.
- P11. Use event-handling techniques in GUI programs.

## Relationships, Communication and TeamWork

At the completion of this unit, students will be able to:

- R1. Document a program correctly
- R2. Produce appropriate documentation for designing and testing a program
- R3. Explain how parts of a program work

## Workload

This is a 6 point unit. At Monash, this means that an average student is expected to spend approximately 12 hours per week, all semester, giving attention to this unit. If you do not spend that much time, you will probably not do so well in this unit as you otherwise might.

For **on campus students**, workload commitments are:

- a weekly two-hour lecture/workshop session, in which concepts will be presented or demonstrated
- a weekly two-hour prac session, in which you will be required to perform problem solving activities to gain practice and new skills
- a minimum of 2 hours of private/personal study in order to satisfy the reading and assignment expectations.

For **Off-campus students**: You generally do not attend lecture and tutorial sessions (but are allowed to if you are near a campus), however, you should plan to spend equivalent time working through the relevant resources and participating in discussion groups each week. Suggested times are listed at the front of each module of the unit study guide.

**All students** will need to allocate time each week (up to 8 hours per week in some weeks), for use of a computer, including time for reading online discussion forums, or doing assignment work.

## Unit relationships

### Prerequisites

Before attempting this unit you must have satisfactorily completed GCO9805 or FIT9008 or equivalent.

From the pre-requisite unit, students commencing FIT9013 are assumed to be able to:

- Use pseudo-code to design algorithms;
- Design object oriented solutions to simple problems using multiple user-defined classes;
- Create and test programming solutions to problems using the Java programming language;
- use fundamental programming elements such as declarations, expressions and statements, control structures, block structure, methods, parameters, data types, and console input and output.
- Edit, compile and execute a computer program;
- Analyse and debug existing programs;
- Write a test plan;
- Produce formal documentation for a program;
- explain the difference between a class and an instance (object)

FIT9013 Computer programming 2 - Semester 2 , 2008

- code a complete, self-contained class in the Java language, including constructor, attributes and methods (mutators and accessors).

## **Relationships**

FIT9013 is an elective in various Masters degrees offered by the Faculty of Information Technology.

It is a prerequisite for FIT9015 and GCO3512.

You may not study this unit if you have already completed any of the following units: GCO9808.

## Continuous improvement

Monash is committed to 'Excellence in education' and strives for the highest possible quality in teaching and learning. To monitor how successful we are in providing quality teaching and learning Monash regularly seeks feedback from students, employers and staff. Two of the formal ways that you are invited to provide feedback are through Unit Evaluations and through Monquest Teaching Evaluations.

One of the key formal ways students have to provide feedback is through Unit Evaluation Surveys. It is Monash policy for every unit offered to be evaluated each year. Students are strongly encouraged to complete the surveys as they are an important avenue for students to "have their say". The feedback is anonymous and provides the Faculty with evidence of aspects that students are satisfied and areas for improvement.

## Student Evaluations

The Faculty of IT administers the Unit Evaluation surveys online through the my.monash portal, although for some smaller classes there may be alternative evaluations conducted in class.

If you wish to view how previous students rated this unit, please go to <http://www.monash.edu.au/unit-evaluation-reports/>

Over the past few years the Faculty of Information Technology has made a number of improvements to its courses as a result of unit evaluation feedback. Some of these include systematic analysis and planning of unit improvements, and consistent assignment return guidelines.

Monquest Teaching Evaluation surveys may be used by some of your academic staff this semester. They are administered by the Centre for Higher Education Quality (CHEQ) and may be completed in class with a facilitator or on-line through the my.monash portal. The data provided to lecturers is completely anonymous. Monquest surveys provide academic staff with evidence of the effectiveness of their teaching and identify areas for improvement. Individual Monquest reports are confidential, however, you can see the summary results of Monquest evaluations for 2006 at <http://www.adm.monash.edu.au/cheq/evaluations/monquest/profiles/index.html>

## Unit staff - contact details

### Unit leader

**Mr Shane Moore**

Lecturer

Phone +61 3 990 26716

**Lecturer(s) :**

**Mr Shane Moore**

Lecturer

Phone +61 3 990 26716

Contact hours : Most days during business hours

### Additional communication information

Unless you have personal enquiries (see below) all communication related to the content of the unit must be via the online Discussion Forums. If you do send the lecturer an email that relates to the content of the unit it may not be answered, or you may be told to ask to the forum.

Personal enquiries include requests for assignment extensions (where warranted by circumstances), special consideration requests, or the need to discuss your personal progress. You are certainly not asked to put anything of a personal nature into your forum postings. Personal matters can also be dealt with by telephone.

On-campus students, and off-campus students who live or work near a campus, may also visit the lecturer at their office.

**Note:** The staff may contact you during the semester, by sending an email to your **@student.monash.edu** address. You are therefore expected to either check that email regularly (at least twice a week), or have it redirect mail to an address which you are going to check regularly. Any email from a student which does not come from your Monash email address can be ignored by the staff member, as sending to other addresses could be a violation of the Privacy provisions of legislation.



## Teaching and learning method

The approach to teaching and learning include a weekly two-hour lecture and a two-hour (laboratory). Additionally, each student should spend a minimum of 8 to 12 hours for personal study (including reading) every week.

Discussion forums are provided as a place where you may ask questions about the content of the unit. You should also use these to clarify the work required in your assignments. They are checked at least twice per week, on Tuesday and Friday mornings, and sometimes even more often than that.

## Off-campus distributed learning or flexible delivery

Students in Singapore may be able to attend classes at TMC. The purpose of these classes is to discuss the tutorial exercises, it is not meant to be a "lecture".

Students in Hong Kong may attend classes at HKU SPACE.

All off-campus students are expected to attempt the weekly exercises, particularly since they lead up to assessments.

You can share your concerns and questions online by posting to the discussion forum.

## Communication, participation and feedback

Monash aims to provide a learning environment in which students receive a range of ongoing feedback throughout their studies. You will receive feedback on your work and progress in this unit. This may take the form of group feedback, individual feedback, peer feedback, self-comparison, verbal and written feedback, discussions (on line and in class) as well as more formal feedback related to assignment marks and grades. You are encouraged to draw on a variety of feedback to enhance your learning.

It is essential that you take action immediately if you realise that you have a problem that is affecting your study. Semesters are short, so we can help you best if you let us know as soon as problems arise. Regardless of whether the problem is related directly to your progress in the unit, if it is likely to interfere with your progress you should discuss it with your lecturer or a Community Service counsellor as soon as possible.

## Unit Schedule

Week	Topic	Study guide	Key dates
1	Revising Java Concepts	Module 1	14/7
2	Designing objects	Module 2	21/7
3	Association/Aggregation relationships	Module 3	28/7 - Week 2 exercises due
4	Inheritance and Polymorphism	Module 4	4/8
5	Interfaces and Abstract Classes	Module 5	11/8
6	GUI and Event Handling	Module 6	18/8 - Week 5 exercises due
7	Exceptions	Module 7	25/8
8	File Input/Output	Module 7 (continued)	1/9
9	Utility Classes	Module 9	8/9
10	Recursion	Module 10	

FIT9013 Computer programming 2 - Semester 2 , 2008

			15/9 - Programming Assignment Due on Fri 19 Sep
11	Dynamic Data Structures, ADTs and Collection Classes	Modules 11 & 12	22/9
Mid semester break			
12	Testing and Debugging	Module 8	6/10 - Week 11 Exercise Due
13	Revision	All Modules	13/10

## Unit Resources

### Prescribed text(s) and readings

#### Prescribed

- Horstmann, C. (2008) "Big Java", 3rd edition, Wiley & Sons. ISBN: 0-470-10554-2

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 text(s) and readings

#### Recommended other references

- Robertson, L. (2007) "Simple program design : a step by step approach", 5th edition, Thomson. ISBN: 0-17-012851-2
- Savitch, W. (2006) "Absolute Java", 2nd edition, Addison Wesley. ISBN: 0-321-33024-2
- Knox-Grant, R. (2005) "Java 2 Beyond the Buttons", 2nd Ed., Greenoil, ISBN: 0-620-33395-2
- Arnold, K., J. Gosling, D. Holmes (2006), "The Java Programming Language", 4th edition, Sun Microsystems/Addison-Wesley. ISBN: 0-321-34980-6

### Required software and/or hardware

#### Prescribed Software

You must have the Java 2 SE SDK version 1.5.0 (also called Java 5) or version 1.6.0 (Java 6) installed on your computer. This is available on the CD-ROM sent to all off-campus students either at the start of semester 1 or 2 each year. It can also be downloaded from the internet by going to <http://java.sun.com/javase/downloads/index.jsp> and selecting the **JDK 6 Update 6** or later.

#### Other Useful Software

##### *JCreator LE*

This is an IDE which provides many useful compilation features. It only works on Windows operating systems. The smallish download can be obtained from <http://www.jcreator.com/download.htm>. Be sure to select the LE version 3.5 file, which is free (unless you want to pay for the more comprehensive version).

##### *TortoiseSVN*

For experienced programmers who want a source-code management system that runs on Windows, TortoiseSVN is strongly recommended. This open-source software can be downloaded from: <http://tortoisesvn.sourceforge.net/downloads>. (Most users should download the very first file, the 32-bit msi file). There are also language-packs for languages other than english. On campus classes might demonstrate this tool at some point in the semester.

### Equipment and consumables required or provided

Students studying off-campus are required to have the minimum system configuration specified by the Faculty as a condition of accepting admission, and regular Internet access. On-campus students, and those studying at supported study locations may use the facilities available in the computing labs. Information about computer use for students

is available from the ITS Student Resource Guide in the Monash University Handbook. You will need to allocate up to **10** hours per week for use of a computer, including time for newsgroups/discussion groups.

## Study resources

Study resources we will provide for your study are:

Study resources we will provide for your study are:

- A printed Unit Book containing 12 Study Guides.
- This *Unit Information Guide* outlining the administrative information for the unit
- The web site on Moodle, shared with FIT2034, where lecture slides, weekly tutorial requirements, assignment specifications, sample solutions and supplementary material will be posted.
- Online Discussion Groups (in moodle).

## Library access

The Monash University Library site contains details about borrowing rights and catalogue searching. To learn more about the library and the various resources available, please go to <http://www.lib.monash.edu.au>. Be sure to obtain a copy of the Library Guide, and if necessary, the instructions for remote access from the library website.

## Monash University Studies Online (MUSO)

All unit and lecture materials are available through MUSO (Monash University Studies Online). Blackboard is the primary application used to deliver your unit resources. Some units will be piloted in Moodle. If your unit is piloted in Moodle, you will see a link from your Blackboard unit to Moodle (<http://moodle.monash.edu.au>) and can bookmark this link to access directly. In Moodle, from the Faculty of Information Technology category, click on the link for your unit.

You can access MUSO and Blackboard via the portal: <http://my.monash.edu.au>

Click on the Study and enrolment tab, then Blackboard under the MUSO learning systems.

In order for your Blackboard unit(s) to function correctly, your computer needs to be correctly configured.

For example:

- Blackboard supported browser
- Supported Java runtime environment

For more information, please visit: <http://www.monash.edu.au/muso/support/students/downloadables-student.html>

**You can contact the MUSO Support by: Phone: (+61 3) 9903 1268**

For further contact information including operational hours, please visit:  
<http://www.monash.edu.au/muso/support/students/contact.html>

Further information can be obtained from the MUSO support site:  
<http://www.monash.edu.au/muso/support/index.html>

## Assessment

### Unit assessment policy

The unit's unsupervised assessment consists of **three (3)** assessable labs and **one (1)** assignment. In addition, there is a supervised three-hour closed book examination. To pass the unit you must:

- complete and submit all labs and the assignment and achieve at least 40% for the total combined assignments result.
- sit for the examination and achieve 40% or more of the marks available from it
- achieve at least 50 after applying the 'final results calculation formula'

If you do not achieve 40% or more in the unit examination or the unit non-examination assessment then a mark of no greater than 44-N will be recorded for your attempt at the unit. Otherwise, you will receive a mark calculated by applying the marks calculation formula.

For example, if you get 70% for assignments portion, but only 39% for the exam portion, you will not be able to get higher than 44-N.

#### Final Result Calculation Formula

The assignments total-marks will contribute 60% of the final result.

The exam will contribute 40% of the final result.

### Assignment tasks

- **Assignment Task**

**Title :** Week 2 Lab Exercises

**Description :**

The exercises comprising the weekly lab for week 2 will be assessed.

**Weighting :** 5%

**Criteria for assessment :**

The specification and marking criteria for the exercise will be distributed in week 2.

**Due date :** Monday 28 July (Week 3)

- **Assignment Task**

**Title :** Week 5 Lab Exercises

**Description :**

The exercises comprising the weekly lab for week 5 will be assessed

**Weighting :** 5%

**Criteria for assessment :**

The specification and marking criteria for the exercise will be distributed in week 5.

**Due date :** Monday 18 August (week 6)

• **Assignment Task**

**Title :** Assignment 1: Major Programming Task

**Description :**

Non-trivial programming assignment

**Weighting :** 25%

**Criteria for assessment :**

The specification and marking criteria will be distributed around week 3.

**Due date :** Friday 19 Sep 2008 (Week 10)

• **Assignment Task**

**Title :** Week 11 Lab Exercises

**Description :**

The exercises comprising the weekly lab for week 11 will be assessed

**Weighting :** 5%

**Criteria for assessment :**

The specification and marking criteria for the exercise will be distributed in Week 11.

**Due date :** Monday 6 October (Week 12)

## Examinations

• **Examination**

**Weighting :** 60%

**Length :** 3 hours

**Type ( open/closed book ) :** Closed book

## Assignment submission

Assignments will be submitted by **electronic** submission. Do not email submissions to the teaching staff. The due date is the date by which the submission must be received for no lateness penalty to apply.

## Assignment coversheets

The work submitted must be your own work. Just as a written assignment requires you to sign a statement that the work is your own work, electronically submitted assignments must also contain a similar declaration. The cover sheet for the Faculty of Information Technology, can be found at <http://www.infotech.monash.edu.au/resources/student/assignments/> . The text of the section titled "Student's Statement" should be typed into a file named **declaration.txt**, to be submitted with your other files.

## University and Faculty policy on assessment

### Due dates and extensions

The due dates for the submission of assignments are given in the previous section. Please make every effort to submit work by the due dates. It is your responsibility to structure your study program around assignment deadlines, family, work and other commitments. Factors such as normal work pressures, vacations, etc. are seldom regarded as appropriate reasons for granting extensions. Students are advised to NOT assume that granting of an extension is a matter of course.

Requests for extensions must be made to the unit lecturer at your campus at least two days before the due date. You will be asked to forward original medical certificates in cases of illness, and may be asked to provide other forms of documentation where necessary. A copy of the email or other written communication of an extension must be attached to the assignment submission.

### Late assignment

Assignments received after the due date will be subject to a penalty of a drop in grade for each 5 day period. Assignments received later than one week after the due date will not normally be accepted, because sample solutions may then be released.

### Return dates

Students can expect assignments to be returned within two weeks of the submission date or after receipt, whichever is later.

Assessment for the unit as a whole is in accordance with the provisions of the Monash University Education Policy at <http://www.policy.monash.edu/policy-bank/academic/education/assessment/>

We will aim to have assignment results made available to you within two weeks after assignment receipt.

### Plagiarism, cheating and collusion

Plagiarism and cheating are regarded as very serious offences. In cases where cheating has been confirmed, students have been severely penalised, from losing all marks for an assignment, to facing disciplinary action at the Faculty level. While we would wish that all our students adhere to sound ethical conduct and honesty, I will ask you to acquaint yourself with Student Rights and Responsibilities (<http://www.infotech.monash.edu.au/about/committees-groups/facboard/policies/studrights.html>) and the Faculty regulations that apply to students detected cheating as these will be applied in all detected cases.

In this University, cheating means seeking to obtain an unfair advantage in any examination or any other written or practical work to be submitted or completed by a student for assessment. It includes the use, or attempted use, of any means to gain an unfair advantage for any assessable work in the unit, where the means is contrary to the instructions for such work.

When you submit an individual assessment item, such as a program, a report, an essay, assignment or other piece of work, under your name you are understood to be stating that this is your own work. If a submission is identical with, or similar to, someone else's work, an assumption of cheating may arise. If you are planning on working with another student, it is acceptable to undertake research together, and discuss problems, but it is not acceptable to jointly develop or share solutions unless this is specified by your lecturer.



Intentionally providing students with your solutions to assignments is classified as "assisting to cheat" and students who do this may be subject to disciplinary action. You should take reasonable care that your solution is not accidentally or deliberately obtained by other students. For example, do not leave copies of your work in progress on the hard drives of shared computers, and do not show your work to other students. If you believe this may have happened, please be sure to contact your lecturer as soon as possible.

Cheating also includes taking into an examination any material contrary to the regulations, including any bilingual dictionary, whether or not with the intention of using it to obtain an advantage.

Plagiarism involves the false representation of another person's ideas, or findings, as your own by either copying material or paraphrasing without citing sources. It is both professional and ethical to reference clearly the ideas and information that you have used from another writer. If the source is not identified, then you have plagiarised work of the other author. Plagiarism is a form of dishonesty that is insulting to the reader and grossly unfair to your student colleagues.

## **Register of counselling about plagiarism**

The university requires faculties to keep a simple and confidential register to record counselling to students about plagiarism (e.g. warnings). The register is accessible to Associate Deans Teaching (or nominees) and, where requested, students concerned have access to their own details in the register. The register is to serve as a record of counselling about the nature of plagiarism, not as a record of allegations; and no provision of appeals in relation to the register is necessary or applicable.

## **Non-discriminatory language**

The Faculty of Information Technology is committed to the use of non-discriminatory language in all forms of communication. Discriminatory language is that which refers in abusive terms to gender, race, age, sexual orientation, citizenship or nationality, ethnic or language background, physical or mental ability, or political or religious views, or which stereotypes groups in an adverse manner. This is not meant to preclude or inhibit legitimate academic debate on any issue; however, the language used in such debate should be non-discriminatory and sensitive to these matters. It is important to avoid the use of discriminatory language in your communications and written work. The most common form of discriminatory language in academic work tends to be in the area of gender inclusiveness. You are, therefore, requested to check for this and to ensure your work and communications are non-discriminatory in all respects.

## **Students with disabilities**

Students with disabilities that may disadvantage them in assessment should seek advice from one of the following before completing assessment tasks and examinations:

- Faculty of Information Technology Student Service staff, and / or
- your Unit Coordinator, or
- [Disabilities Liaison Unit](#)

## **Deferred assessment and special consideration**

Deferred assessment (not to be confused with an extension for submission of an assignment) may be granted in cases of extenuating personal circumstances such as serious personal illness or bereavement. Information and forms for Special Consideration and deferred assessment applications are available at <http://www.monash.edu.au/exams/special-consideration.html>. Contact the Faculty's Student Services staff at your campus for further information and advice.