**FIT1002
Computer programming**

Unit guide

Semester 1, 2009

Table of Contents

FIT1002 Computer programming - Semester 1, 2009

Unit leader :

Angela Carbone

Lecturer(s) :

Berwick

- Matthew Butler

Caulfield

- Dr Angela Carbone

Clayton

- Graham Farr

Gippsland

- Shane Moore

South Africa

- Sheelagh Walton

Malaysia

- Mylini Munusamy

Tutors(s) :

Berwick

- Daniel Waghorn

Introduction

Welcome to FIT1002 Computer Programming semester 1, 2009. This 6 point unit is core to all undergraduate degree programs in the Faculty of IT. The unit has been designed to provide you with an overview of programming, problem solving, testing and debugging. It explores many fundamental programming concepts with emphasis on applying theoretical knowledge to a practical situation.

Unit synopsis

This unit will provide students with an overview of programming and its role in problem-solving and strategies for meeting user requirements and for designing solutions to programming problems. The fundamental programming concepts of the memory model, data types, declarations, expressions and statements, control structures, block structure, modules, parameters and input and output will be applied within the context of objects, attributes, methods, re-use, information-hiding, encapsulation, event-handling and message-passing. Software engineering topics include maintainability, readability, testing, documentation and modularisation.

Learning outcomes

At the completion of this unit students will have a theoretical and conceptual understanding of:

- The relationship between a problem description and program design;
- The semantics of imperative programs;
- The object oriented paradigm as represented by Java;
- The sequence of steps that a computer takes to translate source code into executable code;
- Primitive data types and basic data structures.

At the completion of this unit students will have developed attitudes that enable them to:

- Adopt a problem-solving approach;
- Recognise the importance of programming and documentation conventions;
- Accept the code of professional conduct and practice;
- Act in accordance with best practice, industry standards and professional ethics.

At the completion of this unit students will have the practical skills to:

- Use diagrams to design solutions for programming problems;
- Apply problem solving strategies;
- 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;
- Edit, compile and execute a computer program;
- Analyse and debug existing programs;
- Write a test plan.

At the completion of this unit students will demonstrate the communication skills necessary to:

- Produce formal documentation for a program;
- Explain an existing program.

Workload

For on campus students, workload commitments are:

- two hours of lectures
- one-hour tutorial class in a flat room without computers to discuss theorectical programing concepts and develop problem solving strategies
- two-hour laboratory (practical class) (requiring advance preparation)
- a minimum of 2-3 hours of personal study **per one hour of lecture time** in order to satisfy the reading, tute, prac and assignment expectations.

- You will need to allocate up to 5 hours per week in some weeks, for use of a computer, including time for newsgroups/discussion groups.

Off-campus students generally do not attend lecture, tutorial and laboratory sessions, however, you should plan to spend equivalent time working through the relevant resources and participating in discussion groups each week.

Unit relationships

Prerequisites

There are no prerequisites for this unit.

Relationships

FIT1002 is a core unit in all Faculty of IT undergraduate degrees. There are no prerequisites for this unit. You may not study this unit and CFR2128, CPE1001, CSE 1202, GCO 1811, MMS 1801, MMS 1802, SFT1101 in your degree.

Continuous improvement

Monash is committed to 'Excellence in education' (Monash Directions 2025 - http://www.monash.edu.au/about/monash-directions/directions.html) 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. One of the key formal ways students have to provide feedback is through Unit Evaluation Surveys. The University's Unit Evaluation policy (http://www.policy.monash.edu/policy-bank/academic/education/quality/unit-evaluation-policy.html) requires that every unit offered is 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.

Faculties have the option of administering the Unit Evaluation survey online through the my.monash portal or in class. Lecturers will inform students of the method being used for this unit towards the end of the semester.

Student Evaluations

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

Improvements to this unit

Improvements we have made to the unit from last year include:

1. Introduce methods before constructors

2. Use Moodle instead of Blackboard as the Learning Management System

3. Provide more information and exercises related to plagiarism - aim to reduce plagiarised code.

4. Redevelop some of the tutorial and laboratory exercises to incorporate more testing and debugging of programs;

5. Lecture notes are revised to include one or two thinking activities

6. Some of the lab questions are revised to focus more specifically on testing of classes/ methods/ code and group work

Unit staff - contact details

Unit leader

Dr Angela Carbone
Senior Lecturer
Phone +61 3 990 31911
Lecturer(s) :

Dr Angela Carbone
Senior Lecturer
Phone +61 3 990 31911
Associate Professor Graham Farr
Associate Professor, and Head of School
Phone +61 3 990 55201 +61 3 990 31058
Fax +61 3 990 55146
Contact hours : Monday 11-12am, 3-4pm

Ms Mylini Munusamy
Contact hours : Monday 2-4pm, Tuesday 9-11am, 2-3pm

Mr Shane Moore
Lecturer
Phone +61 3 990 26716
Mr Matthew Butler
Lecturer
Phone +61 3 990 47163
Contact hours : Monday 9-11 am

Sheelagh Walton
Tutor(s) :

Daniel Waghorn
Teaching and learning method

This unit will be delivered via two one hour lectures. Lecturers may go through specific examples, give demonstrations and present slides that contain theoretical concepts.

In tutorials students will discuss in-depth fundamental and interesting aspects about programming which will help them complete their practical work. The tutorials are particularly useful in helping students consolidate concepts and practise their problem solving skills.

Laboratories will be devoted to giving students hands-on experience in implementing a programming solution to a practical problem.

Tutorial allocation

On-campus students should register for tutorials/laboratories using Allocate+.

Off-campus distributed learning or flexible delivery

Off-Campus students should treat the Study Guide Book (consisting of 11 modules) as their primary source for self-directed study. The modules contain text which is directed to leading you through the learning for each week. Please read the welcome message in the Study Guide Book for further detail.

Online Discussion Forums are provided for the primary purpose of enabling off-campus students (including students studying at SPACE in Hong Kong) to engage with each other and the lecturer in Australia. The lecturer will expect all students to read these forums at least twice per week and issues raised in the forum are examinable. In the forums, you may ask questions about the topics or exercises of each module, or to clarify interpretation of assignment tasks and marking criteria.

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	Unit Administration and Introduction	1	
2	Algorithms, Variables and Data Types	2	
3	Using objects and classes, Math Class, String Class, Random Class and I/O	3	
4	Selection	4	Assignment 1 Due End of Lab session (10%)
5	Repetition	5	
6	User defined methods	6	Mid semester test (10%)
Mid semester break			
7	Classes and Objects	7	
8	Class Anatomy; Constructors, mutators and accessors	8	

9	Object refences and Static Class members	9	
10	Arrays (Part I)	10	Assignment 2 Due Start of Lab session (10%)
11	Arrays (Part II)	11	
12	Using multiple classes	11	Assignment 3 Due Fri 29th May(10%)
13	Exam Revision		

Unit Resources

Prescribed text(s) and readings

Lewis J., DePasquale P., and Chase J., *JAVA Foundations*., Pearson Education, 2008, ISBN 0-321-48678-1

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

For all students:

Malik D.S., *Java Programming - From Analysis to Design.*, Thomson Learning 2006, ISBN 0619216085

Robertson LA, *Simple Program Design*, 5th ed., Thomson/Nelson, 2007, ISBN 017010704-3

 For student with advanced programming skills:

Arnold K., Gosling J. & Holmes D., *The Java Programming Language*, Fourth Edition, Addison-Wesley, Upper Saddle River, NJ, 2006.   ISBN 0-321-34980-6 (paperback)

Required software and/or hardware

Java Development Kit, Version j2sdk-1_5_0_06 or later, Sun Microsystems, Inc.

Students should have access to at least one of the following ntegrated Development Environments :

Jcreator - jcreator LE v4.0 is a powerful IDE (Integrated Development Environment) for Java and is strongly recommended. It can be downloaded from the Web Site:

http://www.jcreator.com/

You should download the freeware version. You have no need for the fuller facilities provided in JcreatorPro, and would have to pay for it as well.

jEdit - Text editor written in Java which can auto indent and provides syntax highlighting for more than 130 languages.

http://www.jedit.org/

BlueJ, Version 2.1.2 Programming Development Environment. Although available on CD, version 2.1.3 can be downloaded from

http://www.bluej.org

Eclipse (latest version) downloaded from

http://eclipse.org

Students are encouraged to use JCreator.

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 12 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:

- A MUSO unit web site where lecture slides, weekly tutorial and practical requirements, assignment specifications, and supplementary material will be available
- Discussion groups via MUSO for Off-Campus Learning (OCL) students. The OCL students will liaise with their lecturer via the discussion board for assistance.
- An electronic Unit Book containing the Study Guide Modules for the unit
- This Unit Information outlining the administrative information for the unit

Additionally, students studying by Off-Campus Learning mode will receive:

- A CD-ROM sent at the start of the semester, with software required for their units
- Printed version of the Unit Book containing the Study Guide Modules for the unit.

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.

The Educational Library and Media Resources (LMR) is also a very resourceful place to visit at http://www.education.monash.edu.au/library/

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.

FIT1002 Computer programming - Semester 1, 2009

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 is assessed with three assignments, a one hour mid-semester test and a three hour closed book examination. To pass the unit you must pass each individual hurdle:

- 40% or more in the unit's examination and
- 40% or more in the unit's total non-examination assessment
  and
- an overall unit mark of 50% or more

If a student does 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 the unit.

Furthermore the unit has a hurdle requirement. The hurdle requires students to attend at least **ten** tutorial and **ten** practical classes. Attendance will be monitored. Students unable to attend classes should contact their on-campus lecturer immediately. Students missing classes repeatedly may be contacted by lecturer to explain their absence.

Assignment tasks

- ### Assignment Task

  Title : Assignment 1 - JAVA basics

  Description :

  This assignment will aim to help you to develop programs in Java using classes and objects that are able to read input from the user and format output. You will also learn to use classes and their methods that are available from the Java library.

  Weighting : 10%

  Criteria for assessment :

Detailed assessment criteria will be issued along with the assignment.

1. All prgrams must run and compile correctly. Evidence of testing is required.
2. Programs must meet the problem specification
3. JAVA code should be readable and maintainable and follow the style recommended in the prescribed text book.
4. Programs should be documented
5. Students should be able to answer questions about their own work

Due date : End of Lab week 4

- **Assignment Task**

Title : Assignment 2 - Designing a simple JAVA application

Description :

This assignment will require students to use the selection and iteration control structures and methods.
Weighting : 10%

Criteria for assessment :

Detailed assessment criteria will be issued along with the assignment.

1. All prgrams must run and compile correctly. Evidence of testing is required.
2. Programs must meet the problem specification
3. JAVA code should be readable and maintainable and follow the style recommended in the prescribed text book.
4. Programs should be documented
5. Students should be able to answer questions about their own work during an interview scheduled outside the lab class

Due date : Start of Lab week 10

- **Assignment Task**

Title : Assignment 3

Description :

This assignment will require students to design a class that uses an array.
Weighting : 10%

Criteria for assessment :

Detailed assessment criteria will be issued along with the assignment.

1. All prgrams must run and compile correctly. Evidence of testing is required.
2. Programs must meet the problem specification
3. JAVA code should be readable and maintainable and follow the style recommended in the prescribed text book.
4. Programs should be documented
5. Students should be able to answer questions about their own work during the interview scheduled in week 13.

Due date : Fri 29th May (5pm EST)

- **Assignment Task**

  Title : Test 1

  Description :

  This unit will have a mid-semester test in week 6. On-campus students will sit the test in the lecture, off-campus student will sit the test online via MUSO. The test will include all topics covered in lectures in weeks 1-5.
  Weighting : 10%

  Criteria for assessment :

  Due date : Week 6

Examinations

- **Examination 1**

  Weighting : 60%

  Length : 3 hours

  Type (open/closed book) : Closed book

Assignment submission

Upload zipped Java files via MUSO Assignment tool.

Multiple uploads will be allowed. The latest version uploaded will contain the work that is marked.

Assignment coversheets

Assignment cover sheets must be completed online via MUSO prior to submission and upload of the completed assignment.

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 5% per day, including weekends. Assignments received later than one week (seven days) after the due date will not normally be accepted. In some cases, this period may be shorter if there is a need to release sample solutions.

This policy is strict because comments or guidance will be given on assignments as they are returned, and sample solutions may also be published and distributed, after assignment marking or with the returned assignment.

If you fail to submit an assignment or assessable prac, you will be marked ABSENT (TRANSLATES TO 0), unless...

- If you had an illness or emergency, then If you

  - Obtain Medical Certificate or Police Accident Report
  - Fill out Absentee Form
  - Submit the form and documentation to the General FIT Office then
  - Your mark will be CONSIDERED for ammendment from ABSENT to SICK
  - At the end of the semester:

    - SICK marks are changed to the average of your marks in the assignments, provided you have passed the other assignments

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.