INFS2605
BUSINESS APPLICATION
PROGRAMMING

Course Outline
Semester 1, 2015

Part A: Course-Specific Information

Please consult Part B for key information on Business School policies (including those on plagiarism and special consideration), student responsibilities and student support services.
Table of Contents

PART A: COURSE-SPECIFIC INFORMATION 1
1 STAFF CONTACT DETAILS 1
2 COURSE DETAILS 1
2.1 Teaching Times and Locations 1
2.2 Units of Credit 1
2.3 Summary of Course 1
2.4 Course Aims and Relationship to Other Courses 1
2.5 Student Learning Outcomes 2
3 LEARNING AND TEACHING ACTIVITIES 4
3.1 Learning Activities and Teaching Strategies 4
4 ASSESSMENT 5
4.1 Formal Requirements 5
4.2 Assessment Details 5
5 COURSE RESOURCES 8
6 COURSE EVALUATION AND DEVELOPMENT 8
7 COURSE SCHEDULE 9
PART A: COURSE-SPECIFIC INFORMATION

1 STAFF CONTACT DETAILS

<table>
<thead>
<tr>
<th>Position</th>
<th>Name</th>
<th>Email</th>
<th>Room</th>
<th>Phone</th>
</tr>
</thead>
<tbody>
<tr>
<td>Lecturer-in-charge</td>
<td>Michael Cahalane</td>
<td><a href="mailto:M.Cahalane@unsw.edu.au">M.Cahalane@unsw.edu.au</a></td>
<td>2085</td>
<td>+61 (2) 9385 4473</td>
</tr>
<tr>
<td>Tutors</td>
<td>TBA</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

LIC Consultation Time: Thursday, 12:00-14:00 (by appointment only)

The preferred method of contacting your lecturer or tutor is through e-mail. Your UNSW e-mail account should be used for formal notices and correspondence regarding the course. For security reasons, please avoid using e-mails from anonymous accounts, such as Yahoo, Hotmail, and Gmail. Always start the subject line of e-mails with INFS2605 and sign the e-mail with your full name and student number.

2 COURSE DETAILS

2.1 Teaching Times and Locations
The Time: Thu 16:00 - 17:00
The Location: Old Main Building 145 (K-K15-145)


2.2 Units of Credit
The course is worth 6 units of credit. There is no parallel teaching in this course.

2.3 Summary of Course
This aim of this course is to progress students' knowledge and skills in relation to Java programming concepts and principles, within a business context. The course builds on the knowledge and skills acquired in INFS1609; and is also a prerequisite to INFS3605 (Information Systems Project 2). Students will learn to design solutions for a broad range of problems and implement those solutions in the form of small/medium sized applications, using appropriate Java programming techniques and tools. The course will introduce topics such as testing, peer-programming, user interface development, interfacing with relational databases, creating basic reporting functionality, software development processes, and the general use of tools to build business information systems.

2.4 Course Aims and Relationship to Other Courses
This course covers material that is significant to the discipline of Information Systems. It assumes both a completion and a thorough understanding of the core information systems course INFS1609 Fundamentals of Business Programming. The course will also prepare students for INFS3605 (Information Systems Project 2). In addition to
developing students’ practical programming skills, this course briefly introduces and compares different software development methodologies. From this, students will obtain a good conceptual understanding of the different development processes used throughout industry. This course also aims to expand students’ ability to work collaboratively in solving problems through the application and evaluation of peer-programming techniques. Overall, this course aims to provide students with various concepts and skills that are essential in careers such as project managers, business analysts, systems analysts, designers, and developers.

2.5 Student Learning Outcomes

By the end of this course, you should be able to:

1. Apply abstraction mechanisms for increasing program clarity and reusability
2. Apply your skills and knowledge to write and identify clear, reliable, well-structured, and well-documented programs in Java
3. Evaluate and provide solutions to small to medium scale problems
4. Design programs that interface with relational databases
5. Design programs that generate reporting functionality
6. Explain, apply, and evaluate Pair Programming methodology
7. Explain and apply Unit Testing
8. Discuss different software development processes

The Course Learning Outcomes are what you should be able to DO by the end of this course if you participate fully in learning activities and successfully complete the assessment items.

The Learning Outcomes in this course also help you to achieve some of the overall Program Learning Goals and Outcomes for all undergraduate students in the UNSW Business School. Program Learning Goals are what we want you to BE or HAVE by the time you successfully complete your degree (e.g. ‘be an effective team player’). You demonstrate this by achieving specific Program Learning Outcomes - what you are able to DO by the end of your degree (e.g. ‘participate collaboratively and responsibly in teams’).

The Course Learning Outcomes are what you should be able to DO by the end of this course if you participate fully in learning activities and successfully complete the assessment items.

The Learning Outcomes in this course also help you to achieve some of the overall Program Learning Goals and Outcomes for all undergraduate students in the Business School. Program Learning Goals are what we want you to BE or HAVE by the time you successfully complete your degree (e.g. ‘be an effective team player’). You demonstrate this by achieving specific Program Learning Outcomes – what you are able to DO by the end of your degree (e.g. ‘participate collaboratively and responsibly in teams’).

For more information on the Undergraduate Coursework Program Learning Goals and Outcomes, see Part B of the course outline.
### Business Undergraduate Program Learning Goals and Outcomes

1. **Knowledge**: Our graduates will have in-depth disciplinary knowledge applicable in local and global contexts. You should be able to select and apply disciplinary knowledge to business situations in a local and global environment.

2. **Critical thinking and problem solving**: Our graduates will be critical thinkers and effective problem solvers. You should be able to identify and research issues in business situations, analyse the issues, and propose appropriate and well-justified solutions.

3. **Communication**: Our graduates will be effective professional communicators. You should be able to:
   - Prepare written documents that are clear and concise, using appropriate style and presentation for the intended audience, purpose and context, and
   - Prepare and deliver oral presentations that are clear, focused, well-structured, and delivered in a professional manner.

4. **Teamwork**: Our graduates will be effective team participants. You should be able to participate collaboratively and responsibly in teams, and reflect on your own teamwork, and on the team’s processes and ability to achieve outcomes.

5. **Ethical, social and environmental responsibility**: Our graduates will have a sound awareness of the ethical, social, cultural and environmental implications of business practice. You should be able to:
   - Identify and assess ethical, environmental and/or sustainability considerations in business decision-making and practice, and
   - Identify social and cultural implications of business situations.

The following table shows how your Course Learning Outcomes relate to the overall Program Learning Goals and Outcomes, and indicates where these are assessed (they may also be developed in tutorials and other activities):

<table>
<thead>
<tr>
<th>Program Learning Goals and Outcomes</th>
<th>Course Learning Outcomes</th>
<th>Course Assessment Item</th>
</tr>
</thead>
<tbody>
<tr>
<td>This course helps you to achieve the following learning goals for all Business undergraduate coursework students:</td>
<td>On successful completion of the course, you should be able to:</td>
<td>This learning outcome will be assessed in the following items:</td>
</tr>
<tr>
<td><strong>1</strong> Knowledge</td>
<td>Apply abstraction mechanisms for increasing program clarity and reusability</td>
<td>• Tutorial Problems</td>
</tr>
<tr>
<td></td>
<td>Explain, apply, and evaluate Pair Programming methodology</td>
<td>• Assignment 1 &amp; 2</td>
</tr>
<tr>
<td></td>
<td>Explain and apply Unit Testing</td>
<td>• In-tutorial Tests</td>
</tr>
<tr>
<td></td>
<td>Discuss different software development processes</td>
<td>• Exam</td>
</tr>
<tr>
<td><strong>2</strong> Critical thinking and problem solving</td>
<td>Evaluate and provide solutions to small to medium scale problems</td>
<td>• Tutorial Problems</td>
</tr>
<tr>
<td></td>
<td>Design programs that interface with</td>
<td>• Assignment 1 &amp; 2</td>
</tr>
<tr>
<td></td>
<td></td>
<td>• In-tutorial Tests</td>
</tr>
</tbody>
</table>
3 LEARNING AND TEACHING ACTIVITIES

3.1 Learning Activities and Teaching Strategies

At university the focus is on self-directed search for knowledge. Lectures, tutorials, textbooks and other resources all facilitate this process. Students will need to prepare for this course by revising their knowledge and skills developed in INFS1609. To facilitate your revision, a brief review of Java fundamentals will be provided at the start of this course. However, students will need create and engage with their own revision plan.

The lectures will outline the main concepts and methods for this course. Each week, the LIC will begin by reviewing and clarifying material previously covered. The LIC will then introduce a new topic, highlighting relevant study material (e.g. texts, videos, etc) and presenting students with programming exercises to be completed before the following week’s tutorial. On occasion, the LIC will use the lecture time to pose questions to students and hold class discussions on topics covered. The relevant study material, to be read in your own time, provides more detail about the topics introduced in the lecture. It is expected that you will spend approximately 10 hours per week studying for this course. This time should be made up of reading, revision, working on exercises and problems, and attending classes (lectures and tutorials). In periods where you need to complete assignments or prepare for examinations, the workload may be greater.

This course requires considerable out-of-class reading and programming / problem solving exercises.

The tutorials will be used to reinforce and apply material covered in lectures and study material. Tutorials are an important part of your learning for INFS2605; therefore, being prepared for your tutorials is essential. Student should routinely check what material they are expected to read/complete prior to each session. This includes completing any activities you have been asked to do in preparation for your next tutorial as well as reviewing your lecture notes from your previous lecture.
Tutorials also give you the opportunity to discuss your work with fellow students, and hence gain an indication of your own progress. Students should also use their tutorial time to ask questions for clarifications on the material covered in class as well as their study material.

Over the semester, you will engage in a variety of different problem-solving scenarios that build in complexity and that call for different combinations of knowledge and skills. During the semester tutorials will: (i) assess students’ knowledge through quizzes; and (ii) present students will problems that will require students (both individuals and teams) to design and implement appropriate programming solutions.

This semester the INFS2605 course will use the smart student response system Socrative to (i) conduct weekly coursework quizzes/tests during tutorials and (ii) receive feedback from students. It is important that all students have access to Socrative, therefore, students are encouraged to bring their web-enabled devices to lectures and tutorials (e.g. smart phones, tablets, and laptops). The Socrative app is available in Google Play and iTunes stores. In the event that a web-enabled device is not available, students may be given traditional paper-based quizzes/feedback forms.

In addition, Moodle will be used to facilitate online discussions, post video tutorials, as well as general announcements. Students are responsible for checking Moodle on a regular basis.

4 ASSESSMENT

4.1 Formal Requirements
In order to pass this course, you must:
- achieve a composite mark of at least 50; and
- make a satisfactory attempt at all assessment tasks. This can be interpreted as attaining a mark of 45% or more in each assessment item.

4.2 Assessment Details

<table>
<thead>
<tr>
<th>Assessment Task</th>
<th>Weighting</th>
<th>Length</th>
<th>Due Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Tutorial Preparation and Participation</td>
<td>10%</td>
<td>See below</td>
<td>Tutorials, Weeks 3-12</td>
</tr>
<tr>
<td>2. Weekly Quiz (1-10)</td>
<td>20%</td>
<td>See below</td>
<td>Tutorials, Weeks 3-12</td>
</tr>
<tr>
<td>3. Group Assignment</td>
<td>30%</td>
<td>See below</td>
<td>Week 10 Mon 11 May</td>
</tr>
<tr>
<td>4. Final Examination</td>
<td>40%</td>
<td>2 hours</td>
<td>University Exam Period</td>
</tr>
<tr>
<td>Total</td>
<td>100%</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Tutorial Preparation and Participation
Tutorials will be used to reinforce material covered in lectures as well as study material. Each tutorial will involve a number of problems/exercises which relate to a given topic. During weeks 3-12, active participation in tutorials is included as part of student assessment for INFS2605. Active participation in each tutorial during this period has a weighting of 1%, therefore, tutorial participation has a weighting of 10% in total.

Active participation includes, but is not limited to: providing programming solutions for exercises; working in an assigned group; engaging in tutorial discussions; asking and answering questions; and taking notes. Students will also be required to sign an attendance sheet each week.

Please note: All students are expected to be punctual and to adhere to their allocated tutorial times. Latecomers may not be awarded an assessment mark. Students are required to prepare for each tutorial and the tutorial will require your full participation. Students who are not prepared for a tutorial and/or are not fully engaged during the tutorial itself (e.g. occupied with social networking, surfing the web, checking mail, etc.) may not be awarded an assessment mark or may receive a partial mark.

Quizzes
During weeks 3-12, students will be assessed by way of a weekly quiz. Each individual quiz has a weighting of 2%, therefore, quizzes for INFS2605 have a weighting of 20% in total. The content of each quiz will primarily reflect the study material covered by students during the previous week. The aim of this assessment is to ensure that students are engaged in continual learning throughout the semester, and are prepared for each tutorial workshop.

Please note: To ensure that students truly comprehend the material covered throughout the course, some questions may relate to material covered in previous weeks. Therefore, students are advised to engage in on-going study and revision throughout the semester.

Group Assignment
The group assignment will incorporate a medium size programming task. The assignments for INFS2605 have a weighting of 30%.

Through this assignment students will demonstrate their ability to understand and implement a range of technical skills relevant to the course. The assignments will be in line with the topics covered in the lectures, tutorials, and study material.

The assignment will incorporate pair-programming and other teamwork activities. Teams will typically consist of two students each. Students will be required to evaluate their teammate’s contribution to the assignment (rating 0-5). Individual contribution weighting will impact the final grade given to each student for their assignments.

Further information regarding the specific details and submission procedure for the assignment will be posted on Moodle as well as discussed during lectures and/or tutorials. However, before submitting the assignment solution, students must ensure that their program compiles and runs as intended. Your solution for the programming assignment will require appropriate use of commenting and naming conventions.

Please Note: Each student is responsible for their own work. The assignment will be checked for evidence of plagiarism. Plagiarism includes copying, inappropriate paraphrasing, collusion, as well as self-plagiarism. For instance, deliberately or
recklessly presenting your work in collusion with others, copying or stealing another student’s assignment, or paying for work to be done may all be considered acts of Level 3 Plagiarism. This constitutes as significant plagiarism and serious student misconduct that is in breach of the Student Code.

All students should keep a copy of all work submitted for assignments. Peer assessment will be conducted for work relating to the group assignment; this process will impact the grade/mark students receive for their submitted work. The LiC will have the final say/decision on matters relating to peer assessment. More information on the peer assessment process will be discussed during the semester.

Final Examination
A final written examination will take place during the University Exam Period. The examination time will be 2 hours. The examination is worth 40% of the total marks for this course. Candidates may not bring any course materials to the examination. The examination paper may not be retained by the candidate.

Late Submission
It is your responsibility to adhere to the procedures for submission of assignments otherwise a penalty may apply. The key requirements are:

The group assignments shall be submitted as indicated in the course schedule and according to the instructions of the lecturer-in-charge.

The late submission of assignments carries a penalty of 10% of the maximum marks for that assignment per day of lateness (including weekends and public holidays), unless an extension of time has been granted. For example, an assignment worth 20% will attract a 2-mark penalty per day. An extension in the time of submission will only be granted by the lecturer-in-charge for exceptional circumstances, such as misadventure or illness. There are also provisions for Special Consideration – see later in PART B “Special Consideration”. Applications should be made to the lecturer-in-charge by email or in person. You will be required to substantiate your application with appropriate documentary evidence such as medical certificates, accident reports etc. Please note that work commitments and computer failures are usually considered insufficient grounds for an extension.

Partial submissions of your assignment work will not be accepted.

Quality Assurance
The Business School is actively monitoring student learning and quality of the student experience in all its programs. A random selection of completed assessment tasks may be used for quality assurance, such as to determine the extent to which program learning goals are being achieved. The information is required for accreditation purposes, and aggregated findings will be used to inform changes aimed at improving the quality of Business School programs. All material used for such processes will be treated as confidential.
5 COURSE RESOURCES
The website for this course is on Moodle at:

The required textbook for this course is:

6 COURSE EVALUATION AND DEVELOPMENT
Each year feedback is sought from students and other stakeholders about the courses offered in the School and continual improvements are made based on this feedback. UNSW’s Course and Teaching Evaluation and Improvement (CATEI) Process is one of the ways in which student evaluative feedback is gathered. In this course, we will seek your feedback through end of semester CATEI evaluations.

In addition, throughout the semester we will also use the Socrative App to gage students understanding of topics as well as receive general feedback on material covered in lectures and tutorials.
# 7 COURSE SCHEDULE

Lectures start in Week 1 and finish in Week 12. Please note: Some variations may occur – always check Moodle Announcements

<table>
<thead>
<tr>
<th>Week</th>
<th>Lecture Topic</th>
<th>Tutorial Topic</th>
<th>Other Activities/Assessment</th>
</tr>
</thead>
<tbody>
<tr>
<td>Week 1</td>
<td><strong>Introduction</strong></td>
<td><strong>NO TUTORIALS</strong></td>
<td></td>
</tr>
<tr>
<td>02 March</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Week 2</td>
<td><strong>Objects, Classes, &amp; Strings</strong></td>
<td><strong>Introduction + Java Fundamentals I</strong></td>
<td></td>
</tr>
<tr>
<td>09 March</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Week 3</td>
<td><strong>Inheritance, Polymorphism, &amp; Interfaces</strong></td>
<td><strong>Java Fundamentals II</strong></td>
<td><strong>Quiz 1</strong></td>
</tr>
<tr>
<td>16 March</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Week 4</td>
<td><strong>User Interface</strong></td>
<td><strong>Objects, Classes, &amp; Stings</strong></td>
<td><strong>Quiz 2</strong></td>
</tr>
<tr>
<td>23 March</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Week 5</td>
<td><strong>User Interface</strong></td>
<td><strong>Inheritance, Polymorphism, &amp; Interfaces</strong></td>
<td><strong>Quiz 3</strong></td>
</tr>
<tr>
<td>30 March</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Week 6</td>
<td><strong>Exception Handling</strong></td>
<td><strong>User Interface</strong></td>
<td><strong>Quiz 4</strong></td>
</tr>
<tr>
<td>13 April</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Week 7</td>
<td><strong>Database</strong></td>
<td><strong>Exception Handling</strong></td>
<td><strong>Quiz 5</strong></td>
</tr>
<tr>
<td>20 April</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Week 8</td>
<td><strong>Reporting</strong></td>
<td><strong>Database</strong></td>
<td><strong>Quiz 6</strong></td>
</tr>
<tr>
<td>27 April</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Week 9</td>
<td><strong>Event-Driven Programming</strong></td>
<td><strong>Reporting</strong></td>
<td><strong>Quiz 7</strong></td>
</tr>
<tr>
<td>04 May</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Week 10</td>
<td><strong>Database 2</strong></td>
<td><strong>Event-Driven Programming</strong></td>
<td><strong>Quiz 8 Group Assignment</strong></td>
</tr>
<tr>
<td>11 May</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Week 11</td>
<td><strong>TBA</strong></td>
<td><strong>Database 2</strong></td>
<td><strong>Quiz 9</strong></td>
</tr>
<tr>
<td>18 May</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Week 12</td>
<td><strong>Review</strong></td>
<td><strong>TBA</strong></td>
<td><strong>Quiz 10</strong></td>
</tr>
<tr>
<td>25 May</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Week 13</td>
<td><strong>NO LECTURE</strong></td>
<td><strong>Review</strong></td>
<td></td>
</tr>
<tr>
<td>1 June</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>