

INFS2605 INTERMEDIATE BUSINESS PROGRAMMING

Course Outline Semester 1, 2017

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
<u>1 STAFF CONTACT DETAILS</u>	<u>1</u>
<u>2 COURSE DETAILS</u>	<u>1</u>
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	2
2.5 Student Learning Outcomes	2
<u>3 LEARNING AND TEACHING ACTIVITIES</u>	<u>4</u>
3.1 Learning Activities and Teaching Strategies	4
<u>4 ASSESSMENT</u>	<u>5</u>
4.1 Formal Requirements	5
4.2 Assessment Details	5
<u>5 COURSE RESOURCES</u>	<u>7</u>
<u>6 COURSE EVALUATION AND DEVELOPMENT</u>	<u>7</u>
<u>7 COURSE SCHEDULE</u>	<u>8</u>

PART A: COURSE-SPECIFIC INFORMATION

1 STAFF CONTACT DETAILS

Position	Name	Email	Room	Phone
LIC	Dr Ben Choi	chun.choi@unsw.edu.au	QUAD 2113	9385 9843

LIC Consultation Time: Tuesday, 12:00-13:00 (by appointment only)

The preferred method for 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-mail 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: Monday, 1600 – 1700 or Tuesday, 1100 – 1200

The Location: Colombo ThB

Tutorials start in Week 2 (to Week 13). A full list of tutorials, times and tutors will be on Moodle.

Day	Time	Location	Tutor
Monday	1700 – 1900	Mat 211	Morgan
Tuesday	0900 – 1100	Quad Lab 1	Yenni
	0900 – 1100	Quad 2082	Jacob
	1100 – 1300	Quad Lab 1	Yenni
	1800 – 2000	Quad Lab 1	Yenni
Wednesday	0900 – 1100	Quad Lab 1	Chun-Tang
Thursday	1800 – 2000	Quad Lab 1	Jacob

2.2 Units of Credit

The course is worth 6 units of credit.

2.3 Summary of Course

The 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, 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:

1. Be able to apply abstraction mechanisms for increasing program clarity and reusability
2. Be able to write and identify clear, reliable, well-structured, and well-documented programs in Java
3. Be able to evaluate and provide information system solutions to small to medium scale problems
4. Be able to design programs that interface with relational databases
5. Be able to explain, apply, and evaluate Pair Programming methodology

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 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:

- a. Prepare written documents that are clear and concise, using appropriate style and presentation for the intended audience, purpose and context, and
- b. 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:

- a. Identify and assess ethical, environmental and/or sustainability considerations in business decision-making and practice, and
- b. 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):

Program Learning Goals and Outcomes		Course Learning Outcomes	Course Assessment Item
<i>This course helps you to achieve the following learning goals for all Business undergraduate students:</i>		<i>On successful completion of the course, you should be able to:</i>	<i>This learning outcome will be assessed in the following items:</i>
1	Knowledge	Apply abstraction mechanisms for increasing program clarity and reusability Explain, apply, and evaluate Pair Programming methodology	<ul style="list-style-type: none"> • Tutorial Problems • Group Assignment • Exam
2	Critical thinking and problem solving	Evaluate and provide information system solutions to small to medium scale problems Design programs that interface with relational databases	<ul style="list-style-type: none"> • Tutorial Problems • Group Assignment • Exam
3a	Written communication	Write and identify clear, reliable, well-structured, and well-documented programs in Java	<ul style="list-style-type: none"> • Tutorial Problems • Group Assignment • Exam
3b	Oral communication	Not specifically addressed in this course	<ul style="list-style-type: none"> • Tutorials • Group Assignment
4	Teamwork	Explain, apply, and evaluate Pair Programming methodology	<ul style="list-style-type: none"> • Tutorial Problems • Group Assignment
5a	Ethical, social and environmental responsibility	Not specifically addressed in this course	

5b	Social and cultural awareness	Not specifically addressed in this course	
----	-------------------------------	---	--

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 mid-semester test; and (ii) present students will problems that will require students (both individuals and teams) to design and implement appropriate programming solutions.

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

To receive a pass grade in this course, you must meet ALL of the following criteria:

- Attain an overall mark of at least 50%.
- Attend at least 80% of all scheduled classes.
- Attain a satisfactory performance in each component of the course. A mark of 45 percent or higher is normally regarded as satisfactory.
- Attain a mark of at least 45% in the final exam.

The School reserves the right to scale final marks to a mean of 60%.

4.2 Assessment Details

Assessment Task	Weighting	Length	Due Date
Tutorial Preparation and Participation	10%	See below	Tutorials, Weeks 2-12
Mid-Semester Test	20%	See below	Week 6
Group Assignment	20%	See below	Week 12 and Week 13
Final Examination	50%	2 hours	University Exam Period
Total	100%		

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 2-12, active participation in tutorials is included as part of student assessment for INFS2605.

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 during tutorials.

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.

Mid-Semester Test

This semester there will be a mid-semester test, worth 20%. The content of this test will focus primarily on java fundamentals as well as material covered on object-oriented programming principles (in particular, abstraction, polymorphism, inheritance and encapsulation). This test is designed to make sure that students are actively engaging with the reading material and tutorial work covered during the start of the semester. This test will be held during lecture time in week 6.

Group Assignment

The group assignment will incorporate a medium size development task.

Through this assignment students will demonstrate their ability to understand and implement a range of skills relevant to the course. The assignments will be in line with the topics covered in the lectures, tutorials, and study material. **Further details regarding the assignment were provided on Moodle during week 1.**

Teams will typically consist of four students each. Students will be required to evaluate their teammate's contribution to the assignment. 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.

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 50% 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:

<https://moodle.telt.unsw.edu.au/login/index.php>

The required textbook for this course is:

Introduction to Java Programming, Brief Version by Y. Daniel Liang (**10th edition**).
ISBN: 9781292078564

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 MyExperience survey is one of the ways in which student evaluative feedback is gathered. In this course, we will seek your feedback through end of semester MyExperience survey.

7 COURSE SCHEDULE

Lectures start in Week 1 and finish in Week 13.

Please note: Some variations may occur – always check Moodle Announcements

COURSE SCHEDULE			
Week	Topic	Tutorial Topic	Other activities / assessment
Week 1 27 February	Lecture 1: Java Fundamentals	<i>NO TUTORIALS</i>	<i>Group Assignment Details Released</i>
Week 2 6 March	Lecture 2: Java Fundamentals	Tutorial 1: Java Fundamentals	
Week 3 13 March	Lecture 3: Inheritance & Abstract Classes	Tutorial 2: Java Fundamentals	
Week 4 20 March	Lecture 4: Polymorphism & Interfaces	Tutorial 3: Inheritance & Abstract Classes	
Week 5 27 March	Lecture 5: JavaFX	Tutorial 4: Polymorphism & Interfaces	
Week 6 03 April	Lecture 6: In-class Test	Tutorial 5: JavaFX	Mid-Semester Test
Week 7 10 April	Lecture 7: Exception Handling	Tutorial 6: JavaFX and Group Assignment	1 st Internal review
Mid-semester break: Friday 14 – Saturday 22 April inclusive			
Week 8 24 April	(Tuesday 25 April is Anzac Day public holiday) NO LECTURE	<i>NO TUTORIALS</i>	
Week 9 1 May	Lecture 8: Event-Driven Programming	Tutorial 7: Exception Handling and Group Assignment	
Week 10 8 May	Lecture 9: Database (Relational DB and SQL)	Tutorial 8: Event-Driven Programming and Group Assignment	
Week 11 15 May	Lecture 10: Database (JDBC)	Tutorial 9: Database 1 and Group Assignment	
Week 12 22 May	Lecture 11: HCI	Tutorial 10: Database 2 and Group Assignment Due	Group Assignment
Week 13 29 May	Lecture 12: Review	Assignment Demo	2 nd Internal review