Software Development - Course Outline
SWEN 221: 2016 Trimester 1

This document sets out the workload and assessment requirements. It also provides contact information for staff involved in the course. If the contents of this document are altered during the course, you will be advised of the change by an announcement in lectures and/or on the course web site. A printed copy of this document is held in the School Office.

The Course

This course develops a deeper understanding of object-oriented programming and associated practices. The focus is on programming techniques at the micro scale. Topics include: inheritance, polymorphism, genericity, error handling, testing and debugging. A sequence of short assignments will develop the key ideas and practices; rigour in testing will be developed through (automated) assessment of programme correctness.

The prerequisite for SWEN221 is COMP103. If you have not passed this course recently, you should contact the Course Coordinator as soon as possible.

Objectives

By the end of the course, students should be able to:

1. Implement, test, and debug object-oriented programs using all the facilities of the Java programming language, including exceptions, generics, assertions, and concurrency; (3(b),3(c),3(f))
2. Appreciate the range of programming techniques available, the strengths and weaknesses of particular techniques, and be able to choose and apply an appropriate programming technique to a given problem; (3(b),3(f))
3. Understand the range of programming tools available, their different strengths and weaknesses, and be able to choose the appropriate tools for a software development task; (3(b),3(f))
4. Use appropriate programming tools, such as development environments, test harnesses, frameworks, and libraries. (3(b),3(f))

The assignments, labs and lectures will contribute to all learning objectives.

Note: SWEN 221 is part of the Engineering program at Victoria University of Wellington. BE students are expected to exhibit a number of graduate attributes upon graduation. These course objectives contribute to the graduate attributes as indicated above. A full table of these attributes is available at Graduate Attributes.

Staff

The course coordinator is David J. Pearce. The lecturers for the course are David Pearce and Marco Servetto. Their contact details are:

- *David J. Pearce*
- Cotton 231
- +64 4 463 5833
- djp@ecs.vuw.ac.nz
- Office Hours: Tuesday 3-5pm, Wednesday 3-5pm

- *Marco Servetto*
- Cotton 258
- +64 4 463 5820
- marco.servetto@ecs.vuw.ac.nz
- Office Hours: Tuesday 3-5pm, Wednesday 3-5pm

The Class Rep for SWEN221 this year is:

- *Nanda Hibatullah*
- *hibatunand@ecs.vuw.ac.nz*

Announcements and Communication

The main means of communication outside of lecture will be the web area at http://ecs.victoria.ac.nz/Courses/SWEN221_2016T1/. There you will find, among other things, this document, the lecture

There you will find, among other things, this document, the <u>lecture schedule</u>, and the <u>SWEN 221 Forum</u>. The forum is a web-based bulletin board system. Questions and comments can be posted to the forum, and staff will read these posts and frequently respond to them.

Important announcements for the course will be made at lectures, emailed to the course mailing list and posted on the SWEN 221 web site. We will assume that all students attend all lectures and check the web site and their ECS email at least three times a week.

Trimester Dates and Lecture Times

SWEN 221 is a trimester 1 course. The trimester starts on Monday 29th Februrary. The examination period at the end of the course is 10 June - 29 June,

A <u>schedule</u> of lecture topics, readings, and assignment due dates is available online. A summary of the topics is:

Topic	Lectures
Testing + Debugging	5
Inheritance	2
Polymorphism	2
Generics	2
Concurrency	2
Other Features	4

Lectures are 2pm-3pm on Tuesdays in MCLT101 and 2pm-3pm on Wednesdays in Hunter HULT323 (see the <u>University Timetable</u> for more information). Additionally, optional tutorials will occasionally be run on Fridays from 2pm-3pm in Maclaurin MCLT101.

Attendance at lectures is an essential part of the learning process. Lectures will be used to introduce basic ideas and techniques.

In addition to attending lectures, we expect you to spend at least one hour a week preparing for them by reading the suggested chapters of the textbook. These can be found in the Schedule.

Textbook

There is no official course textbook. However, you should find the following textbook (from COMP102) covers much of the relevant material:

- *Java Foundations: Introduction to Program Design and Data Structures*, by Lewis, DePasquale, and Chase, published by Addison Wesley, 2008.

Other useful text books include the following:

- *Program Development in Java*, Barbara Liskov
- *Object-Oriented Design & Patterns*, Cay Horstmann, Second Edition.
- *Practical Object Oriented Design*, Bhuvan Unhelkar
- *Effective Java*, Josh Bloch

There are a number of other useful books on programming in Java available in the library.

Course Materials

We will usually hand out copies of the lecture slides, though we cannot guarantee to always have them ready for the lecture. All the course handouts will be available on the course web site. Notice that the slides may be incomplete and we expect students to attend lectures and complete them.

Laboratories

Starting in the second week, you will be expected to attend one two-hour supervised laboratory a week. In the laboratories, you will be asked to solve small problems that relate to the course material and/or assignments currently underway. **The laboratories are assessed, and attendance is mandatory**. A schedule for the laboratory times will be posted on the course web site and you will need to sign up to one of the available lab streams. **The laboratory rooms for SWEN221 are** <u>CO242</u> **and** <u>CO243</u>.

Assignments

There will be 6 marked assignments of varying sizes. Each assignment will consist of one or more programming problems that explore different aspects of the material presented in the lectures. Practical work underpins this course.

problems that explore different aspects of the material presented in the lectures. Practical work underpins this course, since it is essential for a proper understanding of the material. **Therefore, you are required to submit a reasonable attempt on 5 of the 6 assignments in order to pass the course**. We expect that you will spend at least 6 hours a week working on the current assignment.

If you have access to a computer outside the labs, you may use it to work on the assignments, but you will need to acquire your own software for writing Java programs. Please note that we do not have the resources to provide assistance if you have difficulties with a computer at home -- the tutors can only answer questions about the assignments and the workstations in the laboratories. Note also that we cannot offer you any help with choosing, setting up, or fixing your own computer system, other than the general advice that we provide on the website.

Self Assessment

An online self-assessment tool will be used throughout the course. This provides a series of questions related to understanding Java programs. Every question has a specific deadline, and questions have been grouped together by the week of their due date. One mark will be awarded for correctly answering a question before the deadline. **You are required to complete 75% of questions in self-assessment tool before the end of Trimester 1.**

Assessment

Your grade will be determined based on your lab and assignment marks, the marks obtained for completed questions in the online self-assessment tool, a mid-term test and a final two-hour examination:

Item	Weight
Labs	15%
Assignments	18%
Self Assessment	7%
Mid-term test	10%
Final Examination (2 hours)	50%

The test and exam will assess your understanding of the material presented in lectures, while the lab/project work will assess your ability to apply the techniques in practice. Each assignment is worth 4%, whilst each lab is worth 1.5%.

Note: Bachelor of Engineering students should be aware that copies of their assessed work may be retained for inspection by accreditation panel.

Marking Criteria

The assignments will be marked on the following criteria:

- Correctness - does the program adhere to the given specification
- Style - does the code follow the style guide and have appropriate comments (inc. Javadoc)

Each lab will be marked on the following criteria:

- Correctness - does the program adhere to the given specification

Tests and Exams

The **mid-term test** will take the form of a short programming test, rather than a written test. The date and time of the test is **Wednesday 20th April**, from 5--8pm. The test will help you gauge your understanding of the material so far.

The timetable for final examinations will be available from the University web site and will be posted on a notice board outside the faculty office. The final examination will be two hours long. No computers, electronic calculators or similar device will be allowed in the final examination. Paper non-English to English dictionaries will be permitted. The study and examination period for Trimester T1 is between 10 June - 29 June.

Practical Work

Hand-in dates for the assignments are:
- Assignment 1 - due Monday 21st March, 2016 @ midnight
- Assignment 2 - due Monday 11th April, 2016 @ midnight
- Assignment 3 - due Monday 2nd May, 2016 @ midnight
- Assignment 4 - due Monday 16th May, 2016 @ midnight
- Assignment 5 - due Monday 30th May, 2016 @ midnight
- Assignment 6 - due Sunday 5th June, 2016 @ midnight

Each piece of work should be handed in on the dates and times specified in the Schedule. Submission should be made via the underline submission system (found on the course homepage).

Unless prior agreement with the course coordinator has been made at least 24 hours in advance, the policy on late submissions is as follows:

- **Late work will be penalised 20% per day after the deadline**. This means after 5 days zero marks will be awarded. In this case, the work should still be submitted in order to pass the mandatory requirements. However, submissions will not be accepted once any model answers have been given out.

- **Each student has three "late days"**. You may choose to use these for any assignment(s) during the course. No penalty will be applied for these late days. You do not need to apply for them - any late days you have left will be automatically applied to assignments that you submit late.

Plagiarism

We encourage you to discuss the principles of the course and assignments with other students, to help and seek help with programming details, technical issues, or problems involving the lab machines. However, any work you hand in must be your own work.

The School policy on Plagiarism (claiming other people's work as your own) is available from the course home page. Please read it. We will penalise anyone we find plagiarising, whether from students currently doing the course, or from other sources. Students who knowingly allow other students to copy their work may also be penalised. If you have had help from someone else (other than a tutor), it is always safe to state the help that you got. For example, if you had help from someone else in writing a component of your code, it is not plagiarism as long as you state (eg, as a comment in the code) who helped you in writing the method.

Mandatory Requirements

This course is focused on practical work and you must demonstrate adequate mastery of this to pass the course. Therefore, it is a mandatory requirement that you make a reasonable attempt on at least 5 of the 6 assignments. You are also required to attend at least 8 of the 10 weekly lab sessions and complete at least 75% of the questions in the online self assessment tool. Finally, you must achieve at least a 'D' grade in the final exam.

Additional Information

Workload

In order to maintain satisfactory progress in this course, you should plan to spend an average of at least 10 hours per week on this paper. A plausible and approximate breakdown for these hours would be:
- Lectures and labs: 4 hours
- Readings: 1 hour
- Assignments: 5 hours

School of Engineering and Computer Science

The School office is located on level three of the Cotton Building (Cotton 358).

Withdrawal

The last date for withdrawal from this course with entitlement to a refund of tuition fees is Friday 11 March. The last date for withdrawal without being regarded as having failed the course is Friday 13 May 2016 -- though later withdrawals may be approved by the Dean in special circumstances.

Rules & Policies

Find key dates, explanations of grades and other useful information at http://www.victoria.ac.nz/home/study.

Find out about academic progress and restricted enrolment at http://www.victoria.ac.nz/home/study/academic-progress.

The University's statutes and policies are available at http://www.victoria.ac.nz/home/about/policy, except qualification statutes, which are available via the Calendar webpage at http://www.victoria.ac.nz/home/study/calendar (See Section C).

Further information about the University's academic processes can be found on the website of the Assistant Vice-Chancellor (Academic) at http://www.victoria.ac.nz/home/about/avcacademic

All students are expected to be familiar with the following regulations and policies, which are available from the school web site:

Grievances

The School of Engineering and Computer Science strives to anticipate all problems associated with its courses, laboratories and equipment. We hope you will find that your courses meet your expectations of a quality learning experience.

If you think we have overlooked something or would like to make a suggestion feel free to talk to your course organiser or lecturer.

Course Outline as PDF