COMPSCI 230: Programming Techniques

2017 Semester 2

15 Points

Prerequisites and restrictions

Prerequisites:

COMPSCI 101 or COMPSCI 105

Course Description

Students will develop a software application of reasonable complexity through the application of established software development techniques. In doing so, students will demonstrate fundamental skills in object-oriented software development, GUI programming and applicationlevel multithreading. The programming language of this course is Java.

Staff involved in the course

Course Coordinator

- Angela Chang, room 303-414, angela@cs.auckland.ac.nz

Lecturers

- Professor Alexei Drummond, room 303S-479, email: alexei@cs.auckland.ac.nz
- Angela Chang (Course Coordinator), room 303-414, email: angela@cs.auckland.ac.nz

Tutors

- Behzad Farokhi, email: bfar713@aucklanduni.ac.nz
- Monica Bian, email: rbia002@aucklanduni.ac.nz

Timetable

Lectures

- Wednesday 4:00PM - 5:00PM, Science Chem, Room G050
- Thursday 4:00PM - 5:00PM, Science Chem, Room G050
- Friday 4:00PM - 4:00PM, Science Chem, Room G050

Please re-check all room allocations closer to the time on SSO as they often change around the start of semester.

Tutorials

- One tutorials each week as selected. Tutorials are either held in 303S-G75 or 303S.175. Tutorials begin in the 2nd week of semester (i.e. first tutorial will be 31st July).
- Mo 08:00AM – 09:00AM, 303S.175
- Mo 11:00AM – 12:00NOON, 303S.G75

- Mo 3:00PM – 4:00PM, 303S.G75
- Th 9:00AM – 10:00AM, 303S.G75
- Th 10:00AM – 11:00AM, 303S.G75
- Fr 10:00AM – 11:00AM, 303S.G75

Please re-check all room allocations closer to the time on SSO as they often change around the start of semester.

Course Outcomes

A student who successfully completes this course should be able to:

- OO Programming: describe and use the features typically offered by an object-oriented programming language, including support for classes, visibility, inheritance, interfaces, polymorphism and dynamic binding
- OO Design: explain and apply key design principles of object-oriented software development, including separation of concerns, abstraction, information hiding, programming to interfaces, coupling and cohesion, resilience to change, and reuse
- create simple OO design models
- Frameworks: describe important concepts of programming frameworks, including APIs, inversion of control and event-driven programming
- use a framework to develop a multithreaded GUI application
- Concurrent Programming: explain and apply the principles of application-level multithreading: threading, condition synchronization, mutual exclusion, and primitives associated with these

---

Lecture Schedule

Timing of the lectures, labs, assignments and test are shown below:

Wednesday	**Thursday**	**Friday**		
26 Jul **Lect 1**	*27 Jul* **Lect 2**	*28 Jul* **Lect 3**		
2 Aug **Lect 4**	*3 Aug* **Lect 5**	*4 Aug* **Lect 6**	CR01	
9 Aug **Lect 7**	*10 Aug* **Lect 8**	*11 Aug* **Lect 9**	CR02	
16 Aug **Lect 10**	*17 Aug* **Lect 11**	*18 Aug* **Lect 12**	CR03	
23 Aug **Lect 13**	*24 Aug* **Lect 14**	*25 Aug* **Lect 15**	CR04	
30 Aug **Lect 16**	*31 Aug* **Lect 17**	*1 Sep* **Term-Test**	CR05	Term Test (20%)
6 Sep **Mid-semester break**	*7 Sep* **Mid-semester break**	*8 Sep* **Mid-semester break**		
13 Sep **Mid-semester break**	*14 Sep* **Mid-semester break**	*15 Sep* **Mid-semester break**		
20 Sep **Lect 18**	*21 Sep* **Lect 19**	*22 Sep* **Lect 20**	CR06	A1 (5%) M018Sept
27 Sep **Lect 21**	*28 Sep* **Lect 22**	*29 Sep* **Lect 23**	CR07	
4 Oct **Lect 24**	*5 Oct* **Lect 25**	*6 Oct* **Lect 26**	CR08	
11 Oct **Lect 27**	*12 Oct* **Lect 28**	*13 Oct* **Lect 29**	CR09	
18 Oct **Lect 30**	*19 Oct* **Lect 31**	*20 Oct* **Lect 32**	CR10	
25 Oct **Lect 33**	*26 Oct* **Lect 34**	*27 Oct* **Lect 35**		A2(5%)

Assessment

Requirements for passing

This course is a practical course. That means you have to pass the practical component (CodeRunner exercises and assignments) as well as the written (test and exam) components.

- CodeRunner Exercises (worth 10%)
- Assignments (worth 10%)
- Test worth 20%
- Final exam worth 60%

Notes:

- To pass the practical component, you will need to have at least 10% out of 20.
- To pass the written component, you will need to have at least 40% out of the combined test and exam total of 80%.
- You also need an overall mark of at least 50% out the full course total of 100%.

CodeRunner Exercises

The CodeRunner exercises are worth 10% of your final mark. The exercises start in the second week of the semester. The exercises are designed to give you practical experience with the concepts which you have learnt in lectures. Each week you should make sure you have read through the lecture slides and have done any other preparation required before you attend the laboratory.

The exercises are submitted using the CodeRunner2 website. Submissions are graded by running a series of test cases of the code in a sandbox and comparing the output of your program with the expected output. This automated testing allows you to obtain immediate feedback on your progress throughout the course. Some submissions will also be graded manually to provide further feedback. You will be able to access your exercises by logging into:

https://www.coderunner2.auckland.ac.nz/moodle/

Assignments

There are two assignments (worth a total of 10%) for which you are required to write one or more programs. All assignments are to be submitted to the assignment dropbox:

https://adb.auckland.ac.nz/Home/

Test

The Test is worth 20% of your final mark, and will be held between 4:00 pm - 5:00 pm on Friday 1st September. Please ensure you are available to sit the Test - if you are enrolled in another course that has a test scheduled for the same time (or if you are unable to attend this test time for another valid reason), then please contact the course coordinator, Angela Chang, as soon as possible.

Exam

The final exam is worth 60% of your final mark. Please check Student Services Online for the exam time and date. The exam is closed book, and calculators are not permitted. Provisional exam results can be obtained from Student Services Online.

Checking your marks on Canvas

You can check your marks by logging onto Canvas

https://canvas.auckland.ac.nz

If there are any problems with your Lab, Assignment or Test marks, please contact Angela Chang.

Assistance

There are a number of places where you can seek assistance with your learning. If you have an administrative problem (e.g. you have been ill, you have a timetable clash with your lab or test, your marks have been incorrectly recorded, etc.), or any other sort of problem that you need help with, please see the course coordinator.

Students are asked to discuss privately any impairment related requirements face-to-face and/or in written form with the course coordinator. If you need extra help with understanding the course material, or preparing for the test or exam, you are very welcome to visit the teaching staff team or any lab tutor at a time when they are available.

There are many other resources available within the University, e.g. the Student Learning Centre, the library, DELNA (to identify where you may need help with your academic English) and ELE (English Language Enrichment - a set of resources to help you improve your English).

Office Hours

All staff have office hours when they are available to students. You are encouraged to come and discuss any matters arising from the course during those hours. Staff are also frequently available at other times.

Lecture Recordings

All lectures are recorded. They may be a delay of 1-2 days before the lecture recordings are distributed through Canvas. You can find the lecture recordings on the Lecture Recordings page (COMPSCI 230 > Pages > Lecture Recordings). Note that although the lectures are recorded, some learning activities conducted in class do not translate well to the recordings. To maximise your learning opportunities, you are encouraged to attend the class in person.

Piazza

The discussion forums (Piazza) within Canvas are regularly monitored by teaching staff. Please make use of the forums to ask any questions that you think might be of interest to other students. If your question is of a personal nature, or relates to a unique situation that will be of little interest to others, then please contact the teaching staff directly.

Textbook

The textbook is Deitel & Deitel, Java How to Program (late objects), 10th Edition, Prentice Hall (2014). Note that the textbook is provided as supplemental material to enhance and extend your understanding. You are encouraged to read the relevant chapters in parallel to the in-class activities. The textbook is not compulsory but comes strongly recommended. Note: Most topics are also covered in the 9th edition.

Class Website

Although we will be using Canvas during semester 1 for announcements and recordings, the COMPSCI 230 website contains some basic information (lecture notes and assessment resources) about the course:

https://www.cs.auckland.ac.nz/courses/compsci230s2c/

Help with Canvas

For help with Canvas see:

https://www.auckland.ac.nz/en/about/learning-and-teaching/CanvasHomepage/canvas-help---support.html.

Handling illness or absence

If you must leave for family emergencies etc., PLEASE talk to the lecturer, or somehow get a message to the department. Very few problems are so urgent that we cannot be told quite quickly.

For problems affecting assignments or tests, see the lecturer, as soon as reasonably possible.

For illness during exams (or other problems that affect exam performance) students MUST contact the University within **one week** of the last affected examination, to apply for an aegrotat pass (for illness) or compassionate pass (other problems). **The one week limit is strictly enforced.**

Refer to the University information about Aegrotat and Compassionate Considerations:

https://www.auckland.ac.nz/en/students/academic-information/exams-and-final-results/during-exams/aegrotat-and-compassionate-consideration.html

Academic Integrity

The University of Auckland will not tolerate cheating, or assisting others to cheat, and views cheating in coursework as a serious academic offence. The work that a student submits for grading must be the student's own work, reflecting his or her learning. Where work from other sources is used, it must be properly acknowledged and referenced. This requirement also applies to sources on the world-wide web. A student's assessed work may be reviewed against electronic source material using computerised detection mechanisms. Upon reasonable request, students may be required to provide an electronic version of their work for computerised review.

Please refer to http://www.auckland.ac.nz/uoa/home/about/teaching-learning/honesty.