Science

COMPSCI 130 : Introduction to Software Fundamentals (15 POINTS)

Course Prescription

Fundamental programming techniques and processes, such as conditionals, iteration, recursion, functions, testing and debugging. Efficient ways to organise and manipulate data, including sorting and searching algorithms. Writing software that uses and implements common abstract data types such as lists, stacks, queues, dictionaries and trees.

Course Overview

This is the entry course to Computer Science for students with prior programming experience. It focuses on the quality of processes used when developing software, and the quality of the software product produced using those processes. The course provides an introduction to fundamental software development techniques and processes, such as reading, writing, and documenting programming code, decomposing problems, testing, debugging, using recursion and handling unexpected errors. It also addresses efficient ways to organize and manipulate data, including sorting and searching algorithms, and writing software that uses and implements common abstract data types such as lists, stacks, queues, dictionaries and trees. The course will be taught using the Python programming language.

Course Requirements

Prerequisite: COMPSCI 101, or Achievement Standard NCEA Level 3: Digital Technologies and Programming: 91637 Develop a complex computer program for a specified task Restriction: COMPSCI 105, 107

Capabilities Developed in this Course

Capability 1:	Disciplinary Knowledge and Practice
Capability 2:	Critical Thinking
Capability 3:	Solution Seeking
Capability 4:	Communication and Engagement

Graduate Profile: Bachelor of Science

Learning Outcomes

By the end of this course, students will be able to:

1. Decompose a problem into several smaller tasks, design and implement a function for each task, and compose these functions. (Capability 1, 2 and 3)
2. Export and import data structures (via file or console I/O) using standard text-based data formats. (Capability 1 and 3)
3. Use common programming statements to implement iterative and recursive algorithms. (Capability 1, 2 and 3)
4. Use simple testing and debugging strategies to correct faulty programs. (Capability 1, 2 and 3)
5. Demonstrate how typical data structures are modelled in memory. (Capability 1)
6. Provide a useful level of documentation for all programs developed. (Capability 1 and 4)
7. Work together with peers to collaboratively develop, and review, programs. (Capability 1, 2, 3 and 4)
8. Write programs that use standard abstract data types (lists, stacks, queues, priority queues, dictionaries). (Capability 1, 2 and 3)
9. Implement standard abstract data types using standard data structures such as arrays, linked lists, hash tables and trees. (Capability 1, 2 and 3)

Assessments

Assessment Type	**Percentage**	**Classification**
Assignment	6%	Individual Coursework
Test	25%	Individual Test
Labs	16%	Individual Coursework
Timed Quizzes	4%	Individual Coursework
Exam	45%	Individual Examination
Group Activities / Attendance	4%	Individual Coursework
6 types	100%	

Assessment Type	**Learning Outcome Addressed**								
	1	2	3	4	5	6	7	8	9
Assignment	✓	✓	✓	✓		✓		✓	✓
Test	✓	✓	✓	✓				✓	✓
Labs	✓	✓	✓	✓				✓	✓
Timed Quizzes	✓	✓	✓	✓				✓	✓
Exam	✓	✓	✓	✓				✓	✓
Group Activities / Attendance	✓				✓		✓		

To pass the course, as well as obtaining at least 50% overall, a student must also pass the invigilated component (Test + Exam).

Tuākana

For more information and to find contact details for the Computer Science Tuākana coordinator, please see https://www.auckland.ac.nz/en/science/study-with-us/maori-and-pacific-at-the-faculty/tuakana-programme.html

The School of Computer Science Tuākana programme provides support for this course. See: https://canvas.auckland.ac.nz/courses/34081

Key Topics

Topic 1: Python revision
Topic 2: Software maintenance, modularity and testing
Topic 3: Exceptions
Topic 4: Complexity of programs; sorting and searching
Topic 5: Abstraction; Classes, and Abstract Data Types
Topic 6: Recursion
Topic 7: Stacks and queues
Topic 8: Linked lists
Topic 9: Hashing
Topic 10: Trees
Topic 11: Binary Search Trees
Topic 12: Priority Queues and Heaps

Special Requirements

There will be one (mid-semester) test which will be held in the evening. The test will be computer-based and will assess programming ability. The exam will also be computer-based and will assess the understanding of concepts related to data structures as well as programming ability.

To pass the course, as well as obtaining at least 50% overall, a student must also pass the invigilated component (Test + Exam).

Workload Expectations

This course is a standard 15 point course and students are expected to spend 20 hours per week involved in each 15 point Summer School course that they are enrolled in. Summer School courses run over 6 weeks rather than 12 weeks so the weekly workload is twice the number of hours expected for other semesters.

For this course, each week, you can expect 2 hours of lectures, 4 two-hour labs, 4 hours of reading and thinking about the content and 6 hours of work on programming practice and/or test and exam preparation.

Delivery Mode

Campus Experience

Attendance is required at scheduled activities to receive credit for the components of the course.

Lectures will be available as recordings. Other learning activities will not be available as recordings.

The course will not include live online events.

Attendance on campus is required for both the mid-semester test and the exam.

The activities for the course are scheduled as a standard weekly timetable.

Learning Resources

This course assumes you know how to program, but makes no assumption about the programming language that you know. The course is taught in Python, and includes a short overview of Python syntax at the beginning of the course. To revise your Python knowledge, the following online text is recommended: https://runestone.academy/runestone/static/thinkcspy/index.html

The text for COMPSCI 130 can be located at: https://onlinetextbook.auckland.ac.nz/

Student Feedback

During the course Class Representatives in each class can take feedback to the staff responsible for the course and staff-student consultative committees.

At the end of the course students will be invited to give feedback on the course and teaching through a tool called SET or Qualtrics. The lecturers and course co-ordinators will consider all feedback.

Your feedback helps to improve the course and its delivery for all students.

Other Information

Please contact the course coordinator if you have any queries or concerns.

Digital Resources

Course materials are made available in a learning and collaboration tool called Canvas which also includes reading lists and lecture recordings (where available).

Please remember that the recording of any class on a personal device requires the permission of the instructor.

The text for COMPSCI 130 can be located at: https://onlinetextbook.auckland.ac.nz/

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 their learning. Where work from other sources is used, it must be properly acknowledged and referenced. This requirement also applies to sources on the internet. A student's assessed work may be reviewed against online source material using computerised detection mechanisms.

Copyright

The content and delivery of content in this course are protected by copyright. Material belonging to others may have been used in this course and copied by and solely for the educational purposes of the University under license.

You may copy the course content for the purposes of private study or research, but you may not upload onto any third party site, make a further copy or sell, alter or further reproduce or distribute any part of the course content to another person.

Inclusive Learning

All students are asked to discuss any impairment related requirements privately, face to face and/or in written form with the course coordinator, lecturer or tutor.

Student Disability Services also provides support for students with a wide range of impairments, both visible and invisible, to succeed and excel at the University. For more information and contact details, please visit the Student Disability Services' website http://disability.auckland.ac.nz

Special Circumstances

If your ability to complete assessed coursework is affected by illness or other personal circumstances outside of your control, contact a member of teaching staff as soon as possible before the assessment is due.

If your personal circumstances significantly affect your performance, or preparation, for an exam or eligible written test, refer to the University's aegrotat or compassionate consideration page https://www.auckland.ac.nz/en/students/academic-information/exams-and-final-results/during-exams/aegrotat-and-compassionate-consideration.html.

This should be done as soon as possible and no later than seven days after the affected test or exam date.

Learning Continuity

In the event of an unexpected disruption we undertake to maintain the continuity and standard of teaching and learning in all your courses throughout the year. If there are unexpected disruptions the University has contingency plans to ensure that access to your course continues and your assessment is fair, and not compromised. Some adjustments may need to be made in emergencies. You will be kept fully informed by your course co-ordinator, and if disruption occurs you should refer to the University Website for information about how to proceed.

Level 1: Delivered normally as specified in the delivery mode

Level 2: You will not be required to attend in person. All teaching and assessment will have a remote option. The following activities will also have an on campus / in person option: lectures, labs, office hours

Level 3 / 4: All teaching activities and assessments are delivered remotely

Student Charter and Responsibilities

The Student Charter assumes and acknowledges that students are active participants in the learning process and that they have responsibilities to the institution and the international community of scholars. The University expects that students will act at all times in a way that demonstrates respect for the rights of other students and staff so that the learning environment is both safe and productive. For further information visit Student Charter https://www.auckland.ac.nz/en/students/forms-policies-and-guidelines/student-policies-and-guidelines/student-charter.html.

Disclaimer

Elements of this outline may be subject to change. The latest information about the course will be available for enrolled students in Canvas.

In this course you may be asked to submit your coursework assessments digitally. The University reserves the right to conduct scheduled tests and examinations for this course online or through the use of computers or other electronic devices. Where tests or examinations are conducted online remote invigilation arrangements may be used. The final decision on the completion mode for a test or examination, and remote invigilation arrangements where applicable, will be advised to students at least 10 days prior to the scheduled date of the assessment, or in the case of an examination when the examination timetable is published.