UCT Computer Science

Honours Student Handbook

2024

UNIVERSITY OF CAPE TOWN

Contact Details

Computer Science Building
18 University Avenue
Upper Campus
Rondebosch
Cape Town
South Africa

University of Cape Town
Private Bag X3
Rondebosch
7701
South Africa

Telephone: +27 (0)21 650 2663
Fax: + 27 (0)21 650 3551
Email: dept@cs.uct.ac.za

Revision January 2024

This booklet contains details of the structure of the Honours (fourth year) programme offered by the Computer Science Department at UCT, as well as some general information about the department itself. It should be read in conjunction with the postgraduate student handbook, which contains details of the department's Masters and Doctoral Degrees (including Masters-level modules that honours students may take).

Our Mission

The mission of the Department of Computer Science is to develop and impart knowledge and skills in the field of Computer Science.

Our Vision

The Department of Computer Science strives to be of the first rank, maintaining excellence in both research and teaching and producing high-quality graduates skilled in problem solving, in order to play an influential role in the development of Information Technology, both within the continent of Africa and internationally.

Table of Contents

1. General Information

The Department of Computer Science of the University of Cape Town has one of the best computer science departments in the country for pursuing postgraduate study. We have a strong core team of established researchers with international recognition in a wide variety of disciplines, with doctorates and experience from some of the world's leading computer science institutes.

1.1. Honours in Computer Science

Our Honours degrees are single-year programmes designed to provide students with the professional basis for a career path in the computer industry, and/or to enable them to embark upon a research programme at Masters level.

The Honours year is an opportunity to undertake advanced courses in computer science, as well as to complete a major research and development project. Honours is an enriching and challenging year which requires a firm commitment to full-time, hard work. No part-time options are available.

1.2. Programme Convenor

The 2024 Honours programme ccordinator is:

- First semester: ***Assoc. Prof. Maria Keet***, Room 314.10, Computer Science Building (email: mkeet@cs.uct.ac.za).

The Honours Teaching Assistant (TA) is Jane Imrie (email: IMRJAN001@myuct.ac.za). The role of the TA is to assist with course administration.

1.3. Role of the Student

Postgraduate students are expected to be interested in deepening their knowledge and experience, particularly in Computer Science, but also in related fields.

Postgraduate students are expected to engage with the process of research and development, to be critical thinkers and to work productively, both independently and as part of a team. Students are expected to go beyond the basic requirements of a course or a project, reading widely in the relevant academic literature to contextualize and frame their work. We expect assignments and research projects to be solved independently and creatively, showing due appreciation for academic concepts and principles. In addition, students are expected to communicate ideas and results clearly in both written deliverables and presentations. In particular, postgraduate students are expected to:

- ensure that they **register for** and **complete successfully** sufficient courses to complete the coursework requirement for their degree;
- behave as a professional, arriving **punctually for all classes**, meetings and seminars;
- **attend all classes** and participate actively in class;
- manage their time effectively, working hard and consistently and **submitting all assignments by the posted deadlines**;
- work largely unsupervised and independently;

- engage in the process of peer review - evaluating their own and others' work and **responding to criticism thoughtfully and dispassionately,** using critique to improve their work;
- to engage actively with research talks and symposia hosted by the department;
- and to **raise any issues timeously** and politely with the course coordinator and/or the teaching assistant.

1.4. Role of Class Representatives

Honours class representatives will be elected at the beginning of the year. There will be a separate representative for 4th year Business Science students.

Class representatives are expected to arrange regular meetings with the course coordinator, in order to resolve in consultation issues that arise during the year. In addition, occasional meetings with the Head of Department will be scheduled. It is the responsibility of the class representative to act as a liaison between the class and the department: issues must be raised promptly, giving a balanced view of the class opinion. Therefore, class representatives are expected to regularly poll the class and give formal class feedback after meetings. In addition, class representatives should oversee the responsible use of the honours lab. Individual course may have additional duties for the class representatives.

1.5. Role of the Department

The UCT Computer Science department is a team of qualified, established researchers, comprising some of the best Computer Scientists in the country and the continent. In general, our role is to:

- produce skilled, high quality graduates who are familiar with the principles, theory and practice of Computer Science;
- carry out innovative research;
- provide services to Industry, through technology transfer and applied research;
- take an active part in the academic and governance affairs of the University;
- provide opportunities and support for students from disadvantaged backgrounds;
- and promote, support and advise schools in the teaching of topics related to Computer Science.

For our postgraduate programmes, our role is more specifically to produce individuals who are educated, articulate, and able to perform research and exercise critical judgement in the field of Computer Science. Our core function, therefore, is not that of providing vocational training, but to impart the fundamental skills that are needed for decision making or creative thinking. We do not aim to train people how to use computers and become programmers to meet the immediate demands of the marketplace (although this is taught as a matter of course) - we want our students to remain useful scientists a decade from now.

1.6. Fees

Fees vary from year to year: consult the latest UCT *Fees Handbook* for accurate figures.[1] The Honours fees are course-based: additional courses over and above the basic requirements will incur an additional fee.

[1] http://www.students.uct.ac.za/students/fees-funding/fees/handbook

International students pay higher rates, which vary according to country of origin, as follows. Citizens and permanent residents of SADC countries pay the same fees as South African Residents. (The SADC countries are Angola, Botswana, Democratic Republic of Congo, Lesotho, Madagascar, Malawi, Mauritius, Mozambique, Namibia, Seychelles, South Africa, Swaziland, Tanzania, Zambia, and Zimbabwe.) Citizens from other countries pay a fee comprising a Course Based Fee, an International Term Fee and an International Administrative Service Fee. The International Term Fee for citizens of non-SADC African countries is typically lower than for citizens from non-African countries.

International students must pay fees prior to registration.

1.7. Financial Assistance

Financial assistance is available for prospective postgraduate students. The *Financial Assistance for Postgraduate Study and Postdoctoral Research Handbook*[2] lists opportunities for both SA and international students – also look at the University's Postgraduate Degree Funding web page.[3] Pay particular attention to the deadlines for applications for financial assistance – deadlines are often as early as June of the preceding year.

The South African National Research Foundation (NRF) provides a *limited* number of bursaries to South African citizens. The closing date for NRF bursaries may be as early as June; the notification date is February of the next year.

The University offers merit awards for students who achieve at least 65% in their undergraduate majors.[4]

Bursaries for Computer Science Honours at UCT can often be obtained from companies who are keen to hire students.

You are advised to apply for all bursaries as early as possible.

In addition to the above, some members of the department have funds for research purposes, which may be available to students involved in specific projects.

1.8. Facilities

*All Honours students are **required to have their own laptops.***

Honours students are accommodated in the **Honours Lab** - a shared-workspace laboratory with 24-hour access, a small kitchen and coffee area, lockers and some workstations.

The department has a server infrastructure that delivers core services to students and staff over high speed wired and wireless networks. Different research projects require different specialised equipment and the research supervisors will address these needs.

[2] http://www.students.uct.ac.za/students/study/handbooks/current

[3] http://www.uct.ac.za/apply/funding/postgraduate/applications

[4] http://www.students.uct.ac.za/students/fees-funding/postgraduate-degree-funding/bursaries-scholarships/merit-need-awards

2. Honours Degree Structure

The Department offers one Honours degree since 2024: BSc (Hons) in Computer Science. Both students with a major in Computer Science from UCT and external students do the same degree. Students in their final year of the Bachelor of Business Science with a major in Computer Science register for the same course codes as Computer Science honours students (although they take fewer elective modules).

2.1. Accreditation

In Computer Science, four years of study are required for a professional qualification: four years are required to cover the ACM/IEEE Computer Science Curriculum[5] in sufficient breadth to practice with confidence in the field of Computer Science and to fulfil the academic requirements of the British Computer Society (BCS) for a professional Computer Science qualification. The UCT BSc (Hons) in Computer Science was accredited as meeting the academic requirements for Chartered IT Professional (CITP) certification[6] and partially meeting the requirements for Chartered Scientist (CSci) registration.

2.2. Admission

Admission to CS Honours is **competitive and not guaranteed**. The number of places in the UCT CS Honours programme is limited and students are selected **on merit** from the list of applicants each year. Admission is primarily based on marks obtained in the second- and third-year undergraduate Computer Science major courses, but the marks obtained in other courses may also be taken into consideration. The minimum entrance requirement is a mark of 60% in each of the final year courses (CSC3002F and CSC3003S), but a third year average of at least 65% is usually required. Applicants are given the opportunity to submit motivations as part of the application; these are also considered as part of the admissions process.

2.3. Employment and Further Study after Honours

Our Honours students from previous years found employment in international companies overseas (e.g. Facebook, Nvidia, Microsoft, Amazon); in local branches of international companies (e.g. Amazon, Oracle, MWR infosecurity); in large South African corporations (e.g. BSG, Investec, Old Mutual and Standard Bank); in smaller local software companies or startups e.g. (Smyte, Nomanini, Praekelt Foundation, Thought Express) and in local animation and visual effects studios (Sea Monster and Black Ginger). Some graduates create their own startup companies.

Students who successfully complete Honours in Computer Science at UCT are eligible to apply to proceed to an M.Sc. in Computer Science. Many of our best students have received scholarships to study further at top international universities.

[5] http://www.acm.org/education/curricula-recommendations

[6] http://www.bcs.org/server.php?show=nav.7065

2.4. Orientation, Registration, and Start of the Year

The academic year for Honours students begins two weeks before the first undergraduate lectures - consult the UCT calendar for start dates.[7]

All students must be present on the first day for the compulsory orientation session. Registration must be completed on the first day of honours.

Registration for the Computer Science Honours students is handled by the Science Faculty.

Registration for 4th Business Science students is handled by the Commerce Faculty – for questions specifically related to Business Science registration contact the responsible student advisor (currently Mr. Aslam Safla).

2.5. Registration of International Students

In order to be able to register at UCT, international students need:

- A study visa. (Study visas must be obtained from the South African Embassy, High Commission or Consulate in your home country. They cannot be obtained from within South Africa.)
- Proof of proficiency in English.
- Health insurance.

The International Academic Programmes Office (IAPO) can provide help and information about these and other requirements.[8]

International students (including those from SADC countries) will apply for pre-registration clearance from the International Academic Programmes Office. This clearance includes fee clearance, presentation of a study permit and health insurance information. International students cannot register for the degree if they have not been cleared by IAPO. For any questions about the pre-registration process, please contact IAPO at intiapo@uct.ac.za or at +27 21 650 2822 / 3740.

2.6. Structure of the Honours Year

Since the courses given in the initial weeks are compulsory, it is not possible to excuse any student from attendance during this period. The remaining modules are arranged across four blocks, where blocks roughly correspond to UCT terms. Modules are scheduled in **blocks one, two** and **four**. In the third block, students devote themselves exclusively to their projects. The only exceptions to this structure are in the case of taking external modules.

All modules given in a block will be completed by the end of that block and ***no extensions will be granted to complete work after this period***. The projects will be allocated during the first block and various project-related milestones have to be met from then on until the end of the year. The final report for the project is due at the end of the third block.

A total of 160 credits must be obtained during the course of the academic year. All compulsory modules must be completed by every student (40 credits). Students may select any remaining modules as electives, with a minimum of 60 credits, up to a maximum of 80 credits. The project comprises the remaining 60 credits.

[7] http://www.uct.ac.za/main/calendar/academic-calendar

[8] http://www.iapo.uct.ac.za/

2.7. Workload

Honours is an intensive, full-time course and may not be taken together with other courses or while you are employed. Permission to deviate from this will only be given in exceptional circumstances by the Programme Coordinator. Your weekly workload will be between 40 and 48 hours per week.

For each lecture hour you should allocate at least two hours of extra work to review material and for the associated tutorials and practicals.[9]

You should allocate at least 5 hours per week to supervisor meetings, planning your project, reading background material etc., during the project time.

2.8. Passing Honours

In order to obtain the Honours in Computer Science degree, students must fulfil ALL these requirements:

- at least **50% for their project mark**.
- A pass in all the compulsory modules (40 credits).
- A pass in a sufficient number elective modules (at least 60 credits) to meet the requirements of at least 100 total coursework credits for the degree.

Students who fail a small number of courses (compulsory or electives) are usually allowed to come back for a second year to complete (only) the outstanding courses for completing the honours degree, subject to approval by the Faculty Examinations Committee. Students who fail the project and/or a majority of courses are not usually allowed to come back. Students who are refused permission to continue may appeal to the Readmission Appeals Committee.

3. Honours Coursework

3.1. Modules

The Departmental coursework modules are listed in the Compulsory and Elective Modules sections for the relevant degrees. We offer sufficient modules at all levels for you to fulfil your coursework requirement. However, you may, subject to the restrictions mentioned below, take selected modules from other departments. Note, however, that any module you register for outside the department must be of an appropriate level, have relevance to Computer Science and **have the prior approval of the respective programme coordinator**. You may only take modules which have a course code.

A 12 credit module typically corresponds to 15 lectures and substantial practical work.

Note well: In order to gain credit for a module, students **have to pass the module.**

3.2. Lecture Periods

Lectures are scheduled in lecture periods 1–8 (8am/9am/10am/11am/12pm/ 2pm/3pm/4pm). The timetable is drawn up in consultation with lecturers to best accommodate their lecture

[9] These two hours could be allocated as one hour of theory review and one hour of practical work for a standard module or some other appropriate combination for a more practical or theoretical module.

commitments as well as the availability of venues. We attempt to avoid lecture clashes, but these will occur in exceptional cases. If the lectures for two different modules coincide, you may only register for one of the modules. Modules run only once in the year: they are not repeated.

3.3. Module Registration

Coursework modules are selected on registration. Changes to the module selection must be done via a **change of curriculum form** (ACA09), submitted on PeopleSoft as a service request to the Science Faculty Office. The request will then be approved by the programme convenor.

Honours students may not register for more than 120 course credits all together (i.e., 180 credits including the project). Module selection changes must be submitted **by the end of the first week of lectures in each teaching block** of the concerned courses.

3.4. Examinations

Modules are usually examined after the completion of the block in which the module was given. Courses offered by other departments are usually examined in the University examination periods (May/June and October/November), as scheduled by the department in question. A 12 credit module in Computer Science Honours typically either has an invigilated two-hour examination, or a 12-hour open book take-home examination. The examination timetable will be made available before each examination period. Results will be released after the exam papers were moderated by the external examiner.

We do not have any supplementary exams for postgraduate courses in the CS department.

3.5. Course Work Requirements

To fulfil the Honours coursework requirement, the following rules apply:

- You must complete the compulsory Research and Innovation module (16 credits) successfully. This module has two components: **New Venture Planning** and **Research Methods**.
- You must complete two compulsory modules: **Functional Programming** (12 credits) and **Compilers 1** (12 credits).
- Computer Science honours students must obtain credit for at least **60 credits of elective material** (you may take at most 80). Business Science students must obtain credit for at least 24 credits of elective courses (and take at most 48 credits).
- **At most 40 credits from Mathematics** courses or at **most 20 credits from Statistics or other external departments** may be taken. Business Science students may take at most 20 credits from Mathematics or Statistics courses.

4. General Policies

4.1. School of IT Seminars

The School of IT (which the Computer Science is part of) has regular seminars where members of the department, visitors and invited speakers present their research. School of IT seminars are normally held 1-2pm on a Thursday. Seminars are announced in advance by email. **Please note that attendance of at least 75% of the seminars is mandatory for**

Honours students. Failure to do this will impact on the mark for Research Methods with a 10% penalty. A register of attendance will be kept.

4.2. Missing assignments, tests or examinations, medical notes and short leave

All students are expected to do all assigned work and submit assignments by the posted deadlines. The standard penalty for late submission is 10% of the total mark per day, or part thereof.

If a student falls ill, they must submit a **medical note** to the **course convenor** as soon as possible and discuss making up for missed work with the course convenor. The authenticity of medical notes may be verified.

Students who miss a test, exam or any other formal assessment will only be allowed admission to an alternate assessment if the circumstances warrant this as per the rules for Deferred Examinations in the UCT General Rules and Policies Handbook.

In other/exceptional circumstances, where a student would like to be away from their studies for a short and defined period of time, the **Short Leave Application form** must be filled out and submitted to the course convenor for recommendation and the HoD for approval.

4.3. Plagiarism

The University of Cape Town has well defined policies on the copying of academic submissions and plagiarism, which are contained in the general Rules for Students and set out in full on the web. The Department of Computer Science has set out the a guide and interpretation of these rules and policies as they apply to courses involving computer programming, and the use of the computer. The rules defined here are in line with the best procedures in other institutions and will be available online or on request from the course convenors. **All students are expected to familiarise themselves with the rules on plagiarism**. In addition, there is an addendum on the use of “Generative AI” systems, such as ChatGPT.

4.4. Appeals Procedure

If a student feels that their marks are incorrect for any piece of marked work (excluding exams), they must first approach the tutor or TA responsible for that test or assignment. Then, if need be, they may appeal to the course convenor. Finally, they may appeal to the HoD.

4.5. Examination Scripts

Students may request to view their exam scripts, but only after the course results have been approved by the Faculty Examinations Committee. Requests should be made through the departmental administrator; a fee is payable. Exam results may not be appealed: The only case where the marks will be updated after external moderation is if is discovered that some questions were not marked or marks were added up incorrectly.

5. The Honours Project

Students are required to complete a major project under the supervision of a member of staff, possibly in conjunction with an outside supervisor. The project comprises a substantial research or software development task.

Projects involve multiple students in a team, but they are structured so that there are *readily identifiable components for each person to complete*. Each contribution to the overall project will be written up separately and so must constitute a piece of work that can be *independently assessed*.

5.1. CSC4002W Computer Science Honours Project (60 NQF credits)

Convener: Assoc. Prof. Maria Keet

This is the course code for the honours project.

5.2. Timing

The project topics are presented to the Honours class at the start of the academic year and allocated early in the first block. Students are expected to start working on the project soon after allocation and meet their project supervisor weekly. A block of about eight weeks has been set aside in the third term, to allow for dedicated work on the project.

A great deal of importance is placed on making regular progress throughout the project period. A detailed list of milestones contains deadlines and specifications of what has to be handed in or presented. The list will be available on Vula when the projects are assigned.

5.3. Project Choice and Allocation

A list of projects is released in the first block and projects are allocated to students by the beginning of the second block. Students form groups of 2 or 3 students. After a period where they should contact academic staff to learn more about the proposed projects, each group submits a ranked list of project preferences. Each supervisor may reserve a project for one group (only) on mutual agreement between the supervisor and the group. All other groups will be allocated projects based as far as possible on their submitted preferences, while also balancing the distribution of projects between staff.

Normally academic staff propose the projects, but students may submit their own project idea, provided that the project has significant Computer Science content, it can be run as a team, and that a staff member agrees to oversee the project. Please note that students are not guaranteed to be assigned their proposed project.

5.4. Deliverables

In the first phrase of the project students will be required to write a literature review (individually) as well as a formal project proposal (as a project group). Both of these will be assessed as part of the Research Methods course. The project proposal will be vetted by staff at a formal project presentation.

The final project paper must be handed in on Vula no later than the specified due date. A maximum of three days beyond the official hand in time is permitted, but will incur a penalty of 10% of the allocated marks per day for such a delay. Extensions are only granted if the delays in completing the project are beyond the reasonable control of the student(s) concerned.

The project paper should constitute a comprehensive description of your project. A document detailing what such a paper should contain will be handed out when the projects are allocated. *No paper may be submitted without the prior approval of the project supervisor*. The supervisor may require alterations and so the final draft must be available in good time for it to be read by your supervisor and for you to then revise it. The project paper comprises 80% of the final project mark. The remaining 20% is made up of additional marked deliverables. These currently include: a prototype demo early in the third block, a final demo after submission of the final project paper, a poster and a project webpage. The code of all software developed as part of the project has to be submitted soon after the final project paper is due.

5.5. Award for Best Project

In 2008, the department together with Business Systems Group (BSG) instituted an award for the best project. This goes to the team that has achieved the best overall result in their project in a particular year. Winners to date are:

2008 *WiiRobot: Teleoperation of Rescue Robots in Urban Search and Rescue Tasks* by Jason Brownbridge and Graeme Smith.

2009 *Dynamic Content in Procedural Generation* by Richard Baxter and Zacharia Crumley.

2010 *Gesture-based Games with the iPad* by Pierre Benz, Nina Schiff and Daniel Wood.

2011 *A Sketch-based Interface for Modelling Trees and Plants* by Matthew Black, Mark Dahoner and Neil Goldberg.

2012 *Smart Security Systems in an Internet of Things Environment* by Alexander Comer-Crook, Simon Groll, Shaun Michaels

2013 *StockOut Web Services* by James Lewis, Sven Siedentopf

2014 *Detection and Visualization of Radio Frequency Interference* by Philippa Hillebrand, Gerard Nothnagel

2015 *Evaluating Three-Dimensional Modelling Interfaces* by Siobahn O'Donovan and Steven Rybicki

2016 *Natural presenter tracking in 4K video* by Charles Fitzhenry, Maximillian Hahn and Mohamed Khatieb.

2017 *Hand Gesture Recognition* by Anna Borysova, Shaheel Kooverjee and Erin Versveld.

2018 *Scaled Passive Haptic Props for Virtual Reality* by Jocelyn van Heerde and Mahnoor Ahmed.

2019 *Defeasible Disjunctive Datalog* by Matthew Morris and Tala Ross

2020 *EVOBAB: Evolving Complexity in Robot Bodies and Brains* by Brooke Steward and Christina Spanallis.

2021 *Propositional Defeasible Explanation* by Lloyd Everatt and Emily Morris.

2022 *Model-Based Defeasible Reasoning* by Jaron Cohen and Carl Combrinck.

2023 *Applying diffusion models to terrain generation* by Oliver Borg, Carl Brann, and Timothy Hitge.

6. Compulsory Honours Modules

The modules listed below are all compulsory for all Computer Science Honours students. Module descriptions be found in the current Science Faculty handbook.

Honours is scheduled in four blocks (numbered 1 to 4, corresponding roughly to UCT terms), with the modules lectured in Blocks 1, 2 and 4. In addition, the first 2 weeks of Honours is pre-Block 1, with an external lecture teaching the New Venture Planning portion of CSC4019Z. The module timetable varies from year to year, **is subject to change**, and will be communicated by the Honours coordinator on the dedicated Honours Google calendar.

6.1. **CSC4019Z - Research and Innovation (*16 NQF credits)***

Convener: Assoc. Prof. Maria Keet

When: Pre-block 1; Block 1 and Block 2

This course introduces students to knowledge essential for computer professionals and researchers.

The first component teaches entrepreneurship as New Venture Planning: a critical element of economic development. This module introduces students to the ideas, theories and concepts associated with entrepreneurial ventures, with a focus on the elements needed to develop a viable business plan.

The second component develops communication and writing skills and introduces basic research methodology. Aspects of scientific writing and research methods for statistical analysis and evaluation of data are also covered.

6.2. **CSC4020Z - Functional Programming (*12 credits)***

Convener: Assoc. Prof. Geoff Nitschke

When: *Block 1*

This course will expose students to the alternative functional programming paradigm, its theoretical underpinnings in the lambda calculus and its practical implementation in specific languages. Students' theoretical understanding of computability will be expanded from the introduction in the undergraduate theory of algorithms module where a Turing machine approach was used. Students will be introduced to the notion of “functions as rules”.

Students will also learn how to use functional programming as a practical programming skill. Topics include side effect free programming and its benefits; first-class functions and higher-order functions; partial application and defining higher-order operations on aggregates, especially map, reduce/fold, and filter. Important new functional programming concepts including lazy evaluation and monads.

6.3. **CSC4021Z - Compilers 1 (*12 credits)***

Convener: Mr. Gary Stewart

When: Block 4

This course will introduce students to the inner mechanics of a modern programming language compiler or interpreter. Students will appreciate why programming languages are designed in particular ways and they will learn how to develop compilers and compiler-related tools. Course content will include: language classes, formal grammars, recursive descent parsing, tokenisers, parsing, and abstract syntax trees.

7. Elective Honours Modules

Coursework modules offered at the Honours level vary from year to year, depending on the current Computer Science staff. The following list of modules is provisional for this year and **subject to change**. You may take any module, as long as you satisfy the individual prerequisites listed. Full module descriptions appear in the Science Faculty handbook.

7.1. **CSC4023Z - Big Data Management and Analysis (*12 credits)***

Convener: Assoc. Prof. Maria Keet

When: Block 1

This course will enable students to understand the challenges of designing and implementing database applications at very large scale. They will know the approaches taken by big data technologies such as relational databases, NoSQL, Hadoop and data mining tools, and have practice in applying this knowledge.

The focus of this course is on systems designed for big data storage and analysis. Topics covered include NoSQL, Hadoop, HBase, HIVE, YARN and Apache Spark, as well as an introduction to data mining techniques and tools. The course concludes with a series of short presentations on new developments in database technology such as spatial, temporal, mobile, multimedia, text and social network data management.

7.2. **CSC4024Z - Human Computer Interaction (*12 credits)***

Convener: Assoc. Prof. Melissa Densmore

When: Block 2

This course will introduce you to basic concepts and practice around user-centred design of digital systems. The course covers how to design and evaluate interactive systems for real users both in the developed and developing worlds. We will look at both theory and practice of designing digital systems.

Topics include the design cycle, sketching and storyboarding, task analysis, contextual inquiry, conceptual models, usability inspection, human information processing, experience design, and qualitative and quantitative study design and evaluation. We will also invite guest speakers from industry and research to talk about their own experiences with user-centred design.

7.3. **CSC4025Z - Artificial Intelligence (*12 credits)***

Convener: Dr. Jan Buys

When: Block 4

This course will expose students to foundational concepts and computational techniques in modern Artificial Intelligence and their theoretical underpinnings in logic, search, optimisation and mathematical statistics. Students will also learn how to select and implement these techniques to solve various real world problems. Core topics will include: problem solving, knowledge representation and reasoning, machine learning and dealing with uncertainty, with selected topics from: planning, agents and natural language processing.

7.4. CSC4026Z - Network and Internetwork Security (*12 credits)*

Convener: Dr. Josiah Chavula

When: Block 2

The objective of this course is to introduce cryptographic techniques and protocols for secure exchange of information on networks and internetworks, and to examine the deployment of these in emerging technologies.

The course will cover risk issues (ISO27000; PoPI act); security services; conventional encryption (classical encryption techniques, DES/AES, key distribution, key generation); public-key encryption (RSA algorithm, key management, certification hierarchies); authentication & digital signatures; authentication and key exchange (Kerberos, Diffie-Hellman); electronic messaging security (S-MIME/PGP/WhatsApp); HTTP security (S-HTTP, SSL, capabilities); secure electronic commerce (SET); web application security (OWASP); web-services security (WS-Security, SAML); cloud computing security (public vs private clouds); critical infrastructure security (Stuxnet etc); Security Information & Event Management (SIEM) and next generation Security Operation Centres.

7.5. CSC4027Z - Computer Games Design (*12 credits)*

Convener: Prof. James Gain

When: Block 4

This course will introduce students to the techniques and technologies used in designing and programming computer games and related applications.

This course introduces high-level game programming concepts and practical game construction. By the end of the course, students will be able to design and implement simple 2D and 3D games. The course content include: appropriate terminology, methods, and tools for computer game development are introduced; fundamental algorithms for 2D game development; design and development of simple 3D and networked games; uncertainty and constantly changing gaming environments; and techniques for multiuser and distributed games.

7.6. CSC4028Z - High Performance Computing (*12 credits)*

Convener: Prof. Michelle Kuttel

When: *not offered in 2024*

Multithreaded computing is increasingly important for effective software development. However, knowledge and experience of both parallel algorithms and architectures is required in order to program a parallel computer effectively, particularly in the case of complex hybrid accelerator/multicore machines. This course cover methods for the practical development of parallel algorithms on multiple cores or GPUs.

This module covers the following areas: An overview of parallel computing, with a history of a parallel computing in general, clusters, multicore and accelerators; Parallel architectures – clusters, multicore machines and accelerators; General comparison of parallel programming models and methods; Thinking in parallel: Parallel algorithms and applications; Multithreaded computing for multiple cores; A motivation for general purpose computation on GPUs (GPGPU); The CUDA approach to multithreaded computing; CUDA threading and memory models; CUDA performance optimization; and Benchmarking, profiling and proving parallel performance.

7.7. CSC4029Z – Computer Graphics (*12 credits)*

Convener: Assoc. Prof. Patrick Marais

When: Block 2

This course will expose students to the foundational theory of 3D computer graphics and provide a short introduction to OpenGL and GLSL shader programming. Theory will encompass the formal description of 3D models and how these can be lit and rendered to produce a desired representation of a 3D scene. On the practical side, a series of short assignments will introduce basic OpenGL and shader programming and allow students to apply the theory they have learned.

Topics include: the rendering pipeline and rasterization; 3D mesh model representation; homogeneous coordinates; modelling and viewing transformations; shading, lighting and texturing; GLSL shader programming; OpenGL/WebGL programming; advanced rendering; ray-tracing.

7.8. CSC4013Z – Visualization (*12 credits)*

Convener: Prof. Michelle Kuttel

When: Block 1

Visualisation is the graphical representation of data with the goal of improving comprehension, communication, hypothesis generation and decision making. This course aims to teach the principles of effective visualisation of large, multidimensional data sets. We cover the field of visual thinking, outlining current understanding of human perception and demonstrating how we can use this knowledge to create more effective data visualisations.

8. Elective *External* Honours Modules

Students are encouraged to take external modules, subject to the subminima for external courses (Section 2.4). These courses may only be taken with permission from the lecturer of course convenor, as well as approval by the Honours Programme Coordinator. External courses have their own prerequisites and it is students' responsibility to ensure that they meet these prerequisites.

8.1. Mathematics Modules

Modules within the Department of Mathematics and Applied Mathematics have been approved as external courses by the Department of Computer Science. Students doing CS Honours can do an extra mathematics course on top of the normal allowance of 20 credits for outside courses. This is a maximum of 40 credits. These courses are counted as 20 credits and carry the (arbitrary) course codes MAM4019Z and MAM4020Z. Additional information can be obtained from the Department of Mathematics.[10]

[10] http://www.mamhonours.uct.ac.za/discrete-mathematics-and-theoretical-computer-science-modules

8.1.1. Complexity Theory (*20 NQF credits)*

Lecturer: Dr. Holger Spakowski

Semester: First

Prerequisites: UCT MAM 3rd year module 3DM (Discrete Mathematics) or equivalent.

Course Description: This course provides an introduction to major topics in computational complexity theory, which is one of the core areas of theoretical computer science. In computational complexity, we investigate the power of efficient computation. That is, we try to distinguish between computational problems that can be solved efficiently in practice and those that, though theoretically solvable, are not solvable in practice because of prohibitively large time or space requirements. The central open problem is the P versus NP problem.

8.1.2. Cryptography (*20 NQF credits)* [NOT offered in 2024]

Prerequisites: The course is geared towards Honours students in either Maths or Computer Science. Having done 2IA and 2LA is an advantage, but is not necessary. We assume some familiarity with matrices, but we will cover all the number theory and probability theory you need in the course.

Course Description: Cryptography is the mathematics of information security, which means keeping digital information secret or ensuring that it cannot be changed without detection. In this course we first cover the two kinds of secret key cryptosystems (block ciphers and stream ciphers), along with cryptographic hash functions. We then learn some computational number theory, before studying the public key cryptosystems and signature schemes RSA and ElGamal, and methods for solving the factoring and discrete log problems. If time permits, we finish with elliptic curve cryptosystems. Emphasis throughout is on how all these systems can be attacked (using maths).

8.1.3. Graph Theory (*20 NQF credits)*

Lecturer: Dr. Imran Allie

Semester: Second

Prerequisites: An undergraduate degree in mathematics, including some group theory.

Course Description: Graph Theory is an increasingly important area of modern mathematics. There are numerous applications of Graph Theory: Modelling the World Wide Web, the spread of disease, driving directions, and electrical networks, to name a few. This course, though, is delivered as a course of Pure Mathematics, i.e., it is a sequence of theorems and proofs.

8.1.4. Advanced Topics in Reinforcement Learning (*20 NQF credits)*

Lecturer: Prof. Jonathan Shock

Semester: Second

Prerequisites: Have read "Reinforcement Learning" by Sutton and Barto: https://web.stanford.edu/class/psych209/Readings/SuttonBartoIPRLBook2ndEd.pdf and to have a good grounding in Python.

Course Description: In this project-based module on modern methods of artificial intelligence, students will investigate advanced topics in Reinforcement Learning (RL). In the past, these have included multi-agent RL, curiosity-based RL, causality, meta-RL and many more. There will be weekly discussion sessions where students will give mini-presentations, and share ideas. The first three weeks gets everyone up to scratch on deep neural networks applied to RL. The final grade is based on a mini-thesis.

8.2. Statistics Module

Students may register for STA4026S (Analytics), provided that they have permission from the course convenor. Applications to take this course will be processed near the start of the second semester, and students may only register for the course after they get permission to take it.

8.2.1. STA4026S - Analytics (*18 NQF credits)*

Course Convenor: Dr. Etienne Pienaar

Semester: First

Prerequisites: Undergraduate degree that included a substantial degree of training in quantitative subjects and programming, as assessed by the course convener.

Course Description: This course will cover computationally-intensive statistical methods for analysing datasets of various sizes. The course will cover three broad sections: (1) Parallel and high-performance computing in R, (2) Supervised Learning and (3) Unsupervised Learning.

In the first section, students will learn how to use R to analyse large datasets on multiple computer processors, and UCT's own HPC cluster. The second section will expose students to machine learning techniques that are used to infer a regression or classification rule based on labelled training data, including regression and classification trees, bagging and random forests, boosting, neural networks. The last section will cover statistical methods for classifying observations into groups where the group memberships of the training data are not known in advance, including self- organising maps, association rule mining and cluster analysis.

8.3. School of Management Studies: Strategic Thinking (BUS4050W)

BUS4050W is the capstone course **available only to final year Business Science students**. This external course counts 36 credits.

BUS4050W aims to give students an opportunity to improve their strategic thinking ability. The course focuses on both classic strategic management thinkers and includes guest lectures who share their real world experience of strategic thinking. Consult the Faculty of Commerce handbook for more details.

9. Elective Masters Modules

The selection of coursework modules offered at the Masters level vary from year to year, depending on the interests of the current Computer Science staff. The current list of modules and details about the courses offered can be found in the Postgraduate student handbook. Some of these courses have additional prerequisites or can only accept a limited number of students.

Students who register for one of two of the course codes below will be able to indicate which Masters module(s) they wish to take; this information will then be passed on to the corresponding course convenors and students will be informed if they may take that particular module. If a student is not accepted to take a particular course, it is their responsibility to find another course to take, or to register for a different course code instead.

9.1. **CSC4007Z – Selected Honours Module in Computer Science (*12 credits)***

This is the first course code for taking a Masters-level course. All students who intend to take a Masters module should register for this course.

9.2. **CSC4010Z – Advanced Topics in Computer Science 2 (*12 credits)***

This is the second course code for taking a Masters-level course. Students should only register for this course code if they plan to take two Masters-level courses (or have already completed CSC4007Z).