UCT Computer Science

Graduate Student Handbook

2019

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 2019

This booklet contains details of the structure of the various graduate courses offered by the Computer Science Department at UCT, covering the Honours Year (fourth year), Masters Degree and Doctoral Degree, as well as some information about the department itself.

The booklet was compiled by the UCT Computer Science MSc, PhD and Honours coordinators, together with the head of department of Computer Science.

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 on our Postgraduate Degrees

The Department of Computer Science of the University of Cape Town has one of the best computer science departments in the country for pursuing graduate work. 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.

All of our postgraduate degrees are **full-time**; we do not offer part-time courses.

1.1. Degrees

1.1.1. PhD

The PhD is a research degree on an advanced topic under supervision. Examination is by thesis alone. A candidate shall undertake doctoral research and advanced study under the guidance of a supervisor(s) appointed by Senate. The thesis must constitute a substantial contribution to knowledge in the chosen subject, must show evidence of original investigation and give a full statement of the literature on the subject. The PhD degree demands that the candidate is able to conduct independent research on his/her own initiative. Through the thesis the candidate must be able to demonstrate that he/she is at the academic forefront in the topic selected, that the work is original and that it advances our knowledge in the relevant field.

1.1.2. Masters in Computer Science (by Coursework and Dissertation)

This course provides training in research in Computer Science, through the medium of a set of courses and a minor dissertation.

The course aims to provide students with an overview of those fields of Computer Science in which the department undertakes research, from which the student has to enroll for the Research Methods module, and selects six of the remaining coursework modules: Computational Geometry for 3D Printing, Distributed Scientific Computing, Evolutionary Computation, Information Retrieval, Intelligent Systems, Introduction to ICT for Development, Logics for Artificial Intelligence, Introduction to Image Processing and Computer Vision, Ontology Engineering, Data Visualization, Networks and & Internet Systems, and Human Computer Interaction. Upon successful completion of the coursework component, students will be required to register for the minor dissertation component and complete a suitable research project under supervision of an appropriate computer science academic staff member. The research component will expose the student to research methodology, experimental design, data analysis techniques, and dissertation writing skills. Students should be in a position to submit the final dissertation before the end of the second year.

1.1.3. Masters in Computer Science (by Dissertation only)

This course is primarily intended for students who completed Honours in Computer Science at UCT in the period 2014-2017. Other students must enrol in the Masters in Computer Science by Coursework and Dissertation, unless permission is

obtained from the Head of Department, based on a motivation provided by a potential supervisor.

This course consists of an investigation of an approved topic chosen for intensive study by the candidate (student), culminating in the submission of a dissertation. The dissertation shall demonstrate the successful completion of a programme of training in research methods, a thorough understanding of the scientific principles underlying the research and an appropriate acquaintance with the relevant literature. It must be clearly presented and conform to the standards of the department and faculty. The dissertation will usually consist of a report detailing the conduct, and analysis of the results of, research performed under the close guidance of a suitably qualified supervisor(s). The dissertation should be well-conceived and acknowledge earlier research in the field. It should demonstrate the ability to undertake a substantial and informed piece of research, and to collect, organise and analyse material.

1.1.4. Masters in Information Technology

The MIT degree is by coursework and research, using mainly self study materials for the coursework - there are no lectures. It is a two year conversion programme aimed at students without a degree in computer science or information technology/systems. The ideal candidate for this programme is someone who uses information technology in his or her job, but who has no formal university qualification in IT. A non-IT honours degree, or equivalent, is the minimum requirement for entry into the MIT course.

The rules pertaining to Masters in IT are maintained separately and can be accessed at http://www.cs.uct.ac.za/mit/

1.1.5. Honours in Computer Science/ Information Technology

Our Honours programmes are are a single year 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.

1.2. Programme Convenors

The 2019 Honours programme coordinator is ***Assoc. Prof Michelle Kuttel***, Room 304.02, Computer Science Building (email: mkuttel@cs.uct.ac.za).

The 2019 Masters in Computer Science and PhD programme coordinator is ***Prof Tommie Meyer***, Room 312, Computer Science Building (email: pgcoordinator@cs.uct.ac.za).

The 2019 Masters in Information Technology programme coordinator is ***Dr Melissa Densmore***, Room 312, Computer Science Building (email: mit@cs.uct.ac.za). Prof Tommie Meyer will take over as programme coordinator from July 2019.

The postgraduate programmes also have senior postgraduate students employed as Teaching Assistants (TAs). The role of the TA is to assist with course administration.

In addition, class representatives will be elected at the beginning of the year.

I.3. Rules and requirements for all postgraduate degrees

I.3.I. Role of the student

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

Graduate 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, graduate 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 r**esponding 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.

I.3.I.I. Role of Class Representatives

Class representatives will be elected for each class. Class representatives are expected to arrange regular meetings with the course coordinator, in order to resolve in consultation issues that arise during the course of 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, graduate class representatives are expected to schedule the examination timetable in the exam period at the end of each block, in

consultation with the appropriate lecturers. Individual course may have additional duties for the class representatives.

1.3.2. 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 graduate 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.

1.3.3. Fees

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

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.

1.3.4. Financial Assistance

Financial assistance is available for prospective graduate students. The *Financial Assistance for Postgraduate Study and Postdoctoral Research Handbook*[2] lists

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

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 July 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 is usually around the 30 September – though in recent years this has come as early as June; the notification date is February of the next year.

The University offers *UCT Council Honours Merit Scholarships* for students who achieve at least 65% in their undergraduate majors. The Faculty of Science offered scholarships to a limited number of PhD students each year - these must be applied for in the previous year.

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.

1.3.5. Facilities

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

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. All other postgraduate students are assigned either shared or dedicated workspaces in the **ICT4D lab** or **Room 300 lab**, with equipment provided for by the supervisors of the research.

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.

1.3.6. CS Department Seminars/Colloquia

All postgraduate students must allocate one hour per week for attendance at departmental colloquia. Computer Science colloquia are normally held during the lunch hour (1-2pm) and normally on a Thursday. **Please note that attendance of at least 75% of the colloquia is mandatory for Honours and Masters by C&D students**. Failure to do this will impact on the mark for Research Methods with a 10% penalty. A register of attendance at colloquia will be kept.

1.3.7. 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.

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

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.

Note that:

- A note from the UCT Wellness Centre is not a medical note.
- Generally, self-reported symptoms, after the fact, e.g. "Two days ago I was feeling ill", are not enough to constitute a medical note, even if you report this to a doctor and they issue a medical. Such medicals will usually say "according to the patient...". Such a note is not valid, since there is no way of corroborating the illness.

Students who miss a test, exam or any other formal assessment **will only be allowed admission to an alternate assessment**, where this is possible and scheduled, 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 HoD for approval.

I.3.8. Policy on 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 are available online or on request from the course convenors. **All graduate students are expected to familiarise themselves with the rule on plagiarism**.

I.3.9. Appeals Procedure

If a student feels that their marks are incorrect for any piece of marked work, 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.

I.4. Coursework Information

I.4.1. 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**.

1.4.2. 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 commitments and to even out the workload. 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.

1.4.3. Module Registration

Coursework modules are selected on registration, and changed to the module selection must be done via a change of curriculum form, signed by the course coordinator and the programme convenor; and then submitted to the Science Faculty Office.

Honours students may not register for more than 120 course credits all together. Masters students do not have restrictions but must give due consideration to the other requirements of the course.

You may only drop or add a module with the approval of the lecturer concerned and the programme coordinator. Such approval will not be granted if more than 1/6th of the lectures have already been given in the module concerned.

1.4.4. Examinations

Modules are usually examined after the completion of the block in which the module was given. External courses are usually examined in the University examination periods (May/June and October/November). However, the Department is free to schedule examinations at any sensible time after the completion of the relevant coursework. Examinations written outside the department are scheduled by the department in question. There is typically one two-hour examination per 12 credit module in Computer Science Honours and Masters. Open book and take-home examinations are preferred by some lecturers.

The examination timetable is the responsibility of the teaching assistant, the tutors and lecturers concerned and is drawn up shortly before the examination period.

After the mid-year examinations, students may be given an indication of how they performed. Note that only a provisional symbol is released as the exam papers will not have been seen by the external examiner at this stage.

2. PhD and Masters Degree Structure and Rules

1. Course Codes

PhD - CSC6000W

Masters by Coursework and Dissertation - CSC5001W (first year) and CSC5002W (second/subsequent year)

Masters by Dissertation - CSC5000W

2. Further Study

Students who successfully complete Masters in Computer Science at UCT are eligible to apply to proceed to a PhD in Computer Science.

3. Admission Requirements

The number of places in the Masters programme is limited and students are selected on merit from the list of applicants each year. Criteria for selection include the strength of your previous Computer Science qualifications, your relevant industrial experience and your maturity as a postgraduate student.

Students are admitted to the PhD degree on an individual basis only, after being accepted by a research supervisor in the department.

4. Orientation, Registration and Start of Year

The academic year for new MSc and PhD students begins on the first day of undergraduate lectures - consult the UCT calendar for start dates[4].

The orientation meeting will take place on the first day of term for all students. Attendance is compulsory!

Details about registration can be found here.

5. Coursework

Research Methods Course

ALL Masters in Computer Science and PhD students MUST attend the Research Methods seminar-based course in their first year of enrolment.

All students enrolled in the Masters by Coursework and Dissertation degree must complete 90 credits of coursework from the courses on offer for this degree. This includes the Research Methods course (CSC5020Z) for 18 credits, together with six elective coursework Masters courses. Details about the Research Methods course, as well as the available elective courses, can be found at the end of this booklet.

In exceptional cases, students may take approved courses in cognate departments with the prior permission of the course convenor.

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

6. Research Proposal

The research proposal is a document and presentation where you are proposing your research topic for your degree to a committee to get feedback on the work you wish to do. All PhD and Masters in Computer Science students need to prepare a research proposal, and give a presentation about the proposal in their first year of enrolment. Details about the proposal and the presentation can be found here.

7. Thesis

The thesis is the major written document produced at the end of the research conducted and submitted for examination. More details on the form and requirements of the thesis can be found in the Science Faculty Postgraduate Handbook.

Once the student has completed the thesis to the satisfaction of the university, and completed all coursework associated with the degree registered for, the student will then qualify for the degree.

3. Honours Degree Structure and Rules

The Department offers two Honours degree options: BSc (Hons) in Computer Science; and BSc (Hons) in Information Technology. The former degree is for students with a major in Computer Science from UCT, external students do the latter degree. Students in their final year of the Bachelor of Business Science with a major in Computer Science)are assigned the course code CSC4003W.

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 is accredited as meeting the academic requirements for Chartered IT Professional (CITP) certification[6] and partially meeting the requirements for Chartered Scientist (CSci) registration.

Our Honours students from previous years found employment in international companies in the USA (e.g. Facebook, Asana); 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 are creating their own startup companies. Many of our alumni work internationally, for software giants such as nVidia, Facebook, Microsoft and Amazon.

3. Further Study

Students who successfully complete Honours in Computer Science at UCT are eligible to apply to proceed to an M.Sc. in Computer Science by coursework.

4. Admission Requirements

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. Criteria for selection are the Computer Science mark achieved in each of your three years of undergraduate studies and, to a lesser degree, the marks achieved in mathematics. Students who have not achieved at least a 65% average in their final year of Computer Science will only be admitted under exceptional circumstances.

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

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

5. Orientation, Registration and Start of 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.

5.1. Registration for South African students

Please bring the following documents to registration on the first day:

- a certified paper copy of your Identity Document (ID).
- certified paper copies of your final transcript and proof of graduation (only for non-UCT Bachelor's degrees.).

5.2. Registration for 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 clearances from the International Academic Programmes Office. This pre-registration 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.

5.3. CSC4003W Registration

Registration for the CSC4003W course (Business Science students) is handled by the Commerce Faculty.

6. Coursework and Project

6.1. Structure of The Honours Year

NOTE: The Honours year commences two weeks before the undergraduate courses.

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

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

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

their full-time projects. The only exceptions to this structure are in the case of external modules taken by students.

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 by the start of the second quarter and various project-related milestones have to be met from then on until the final report for the project is due (usually in late September).

A total of 160 credits must be obtained during the course of the academic year. All compulsory modules must be completed by every student. Students may select any remaining modules as electives, with a minimum of 48 credits, up to a maximum of 68 credits. The project comprises the remaining 60 credits.

6.2. 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.

6.3. Project block

Approximately eight weeks have been reserved during the course of the year to allow students to focus entirely on their Honours project. Lectures and practicals will not run in the project block (currently this is Block 3).

6.4. Module Credits

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.**

6.5. Course Work

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

- You must complete the compulsory **Research Methods** (8 credits) module successfully.
- You must complete the compulsory **New Venture Planning** (8 credits) module successfully.
- You must complete the three compulsory modules: **Functional Programming** (12 credits), **Compilers 1** (12 credits) and **Compilers 2** (12 credits).
- You must obtain credit for at least **48 credits of elective material** (you may take at most 68).

[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.

- **At most 40 credits from Mathematics** courses or at **most 20 credits from other external departments** may be taken (this *does not apply to Business Science* students[10]).

6.6. The Major 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*.

6.6.1. Timing

The project topics are presented to the Honours class towards the end of the first quarter and allocated to the teams by the start of the second quarter.

Students are expected to start work from the second quarter onwards and meet their project supervisor weekly. A block of about eight weeks has been set aside in the second and third terms, 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 is handed out when the projects are assigned.

6.6.2. 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.

Each project group is required to produce a formal project proposal, which will be vetted by the staff at a formal project presentation. Guidelines for the proposals will be distributed once the projects have been approved.

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.** Contact the Honours programme coordinator at the start of the year for the full requirements. Please note that students are not guaranteed to be assigned their proposed project and the department reserves the right to reject such proposals.

6.6.3. Deliverables

The final project paper must be handed in to the Honours Coordinator 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.

[10] The CSC4003W course for Business Science students has only 80 credits of coursework. A course in Business Strategy is taken and **no courses outside of Computer Science or Mathematics are counted**. The average mark is therefore calculated as described on the best 80 credits.

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 may change from year to year, but are likely to include: a formal project proposal and presentation, a prototype, a poster, a project webpage and a self-reflection report.

6.6.4. 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 who 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.

6.7. Duly Performed Certificate

Students will only be allowed to proceed with the second semester if, by the end of the first semester, they have gained credit for at least:

1. **64 credits of coursework** (this includes the compulsory modules) or
2. 40 credits of coursework in CSC4003W.

Students who do not meet these requirements will be listed as having been *Refused a Duly Performed Certificate* and their class record will show DPR. Such students will be entitled to a refund of 50% of their course fees and may apply to repeat the course as outlined below.

6.8. Passing Honours

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

- at least **50% for their project mark**.

- A pass in all the compulsory modules.
- Sufficient elective modules to meet the requirements of at at least 100 coursework credits for the degree.

A student who achieves all of the above subminima will pass the course.

6.9. Policy on Repeating Honours

Students have no automatic right to repeat Honours if they fail to meet the requirements for awarding the degree. If a student wishes to repeat Honours, a new internal Honours application and a letter of motivation have to be addressed to the Honours Course Coordinator. Such applicants will be considered after all new students for the course who meet the criteria for admission have been accommodated. All applicants wishing to repeat the year, as well as students who do not meet normal admission criteria, will be considered together. All special applications for admission to Honours have to be made by the end of last week of December.

7. 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[11].

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 external lectures teaching the New Venture Planning and Professional Communications portions of CSC4017Z.

The module time table varies from year to year, **is subject to change**, and will be communicated by the Honours coordinator on the dedicated Honours Google calendar.

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

Convener: Associate Professor M M Kuttel

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

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

The course develops communication and writing skills and introduces basic research methodology. The first module of the course focuses on Professional Communications in general, including written and visual communication.

A second 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.

A third module teaches scientific writing and research methods for statistical analysis and evaluation of data.

7.2. CSC4020Z - Functional programming (*12 credits)*

Convener: Dr G Nitschke

When: Block 1 (may change)

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.

7.3. CSC4021Z - Compilers I (*12 credits)*

Convener: Prof. Maria Keet

When: Block 1 (may change)

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.

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

7.4. CSC4022Z - Compilers 2 (*12 credits*)

Convener: TBA

When: possibly *Block 2* (with a strong likelihood of moving to **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: semantic analysis, activation records, intermediate code, optimisations, basic block analysis, instruction selection, liveness analysis and register allocation.

8. 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.

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

Convener: Assoc. Prof. Sonia Berman

When: *Block 2*

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.

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

Convener: Dr Melissa Densmore

When: Block 1

This course will introduce you to basic concepts and practice around user-centred design of digital systems.

This 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.

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

Convener: Assoc. Prof. Deshen Moodley

When: **Block 4** (May move to *Block 2*)

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

8.4. CSC4026Z - Network and Internetwork Security (*12 credits)* (may not be offered in 2019)

Convener: Assoc. Prof. Andrew Hutchinson

When: **Block 4** (May move to *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.

8.5. CSC4027Z - Computer Games Design (*12 credits)* (will not be offered in 2019)

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

Convener: Assoc. Prof. Michelle Kuttel

When: **Block 4** (May move to *Block 2*)

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.

9. Elective *External* Honours Modules

Students are encouraged to take external modules, subject to the subminima for external courses (Section 2.4). **These external courses have to be approved by the Honours Programme Coordinator.** They will be weighted according to their relevance to computing in general. A critical aspect of undertaking such a module is obtaining permission from both

the lecturer and the head of the relevant department to undertake the course: they have to provide a contact person who will be responsible for providing your final mark on time. Such arrangements obviously have to be confirmed in writing.

9.I. Mathematics Modules

Modules within the Department of 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. In any year, two of the set of courses comprising Graph Theory, Cryptography and Complexity Theory are offered by the Mathematics department. In 2019, the courses are **Graph Theory** and **Cryptography**.

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.

9.I.1. Complexity Theory (CT, 36 lectures, first semester)

Lecturer or Convener: *Dr Holger Spakowski*

Semester: First

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

Course Objectives: 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.

9.I.2. Cryptography (CRYPT, 36 lectures, second semester)

Lecturer or Convener: *Dr Christine Swart*

Semester: Second

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 Content: 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).

9.1.3. Graph Theory (CGT, 36 lectures, Second semester) [NOT offered in 2018]

Lecturer: ***Dr David Erwin***

Semester: Not offered in 2019

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.

9.2. Statistics Modules

Students may register for STA4026S (Analytics), provided that they have permission from the Statistics department. This is typically run in the second semester.

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

BUS4050W is the capstone course **available only to final year Business Science students**. This external course counts 20 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.

10.Compulsory Masters/PhD Modules

The coursework module offered below are required for ALL Masters and PhD students. The details may vary from year to year, depending on the current Computer Science staff.

10.1. Research Methods (RM) (Course code: CSC5020Z)

Prerequisites: None

Course Objectives: The Research Methods course introduces students to research methods from the perspective of Computer Science, preparing them for the minor dissertation component of the degree.

Course content includes: Types of research, how to find papers and how to read papers; Research Ethics; Scientific and technical writing; Research methods: qualitative and quantitative and both; Literature reviews; Research proposals; Problem statements, research questions and hypotheses; Research statistics; Research planning and grant writing; and Academic career planning.

Credits: 18 credits (18+ lectures)

Lecturer or Convener: ***Prof. Tommie Meyer***

Assessment: Students in the Masters by Coursework and Dissertation degree will be assessed by a mark assigned to their literature review and research proposal.

Prescribed Book: Notes and slides will be distributed.

11. Elective Masters Modules

Coursework modules offered at the Masters level vary from year to year, depending on the interests of the current Computer Science staff. The following list of modules, as well as their details, is provisional for this year and **subject to change**. Modules are listed according to course code. You may enrol for any module, as long as you satisfy the individual prerequisites listed. You have to pass six elective modules in order to complete the coursework component of the Masters by Coursework and Dissertation programme. Once you have completed the coursework component, you will be eligible to register for the dissertation component of the degree.

II.I. Multi-Dimensional Data Visualization (MDVIS): CSC5008Z

Prerequisites: There are no specific prerequisites for this module, other than a background in computing. However, interest the design and development of graphical displays, data and graphics/visual art/aethetics/design is required to appreciate the course content. In addition, as this is an M.Sc course**, space may be limited**.

Keywords: data analysis, visualization, design, big data.

Course Objectives: This course forms part of the CSC5008Z Data Visualization course offered as part of the M.Sc in Data Science. Visualization is the graphical representation of data with the goal of improving comprehension, communication, hypothesis generation and decision making. As visualization is recognized a valuable tool for presentation and exploration of complex , multidimensional data sets, there is an increasing demand for data scientists with the ability to create effective and sophisticated visualizations. This course aims to teach the principles of effective visualization 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 in the design of effective interactive data visualizations.

Credits: 12 credits (8 lectures and one large assignment, which encompasses two in-class presentations and critique), with an option to do a 15-credit MSc version for students in other streams.

Lecturer or Convener: ***Assoc. Prof. Michelle Kuttel***

Course Content: This module will cover the following topics:

- Visual queries and how the mind works to process visual information
- Structuring two dimensional space
- Colour
- Visual space and time: depth perception and motion
- Visual objects: how to design visual objects that are easy to identify
- Theory and best practice in the design of multi-dimensional data graphics, interfaces and visualizations.

Number of lectures: 8, plus 2 long presentation sessions (a total of 14 hours).

Practical work: 30 hours

Practical Assignments: The single major practical will involve multi-stage design and testing of a graphical display of multi-dimensional data. Topics will be listed in the first week of the course and design stages will be presented to class for discussion and critique. The practical is expected to involve about 16 hours of work.

Assessment: Exam - 50%, Practical - 50%.

Recommended Books:

- Visual Thinking for Design by Colin Ware.
- The Visual Display of Quantitative Information by Edward R. Tufte (second edition).
- Visualization Analysis and Design by Tamara Munzner.

These recommended books are highly regarded internationally and will make wonderful additions to any Computer Scientist's library, but you don't *have* to buy them for this course.

II.2. Computational Geometry for 3D Printing (CGP): CSC5021Z Prerequisites: Computer Graphics from CSC3020H and C++ from CSC3022H. These prerequisites may be waived based on appeal, but sufficient evidence of the necessary skills will be required.

Keywords: computer-aided geometric design, fabrication, computer graphics

Course Objectives: To master surface and volumetric modelling concepts applicable to 3D printing.

Credits: 12 credits (10 lectures and three practicals)

Lecturer or Convener: ***Prof James Gain***

Course Content: The use of 3D printers for rapid prototyping is becoming increasingly prevalent. However, the process used by most current 3D printers of depositing thin layers of semi-molten material, which is known as Fused Deposition Modelling (FDM), is not without limitations. Factors such as material thickness and support structures need to be considered. This course will cover the theoretical concepts required for creating geometric models suitable for 3D printing. From a practical perspective, students will code modelling software, then design and ultimately print a 3D model.

Topics covered include:

- Geometry and Topology for Computer Graphics
- 3D Printing Concepts: Printing Hardware, Overhang Support, Applications
- Volumetric Concepts: Voxels, Computational Solid Geometry, Isosurface Extraction
- Surface Concepts: Parametric Surfaces, Mesh Smoothing, Free-Form Deformation

Number of lectures: 12 **Practical work:** 39 hours

Practical Assignments: Three practicals (10-15 hours each). These build successively towards a final self-contained modelling package and a 3D printed showpiece. A substantial practical extension (5 credits) is optional. There will be course readings assigned and completion of associated understanding assessments will be required.

Assessment: Exam: open book, 2 hours, 40%. Practical assessments, 15-20% each; Final printed show piece, 10%

Prescribed/Recommended Book: No prescribed text but research papers and readings will be provided.

II.3. Distributed Scientific Computing (DSC): CSC5022Z

Prerequisites: A basic understanding of computer networking and software systems.

Keywords: Grid computing, Cloud Computing, Infrastructure as a Service, Platform as a Service, Software as a Service.

Course Objectives: To provide an understanding of the basic components used to build Grid and Cloud computing systems, with a focus on how these can support Scientific Computing.

Credits: 12 credits.

Lecturer or Convener: ***Prof. Rob Simmonds***

Course Content: This course gives an overview of the components that make up Grid and Cloud computing environments. These include the components used to build distributed data and computing grids and the various "as a Service" systems referred to as Cloud computing. It also looks at how these are used for a range of activities, including supporting large scale Scientific Computing.

Number of lectures: 12 **Practical work:** 40 hours

Practical Assignments: 2

Assessment: Exam: open book, 2 hours, 60%; Practical assessments: 40%.

Prescribed/Recommended Book: There is no prescribed text book, but research papers and online references will be provided.

II.4. Evolutionary computation (EC): CSC5023Z

Prerequisites: Programming skills in Java (including data structures and algorithms) are required. A basic understanding of genetics and evolution is useful, but not required.
Keywords: evolutionary algorithms.

Course Objectives: Evolutionary computation entails the use of simulated biological evolution to solve problems that are difficult to solve using traditional computer science and engineering methods. This course examines different Evolutionary Algorithms (EAs) and the types of problems EAs are best suited to solve. Course objectives include: gaining an understanding of various evolutionary computation techniques, identifying EAs suitable for solving different types of problems, and how to apply EAs to optimisation, machine learning, or design tasks.

Credits: 12 credits.

Lecturer or Convener: ***Dr G.S. Nitschke***

Course Content: The topics covered include:

1. Introduction to Evolutionary Computation.
2. What is an Evolutionary Algorithm?
3. Genetic Algorithms.
4. Evolution Strategies.
5. Evolutionary Programming.
6. Genetic Programming.
7. Niching
8. Multi-Objective Optimisation.
9. Co-evolution.
10. Working with EAs.

Number of lectures: 12 **Practical work:** 32 hours

Practical Assignment: Implement an evolutionary algorithm to solve a given optimisation problem and use statistical analysis to compare results with another evolutionary algorithm (implemented by a classmate).

Assessment: Exam: closed book, 2 hours, 60%; Practical assignment: 40%.

II.5. Information Retrieval (IR): CSC5024Z (not offered in 2019)

Prerequisites: Basic understanding of XML data is required. Some background on statistics and linear algebra will be useful.

Keywords: search engines

Course Objectives: Understand how search engines work at an algorithmic level. Learn how to build and incorporate basic and specialized search engines into your own projects.

Credits: 12 credits

Lecturer or Convener: ***Assoc. Prof. Hussein Suleman***

Course Content:

- Introduction to Information Retrieval (IR)
- Models of Basic IR (Boolean, Vector, Probabilistic)
- IR evaluation and testbeds
- Stemming, Stopping, Relevance Feedback
- Models of Web and linked-data retrieval (Pagerank, HITS)
- Latent Semantic Analysis and Clustering
- Multimedia IR
- Cross-lingual and multilingual IR
- IR in Practice (CMSes, digital libraries, Web, social media, etc.)
- Selected topics from:
    - Distributed and Federated IR
    - Recommender Systems
    - Natural Language Processing for IR
    - Sentiment Analysis
    - Opinion Retrieval
    - Text Summarization

Number of lectures: 15 **Practical work:** 22 hours

Practical Assignments: 1-2 programming assignments: to use and/or extend existing IR tools or build a new tool from scratch.

Assessment: Exam (take-home): 40%; Assignments: 40%; Class participation: 20%

Prescribed/Recommended Book: There is no prescribed book, but after the course you will know how to find all the information you need online!

II.6. Intelligent Systems (INTSYS): CSC5025Z

Prerequisites: A strong mathematics background.

Keywords: Artificial Intelligence, Bayesian networks, machine learning, intelligent systems

Course Objectives: This module provides provides an overview of modern intelligent systems, including their design and implementation, and current research trends in the area. The course will focus on Bayesian Artificial Intelligence, decision theory and statistical

Credits: 12 credits.

Lecturer or Convener: Assoc. Prof. Deshen Moodley

Course content: Topics will include:

- Overview of intelligent systems: Top down versus bottom-up AI, Cognitive computer systems
- Bayesian AI
- Statistical machine learning
- Designing and implementing intelligent systems
- Streaming sensor data

Number of lectures: 14-16 **Practical work:** 24 hours

Practical Assignments: Three assignments – Bayesian AI, statistical machine learning, design and implementation of an intelligent system.

Assessment: 2 hr open book exam: 50%, Practical assessments: 15-20% each.

Prescribed/Recommended Book: Extensive readings, lecture notes and papers will be provided.

II.7. Introduction to ICT for Development (ICT4D): CSC5026Z

Prerequisites: None.

Keywords: ict4d, hci4d, socio-economic development, social good

Course Objectives: Understand basic ideas underlying ICT4D and how they are used in practice. Learn about and critically evaluate ICT4D projects. Learn how to design and evaluate development-oriented computing projects.

Credits: 12 credits.

Lecturer or Convener: ***Dr Melissa Densmore***

Course Content:

- Introduction to key terminology around socio-economic development
- Key concepts in ICT4D (e.g. social inclusion, after access)
- Case studies in specific domains, including healthcare, agriculture, mobile money, education, etc.
- Critical evaluation of ICT4D projects

Number of lectures: 16 **Practical work:** 16 hours

Practical Assignments: There will be three individual assignments, and students will be expected to work in groups to lead one case study discussion.

Assessment: Exam: none; Practical assessments: 25% each; Case Study Presentation: 25%

Prescribed/Recommended Book: Geek Heresy, by Kentaro Toyama

II.8. Logics for Artificial Intelligence (LAI): CSC5027Z

Prerequisites: Familiarity with basic discrete mathematics is required. Prior exposure to logic is recommended.

Keywords: logic, knowledge representation and reasoning, computational logic, description logics, logic-based ontologies.

Course Objectives: This course will introduce students to logics used in the area of Knowledge Representation - a subarea of Artificial Intelligence.

Credits: 12 credits.

Lecturer or Convener: ***Prof. Tommie Meyer***

Course Content: Logic plays a central role in many areas of Artificial Intelligence. This course will introduce students to Description Logics, a family of logics frequently used in the area of Knowledge Representation and Reasoning. Description Logics are frequently used to represent formal ontologies. Topics covered include the following:

1. The Description Logic ALC
2. Reasoning in Description Logics with Tableaux Algorithms
3. Reasoning in the EL family of Description Logics
4. Query Answering

Number of lectures: 16 **Practical work:** 22 hours

Practical Assignments: Students will be given a number of assignments. This may include an assignment involving the Protégé ontology development environment.

Assessment: Exam: open book, 3 hours, 50%; Assignments: 50%.

Prescribed/Recommended Book: None. Extensive lecture notes will be provided.

II.9. Ontology Engineering (OE): CSC5028Z

Prerequisites: Experience in modelling (ER, UML Class diagrams) and some familiarity with logic will be helpful.

Keywords: ontologies, modelling, OWL, Description logics, Semantic Web, automated reasoning

Course Objectives: The principal aim of this module is to provide the participant with an overview of ontology engineering—including language features, automated reasoning, and top-down and bottom-up ontology development—and a main application field being the Semantic Web.

Credits: 12 credits.

Lecturer or Convener: ***Assoc. Prof. Maria Keet***

Course Content: Ontologies are used in a wide range of applications, such as data integration, recommender systems, e-learning, semantic scientific workflows, and natural language processing. While some of these applications pass the revue, the main focus of the course is on the ontologies. The topics covered include the following:

1. Logic foundations for ontologies
   - Languages (Description Logics, OWL)
   - Automated reasoning (class and instance classification, satisfiability and ontology consistency checking)
2. Ontology development
   - Ontology engineering, top-down: foundational ontologies, ontology design patterns

- Ontology engineering, bottom-up: exploiting legacy material, such as relational databases, thesauri, text
- Methodologies for ontology development and maintenance, methods to enhance ontology quality and to automate some aspect of the methodology

Number of lectures: 16 **Practical work:** 20 hours **Lecture block:** Block 3

Practical Assignments: There will be two assignments: developing a small ontology in an Ontology Development Environment and a group project on a selected topic that delves deeper into a specific OE topic.

Assessment: Exam (closed-book but with some material provided) - 50%, assignments - 50%.

Prescribed/Recommended Book: An Introduction to Ontology Engineering by Maria Keet.

II.10. Introduction to image processing and computer vision (ICV): CSC5029Z

Prerequisites: Basic Linear Algebra (matrices, vectors etc); familiarity with Fourier Analysis or functional analysis would be useful. For the practical work, familiarity with a GUI toolkit would be useful.

Keywords: image processing; computer vision; segmentation; feature detection

Course Objectives: To introduce students to basic concepts in computer vision and image processing, oriented towards solving real world, practical image analysis problems. The student will be introduced to basic concepts from digital signal processing, and a foundation built that will allow understanding of how more sophisticated schemes such as image analysis/segmentation which can be used to describe image and volumetric data at a higher, more useful, levels of abstraction. Research papers and a case study will be examined which relate this to real-world problems. Students are expected to read the papers, and will be assigned papers to present. A class participation mark will be kept as part of the practical record.

Credits: 12 credits.

Lecturer or Convener: ***Assoc. Prof. Patrick Marais***

Course Content: A number of lectures (as indicated below) will be presented by the course convener, interspersed with paper/review sessions in which topical papers are discussed by the students and convener.

- Basic Signal processing 1
- Image Transforms & Operations 2
- Simple Image Features 2
- Image Segmentation & Registration 1
- Fundamental Segmentation techniques 3
- Machine Learning & GAs in CVision 2
- Case Study 1
- Paper Reviews 6

Number of lectures: 12 + 6 Paper Presentation slots

Practical work: 30 hours

Practical Assignments:

- Self-assessment exercises available (not for credit)
- 3 Paper Sessions (2 papers each plus student presentations)
- 4 week programming project

Assessment: Exam: Open Book; 2 hours. DP Requirement: 50% in class record (composed of prac and review questions). Class Record: Practical 60%, Paper Discussions 40%. Final Mark: Exam 30%, Class Record 70%.

Prescribed/Recommended Book: There is no prescribed book: some notes and web resources will be provided.

Recommended Book: Algorithms for Image Processing and Computer Vision, J. R. Parker, Wiley Computer Publishing, 1997.

Useful Reference: Image processing, Analysis and Machine Vision (4th Ed.), Milan Sonka et al, Wadsworth Publishing, 2014.

II.II. Advanced Topics in Computer Science Master's I: CSC5030Z

In 2019 this will be a course on High Performance Computing. Classes will be at the same time as the classes for the honours course CSC4028Z. To register for this course, contact Assoc. Prof. Michelle Kuttel and cc the Postgraduate Coordinator.

Convener: Assoc. Prof. Michelle Kuttel

II.I2. Advanced Topics in Computer Science Master's II: CSC5031Z

Do not register for this course unless you have spoken to the Postgraduate Coordinator.

II.I3. Networks & Internet Systems (NIS): CSC5032Z

Prerequisites: Working knowledge of computer networks. For practicals, some familiarity with the Linux command line interface will be useful.

Keywords:

Course Objectives: The objective is to gain advanced understanding of techniques for traffic engineering and quality of service in internet architectures. The course focuses on advanced topics in internetworking, traffic engineering, and mechanisms for measuring performance and Quality of Service (QoS) for network services and the Internet.

Credits: 12 credits.

Lecturer or Convener: *Dr. Josiah Chavula*

Course Content: New Network and Transport Protocols (IPv6, Mobile IP, IP Multicast, Multipath TCP, QUIC); Routing and Traffic Engineering (Interdomain Routing and Traffic Engineering with Border Gateway Protocol); Traffic Engineering with Overlay Networking (MPLS/GMPL, Location/Identifier Separation Protocols, Software Defined Networking and

Network Function Virtualization); **Internet Measurements** (Quality of Service and Quality of Experience (QoS and QoE), IP Traffic Monitoring and Analysis). In addition, selected reading/discussion topics will be included: Cloud Infrastructure; Content Delivery Networks; Internet Access in the Developing World, Community Networks; ICT4D, Online Data Protection and Online Censorship.

Number of lectures: 15 **Practical work:** 25 hours

Practical Assignments:

- 1 Paper Session (paper reading plus student presentations)
- 3 week network/architecture analysis project

Assessment: Assessment: Assignments: 40%. Discussion sessions: 15%. Active Participation in Class: 5%. Final Exam: 40%

Prescribed Book: There is no prescribed book: some notes and web resources will be provided.

Recommended:

- James F. Kurose, Keith W. Ross. Computer Networking : A Top-down Approach, Pearson, 2012 (6th edition)
- Research papers (from Journals and Conferences)

II.I4. Human Computer Interaction (HCI): CSC5033Z

Prerequisites: None

Keywords: User-centered Design

Course Objectives:

This course will introduce you to advanced concepts and practice around user-centred design of digital systems.

Credits: 12 credits.

Lecturer or Convener: ***Dr. Melissa Densmore***

Course Content:

This 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.

This course is offered concurrently with Honours HCI. Masters students will be expected to do additional readings, to attend a weekly discussion on the reading, and to develop and evaluate an artefact produced for the course.

Number of lectures: 15 **Practical work:** 30h

Practical Assignments: Individual Assignments, Group Project

Assessment: Participation 10%, Practical Assessment: 20%, Group Project: 40%, Final Exam: 30%

Prescribed/Recommended Book: None