Handbook for MSc Programs in ComputerScience, Swansea University Academic Year 2017/18 Vers 2

Anton Setzer (Editor)[1]

28 September 2017

[1]Based on handbooks edited by John Sharp.

1 Abbreviation

In the following

- CS stands for the MSc in Computer Science (generalist);
- ACS stands for the MSc in Advanced Computer Science;
- AST stands for the MSc Advanced Software Technology;
- DS stands for the MSc Data Science.

2 Main Points of Contact

- The computer science and mathematics student office is located at Far-123, and will help with most administrative issues.
- For problems regarding lectures please contact first your lecturer, who should be willing to help you.
- For general academic problems, please contact your academic mentor (often referred to as tutor), who will be assigned to you soon.
- The course coordinator for the MSc in Computer Science (CS) is Dr. O Kullmann, Room 958, Talbot Building, O.Kullmann@swansea.ac.uk.
- The course coordinator for the MSc in Advanced Computer Science (AST) and MSc in Software Technology (AST) is Dr. AG Setzer, Room 952, Talbot Building, a.g.setzer@swansea.ac.uk.
- The course coordinator for the MSc in Data Science (DS) is Dr. X Xie, Room 510, Faraday Tower, X.Xie@swansea.ac.uk
- The coordinator for the MSc project is Mr CJ Whyley, Room 954, Talbot Building, C.J.Whyley@Swansea.ac.uk.
- The overall coordinator for the postgraduate taught schemata is Dr AG Setzer, details as above.
- The Programme Director Computer Science for all teaching in computer Science is DR M Seisenberger.
- The Head of Department is Professor A Beckmann.
- The Director of Learning and Teaching of the College of Science is Dr N Harman.
- The head of College is Professor Matt Jones.

3 Overview over Academic Year

All dates are subject to confirmation.

Thursday 28th September 2017
Induction Lecture + Enrolment, Faraday H, Faraday Building
1pm for MSc in Computer Science (generalist)
3pm for MSc Adv. CS, Adv Sofware Technology, Data Science.

Monday 2 October - Friday 15 December 2017
First Semester teaching

Friday 6 October 2017, 2:00 pm
Hand in module selections to student information office room 123 Faraday Building
Please use the form for your degree scheme (ACS, AST, or DS)

Thursday 11 January – Friday 26 January 2018
First assessment period (subject to confirmation)

Monday 29 January – Friday 2 February 2018
Employability week (details to be announced later)
Note that students are expected to be available in case their attendance is required by the university.
Any examinations deferred due to adverse weather will be scheduled during this week.

Monday 5 February – Friday 23 March and
Monday 16 April – Friday 11 May 2018
Second Semester teaching

Monday 14 May – Friday 18 May 2018
Revision week for semester 2 modules
Note that students are expected to be in attendance this week and timetabled revision sessions may be arranged

Monday 21 May – Friday 15 June 2018
Second assessment period (subject to confirmation)

Wednesday 15 August – Friday 24 August 2018 (subject to confirmation)
Supplementary assessment period

Sunday 30 September 2018 (subject to confirmation – likely to change because of Sunday)
Deadline for submission of Dissertations for students who do not need to sit

supplementary examinations. It is expected that these will be considered at an Award Board held in mid November 2017 (date might change).

Saturday 15 December 2018
Anticipated deadline for submission of Dissertations for students who sat supplementary examinations. It is expected that these will be considered at an Award Board held in March 2018 (date might change).

January 2019 (exact date to be confirmed)
Graduation ceremony for students who submitted in September 2018

July 2019 (exact date to be confirmed)
Graduation ceremony for students who submitted their dissertations in December 2018

4 Useful Web Pages

4.1 Link to this Handbook

- Page containing this handbook and other material for access without blackboard (only temporarily until all students have access to blackboard)
  `http://www.cs.swan.ac.uk/~csetzer/admin/mscOrganisation/2017/`
- Afterwards it will be moved to the computer science student information pages on blackboard (folder Handbooks 2017-18)

4.2 Other Links

- The login details for most websites (if login is required) are the same as your login details given to you by the university.
- University Academic Guide with Official Regulations
  `http://www.swansea.ac.uk/registry/academicguide/`
  In doubt please refer to the regulations as detailed there.
- College of Science Intranet
  Contains e.g. information about assessment schedule, timetable, coursework submission (especially when physical copies are required) and much more:
  `https://science.swansea.ac.uk/intranet/`
  - College of Science Postgraduate Taught Student Handbook is accessible from the College of Science Intranet.
    `https://science.swansea.ac.uk/intranet/handbook/pg`
- MyUni
  `http://myuni.swan.ac.uk/`
  Access to most university services, especially
  - University Intranet.
    Contains e.g. details about your enrolment status, information about modules selected, provisional results, etc:
    `https://intranet.swan.ac.uk/login/`
  - Blackboard, the University's E-learning Platform.
    Here you will find pages for all the modules you are enrolled on (there may be a short delay between enrolment on an optional module and you being enrolled on its Blackboard site). There are also "departmental student information pages" containing handbooks, general non-module specific material, and useful forms.
    `https://blackboard.swan.ac.uk/webapps/portal/frameset.jsp`
  - The **student information pages** on blackboard contain documents provided by the Department of Computer Science for students of more general nature, including **past exam papers**, **handbooks** etc.

  - Swansea University Login (e.g. **Email**)
    `https://home.swan.ac.uk/`

- Library Services (iFind)
  `https://ifind.swan.ac.uk/discover/`
- Department of Computer Science.
  `http://www.swansea.ac.uk/compsci/`
- College of Science.
  `http://www.swansea.ac.uk/science/`
- Swansea University.
  `http://www.swansea.ac.uk/`

5 Extra Support Available

- Free English language classes will be provided by English Language Training Service.
  Details will be given to you later.
- Centre for Academic Success (Library)
  - Academic Success Programme – includes a variety of free courses such as academic writing, critical thinking and presentation skills.
  - Specialist free tuition for Academic Success – available to eligible students in receipt of the Disabled Students Allowance (DSA) or other funding streams.
- Careers service is available.
- Library information centre (LIC) provides books and PCs.
- Departmental labs:
  - Windows PCs as in LIC: Talbot 043 and Faraday 001.
  - Linux PCs in Faraday 500.
  - Embedded system lab in Talbot 946.
  - Additional PCs are available in the Wallace and other buildings of the university.
- MyUni Student Hub: Located in the stable block in Singleton Abbey.
- Student Support Services currently in the Keir Hardy Building.
- Wellbeing service in Horton Building (behind the Digital Technium).
- Student Union Advice and Support Centre located in Fulton House.

6 General Information

- All students will be allocated an academic mentor (formerly called personal tutor), and tutorial sessions to discuss the course and related matters will be held once a fortnight. Attendance at these is compulsory. Details will be made available later.

- Computer Science is mainly located in the Faraday Tower and Talbot building. The 3 buildings Faraday Building, Faraday Tower and Talbot building are connected at level 1 and 2. The main entrance is from the mall, where one enters level 1 of the buildings. One enters first the Faraday Building, than, beginning with the stairs, the Faraday Tower, and at the end of the corridor one enters the Talbot building. Most computer science staff offices are either on third or fifth floor of the Faraday Tower or on the first (mall level) floor of the Talbot building. For access to the Computer Science offices turn right at the end of the corridor coming from the entrance at the mall.

- The main Student Information Office is Room 123 on the first (mall level) floor of the Faraday Tower.

- If you have a problem with a particular lecture course, the first person you should approach is the lecturer concerned. For problems of a more general nature, you should contact your academic mentor..
  If all else fails you should see the CS course coordinator (Dr O Kullmann), ACS/AST course coordinator (Dr Setzer, room Talbot 952), or the MSc data science course coordinator (Dr. Xiangha Xie).

- Absences

  - All requests for absence for any reason by any Tier 4 visa students (UG, PGT, PGR) must be handled by the Compliance Team (tier4attendance@swansea.ac.uk) of the university. Please don't ask lecturers on advice, since they haven't been trained regarding this.
  - Any other students unable to attend lectures etc for a period exceeding three consecutive days are required to complete an absence form. These are available from the Student Information Office (Faraday Tower, room 123) Students of this category absent from lectures for 7 or more consecutive days need to provide a medical certificate.
  - There is an upper limit of 60 days of absence permitted per Academic Year.
  - Attendance of examinations is compulsory. You need a medical certificate or other documentation in case you miss an exam.

- In general we recommend if possible that you let us know about any circumstances that may affect your study as soon as they happen. This

allows us to give you appropriate advice, and to direct you to appropriate organisations within the university which can deal with such matter in a professional way. If you give proper documentation in time regarding extenuating circumstances using the appropriate forms, we can take them into account regarding your assessments and examinations. Note that proper documentation is needed for that. Extenuating Circumstances forms are available from the Student Information Office (Room 123).

- You should always let the University and the department know of any change of address etc so that we can keep our records up-to-date, and know where to contact you if necessary.

7 Progression and Assessment

Full details of the rules for assessment and progression can be found in the Academic Guide published on-line by the Academic Registry. The following information is provided for guidance only and in the event of there being any inconsistency **the University published rules will apply**..

- The MSc degree is divided into two parts: Part One consists of 120 credits of taught modules. Part Two is a 60 credit project.
- The pass mark for all Level M modules is 50%.
- Students who pass 120 credits may proceed to Part Two.
- Students who fail to pass all 120 credits in Part One at the first attempt are normally given the opportunity to sit supplementary examinations in August or present new coursework as appropriate in all failed modules. The marks for such modules will be capped at 50%. Students who choose not to resit a failed module, or assessment component, will be awarded a mark of 0%, but the Progression Board may refer to a higher January or May/June mark in order to make its progression decisions.
- If after the supplementary examinations students have accumulated at least 90 credits at 50% or more, an average of 50% or more, and no module marks below 40%, then they may proceed to Part Two. Students who fail to reach the standard required will fail the scheme, and will not be allowed any further attempts to redeem their failure.
- Students who pass 60 credits or more of the modules in part 1 are eligible for the award of a Postgraduate Certificate. Students who pass at least 90 credits of the modules in part 1 and have up to 30 credits of condonable failures, i.e. modules between 40 and 49 %, but choose not to proceed to the project phase, or fail their project are eligible for the award of a Postgraduate Diploma.
- Students who have passed part 1 with an average of 70% or more will be awarded an MSc with Distinction, if the mark for their project is also 70% or more.
- Students who have passed part 1 with an average between 65% and 69 %, will be awarded an MSc with Distinction, if the average of their Part One average and the mark for their project is 70% or more.
- Students who have an average of 60% or more in Part One and a mark of 60% or more for their dissertation, but fail to qualify for an MSc with Distinction, will be awarded an MSc with Merit.
- Students who have an average between 57% and 59% in Part One, will be awarded an MSc with Merit if the average of their Part One average and the mark for their project is 60% or more.

- Students who qualify for a Postgraduate Diploma or Certificate may be eligible for the award of a Distinction or Merit grade if their average is 70% or more, or 60% or more respectively.
- If a candidate is unable to attend an examination during the normal assessment periods due to illness or other extenuating circumstances then they may be allowed to sit the examinations during the supplementary examination period as a "first-sit" candidate.

7.1 Assessment

- Coursework needs to be your own work. Plagiarism is not allowed. Software is in use to check for plagiarism.
- Examinations are unseen. You are not allowed to have notes and books with you. It is up to the examiners to allow the use of university dictionaries (English only) and/or university calculators.
- Some components of modules may be assessed by oral exams. This could be vivas, demonstration of software, or presentations.

7.2 Attendance

- Tutorials are compulsory, registers are taken.
- You are expected to attend lectures. You need to swipe your student card at the beginning of the lecture.
- Extensions for coursework may be possible if you have well documented extenuating circumstances.

8 Module Requirements for Programmes

8.1 MSc in Computer Science (CS)

Compulsory Modules

CSCM41 Programming in Java (15 credits, Semester 1)
CSCM53 Computer Science Concepts (15 credits, Semester 1)
CSCM59 Relational and Object-Oriented Database Systems (15 credits, Semester 1)
CSCM10 Computer Science Project Research Methods (15 credits, Semester 1 + 2)
CSCM94 Software Engineering Principles (15 credits, Semester 1 + 2)

CSCM12 Software Concepts and Efficiency (15 credits, Semester 2)

CS-M20 MSc Project (60 credits, Summer)

Optional Modules (all 15 credits)

Select 2 modules (30 credits) from both semesters together
Please note that CSCM18 IT-Security is not available for the MSc in CS

Optional Modules Semester 1

CSCM13 Critical Systems
CSCM27 Visual Analytics**
CSCM35 Big Data and Data Mining*, **
CSCM48 Web Application Development
CSCM58 High Performance Computing in C/C++*, **
CSCM75 Logic in Computer Science
CSCM77 Computer Vision and Pattern Recognition**
CSCM98 Operating Systems and Architectures

Optional Modules Semester 2

CSCM37 Data Visualization
CSCM39 Human Computer Interaction
CSCM45 Big Data and Machine Learning
CSCM64 Software Testing
CSCM67 Graphics Processor Programming
CSCM68 Embedded Systems Design*
CSCM79 Hardware and Devices*
CSCM85 Modelling and Verification Techniques

* **Places on this module are restricted**

** **This module is only suitable for students who have a substantial programming background; there may be timetable clashes. Please consult the lecturers of that module for advice**

Unavailable modules:	Semester 1:	CS-M41, CSCM01, CSCM70, CSCM78
	Semester 1+2:	CSCM04, CSCM94
	Semester 2:	CSCM18

8.2 MSc in Advanced Computer Science (ACS)

Compulsory Modules

CSCM10 Computer Science Project Research Methods (15 credits, Semester 1 + 2)
CS-M20 MSc Project (60 credits, Summer)

Optional Modules (all 15 credits)

Select 7 modules (105 credits) from both semesters together

Optional Modules Semester 1

CSCM13 Critical Systems
CSCM27 Visual Analytics
CSCM35 Big Data and Data Mining*
CSCM37 Data Visualization
CSCM48 Web Application Development
CSCM58 High Performance Computing in C/C++*
CSCM75 Logic in Computer Science
CSCM77 Computer Vision and Pattern Recognition
CSCM98 Operating Systems and Architectures

Optional Modules Semester 2

CSCM18 IT-Security: Theory and Practice
CSCM39 Human Computer Interaction
CSCM45 Big Data and Machine Learning
CSCM64 Software Testing
CSCM67 Graphics Processor Programming
CSCM68 Embedded Systems Design*
CSCM79 Hardware and Devices*
CSCM85 Modelling and Verification Techniques

***Places on this module are restricted**

Unavailable modules:	Semester 1:	CS-M41, CSCM01, CSCM41, CSCM53, CSCM59, CSCM70, CSCM78
	Semester 1+2:	CSCM04, CSCM94
	Semester 2:	CSCM12

8.3 MSc in Advanced Software Technology (AST)

Compulsory Modules (all 15 credits except summer Project)

CSCM01	Software Engineering Project Planning and Management (Semester 1)
CSCM04	Software Team Project (Semester 1 + 2)
CSCM10	Computer Science Project Research Methods (15 credits, Semester 1 + 2)
CSCM64	Software Testing (Semester 2)
CS-M20	MSc Project (60 credits, Summer)

Optional Modules (all 15 credits)
Select 4 modules (60 credits) from both semesters together

Optional Modules Semester 1

CSCM13	Critical Systems
CSCM27	Visual Analytics
CSCM35	Big Data and Data Mining*
CSCM48	Web Application Development
CSCM58	High Performance Computing in C/C++*
CSCmM75	Logic in Computer Science
CSCM77	Computer Vision and Pattern Recognition
CSCM98	Operating Systems and Architectures

Optional Modules Semester 2

CSCM18	IT-Security: Theory and Practice
CSCM37	Data Visualization
CSCM39	Human Computer Interaction
CSCM45	Big Data and Machine Learning
CSCM67	Graphics Processor Programming
CSCM68	Embedded Systems Design*
CSCM79	Hardware and Devices*
CSCM85	Modelling and Verification Techniques

***Places on this module are restricted**

Unavailable modules:	Semester 1:	CS-M41, CSCM41, CSCM53, CSCM59, CSCM70, CSCM78;
	Semester 1+2:	CSCM94;
	Semester 2:	CSCM12

8.4 MSc in Data Science (DS)

Compulsory Modules (all 15 credits except summer Project)

CSCM27 Visual Analytics (Semester 1)
CSCM35 Big Data and Data Mining (Semester 1)
CSCM70 Mathematical Skills for Data Scientists (Semester 1)

CSCM10 Computer Science Project Research Methods (15 credits, Semester 1 + 2)

CSCM45 Big Data and Machine Learning (Semester 2)

CS-M20 MSc Project (60 credits, Summer)

Optional Modules (all 15 credits)
Select 3 modules (45 credits) from both semesters together
Please note that optional modules are restricted to these modules, which are relevant to Data Science

Optional Modules Semester 1
CSCM58 High Performance Computing in C/C++*
CSCM77 Computer Vision and Pattern Recognition
CSCM98 Operating Systems and Architectures

Optional Modules Semester 2
CSCM37 Data Visualization
CSCM39 Human Computer Interaction
CSCM67 Graphics Processor Programming
CSCM85 Modelling and Verification Techniques

*__Places on this module are restricted__

Unavailable modules:	Semester 1:	CS-M41, CSCM01, CSCM13, CSCM41, CSCM48, CSCM53, CSCM59, CSCM75, CSCM78;
	Semester 1+2:	CSCM04, CSCM94;
	Semester 2:	CSCM12, CSCM18, CSCM64, CSCM68, CSCM79, CSCM85

8.5 Part-Time 3-year Schemes

Students on a part time 3-year scheme have to take

- 60 credits in year 1
- 60 credits in year 2
- carry out the project (60 credits) in year 3.

Students have with their first 2 years the same compulsory and same optional modules as the corresponding full time scheme.

Please discuss the choices of modules with the coordinator of your scheme (Dr. Kullmann in case of MSc in Computer Science, Dr Setzer in case of AST and ACS Dr Xie in case of MSc in data science.

The distribution of compulsory modules is as follows:

- MSc in Computer Computer Science:
  - Year 1: CSCM41, CSCM53 (semester 1)
    CSCM94 (semester 1 + 2)
    CSCM12 (semester 2)
    No options to be taken
  - Year 2: CSCM59 (semester 1)
    CSCM10 (semester 1 + 2)
    Choose 30 credits from optional modules (note CSCM18 not available).
  - Year 3: CS-M20 (project, 60 credits).
- MSc in Advanced Computer Science:
  - Year 1: No compulsories Options: choose 60 credits of optional modules
  - Year 2: CSCM10 (semester 1 + 2)
    Choose 45 credits of optional modules
  - Year 3: CS-M20 (project, 60 credits).
- MSc in Advanced Software Technology:
  - Year 1: CSCM01 (Semester 1)
    CSCM04 (Semester 1/2)
    CSCM64 (Semester 2
    Options: choose 15 credits of optional modules
  - Year 2: CSCM10 (semester 1 + 2)
    Choose 45 credits of optional modules
  - Year 3: CS-M20 (project, 60 credits).

- MSc in Data Science
  - Year 1: CSCM70, CSCM35 (Semester 1)
    CSCM45 (Semester 2)
    Options: choose 15 credits of optional modules (note that the list of optional modules is restricted to those relevant to data science)
  - Year 2: CSCM27 (Semester 1) CSCM10 (semester 1 + 2)
    Choose 30 credits of optional modules (same restrictions as in year 1 apply)
  - Year 3: CS-M20 (project, 60 credits).

8.6 Part-Time 2-year Schemes

Students on a part time 2-year scheme have to take

- 90 credits in year 1
- 30 credits of taught modules and the project (60 credits) in year 2.

Students have with their first 2 years the same compulsory and same optional modules as the corresponding full time scheme.

Please discuss the choices of modules with the coordinator of your scheme (Dr. Kullmann in case of MSc in Computer Science, Dr Setzer in case of AST and ACS Dr Xie in case of MSc in data science.

The distribution of compulsory modules is as follows:

- MSc in Computer Computer Science:
  - Year 1: CSCM41, CSCM53 CSCM59 (semester 1)
    CSCM94 (semester 1 + 2)
    CSCM12 (semester 2)
    Choose 15 credits from optional modules (note CSCM18 not available).
  - Year 2: CSCM59 (semester 1)
    CSCM10 (semester 1 + 2)
    Choose 15 credits from optional modules (note CSCM18 not available).
    CS-M20 (project, 60 credits).
- MSc in Advanced Computer Science:
  - Year 1: No compulsories Options: choose 90 credits of optional modules
  - Year 2: CSCM10 (semester 1 + 2)
    Choose 15 credits of optional modules
    CS-M20 (project, 60 credits).
- MSc in Advance Software Technology:
  - Year 1: CSCM01 (Semester 1)
    CSCM04 (Semester 1/2)
    CSCM64 (Semester 2
    Options: choose 45 credits of optional modules
  - Year 2: CSCM10 (semester 1 + 2)
    Choose 15 credits of optional modules
    CS-M20 (project, 60 credits).
- MSc in Data Science

– Year 1: CSCM70, CSCM27, CSCM35 (Semester 1)
  CSCM45 (Semester 2)
  Options: choose 30 credits of optional modules (note that the list of optional modules is restricted to those relevant to data science)
– Year 2: CSCM10 (semester 1 + 2)
  Choose q6 credits of optional modules (same restrictions as in year 1 apply)
  CS-M20 (project, 60 credits).

9 Restrictions for BSc graduates in Swansea

Graduates from our BScs in Computer Science in Swansea are usually not allowed to take modules of which they have already taken the level 3 version.

In case this leads to complications regarding the required modules for the AST modules, please contact the degree scheme coordinator who will decide how to resolve this.
The incompatibility applies to the following modules. We reserve the right to add other modules to this list of incompatibilities. Older modules which are no longer available might as well be incompatible. If a module has changed sufficiently, taking a module despite it being incompatible, might be acceptable, subject to approval by the lecturer.

- CSCM13/CSC313 Critical/High Integrity Systems
- CSCM18/CSC318 Cryptography and IT-Security/IT-Security: Theory and Practice
- CSCM37/CSC337 Data Visualization
- CSCM45/CSC345 Big Data and Machine Learning.
- CSCM48/CSC348 Web Application Development
- CSCM58/CS-358 (former module) High-Performance Computing in C/C++
- CSCM64/CSC364 Software Testing
- CSCM68/CSC368 Embedded Systems Design
- CSCM75/CSC375 Logic for Computer Science
- CSCM77/CS-377 (former module) Computer Vision and Pattern Recognition
- CSCM85/CSC385 Modelling and Verification Techniques

10 Module Selection

- For MSc in Computer Science our recommendation is to only take the compulsory modules, since they amount already to 60 credits of work. The optional modules should then be taken in Semester 2. Students wanting to take optional modules in Semester 1 should discuss this with the course coordinator Dr. Kullmann.
- For all other schemes and those wanting to take an optional module in Semester 1, we recommend to attend at least the first lecture of all modules in week 1 and then select the modules at the end of week 1 (deadline Friday 6 October, 2pm).
- We highly recommend to have a balanced module load between semester 1 and 2, ideally 60 credits in each semester.
  - For AST this would mean to have 2 optional modules (30 credits) in Semester 1 and the same amount in semester 2.
  - For ACS this cannot be achieved completely because of one module (CSCM10) being for semester $1 + 2$. Note that the main workload for CSCM10 is in Semester 2. The recommendation would be to have, in addition to CSCM10, 3 modules in one Semester and 4 modules in the other semester, with preference to 4 modules in Semester 1 and 3 modules in Semester 2.
  - In data science there is the same problem of CSCM10. The balanced options would be 1 module in Semester 1 and 2 modules in Semester 2, or no module in Semester 1 and 3 modules in Semester 2. Because the main workload of CSCM10 is in Semester the recommendation is to have one optional module in Semester 1 and 2 modules in semester 2. Please note that the list of optional modules for data science is more restricted, since only modules directly relevant for data science can be chosen.
  - Please make sure you use the module selection form for your degree scheme (CS vs AST vs ACS vs DS)

11 Module Catalogue

On the next pages you find the module catalogue with details as of 26 September 2017. In case of discrepancies with the official university catalogue, the university catalogue is considered as the definite source. Please note as well that there might be changes (including changes to the content of a module as to the assessment), as determined by the module coordinators.

11.1 Modules Semester 1

CSCM01 Software Engineering Project Planning and Management

Availability: Compulsory for AST; unavailable for CS, ACS, DS.

Semester: 1

Lecturer: Dr A. Denisova

Assessment: Report (40%), Report (60%)

Synopsis:
Software projects have long had a reputation for cost and time overruns - but they need not, and there are well-established, and emerging, techniques and processes to manage them well and effectively: for example, agile methodologies like SCRUM which are becoming a de-facto standard in the industry. Also, many projects have significant legal, social, ethical and professional consequences that a practitioner needs to be aware of and sensitive to. This module is about the process of successfully building complex software systems, and the implications, including on wider society, of doing so. It will also prepare students for their dissertation project by equipping them with the skills to successfully plan it, and to commence that planning process.

Syllabus:
Project planning and management principles:

- Timescales and dependencies
- Scoping and resources
- Risks: identifying, quantifying, managing, monitoring and mitigating
- Team management

Requirements and Specification

- Heavyweight and lightweight models

Methodologies for developing software:

- Traditional – waterfall, prototyping, spiral
- Rapid Application Development (RAD)
- Iterative and incremental
- Agile development
- Hybrid models – Scrum (agile/incremental)

Legal, Social, Ethical and Professional Issues

CSCM13 Critical Systems

Semester: 1

Availability: Optional for CS, ACS, AST; unavailable for DS.

Lecturer: Dr. A. Setzer

Assessment: 40% Practical assignments and case study; 60% Written Examination

Synopsis: This module introduces techniques for developing critical systems, especially safety critical systems. Students will gain experience in applying modern tools in the development of critical software.

Syllabus:
Introduction and Motivation:
What are high integrity and critical systems? Legal and ethical issues. Examples of major failures of high integrity systems. Successes and how/why they worked. Standards for safety-critical software and their shortcomings.

Analysis:
The hazard analysis process. Safety analysis and the safety case. Safety issues related to, but outside software. Human factors - the role of poor interfaces in software failures.

Specification and Verification:
Languages and tools for formal specification and verification of software. Detailed demonstration of one tool and its underlying theory.

Software Production:
Issues in program language selection to minimise failure. The software engineering process in the production of high-integrity software;

Correctness:
Validation and verification - the advantages and disadvantages of testing and formal verification.

CSCM27 Visual Analytics

Semester: 1

Availability: Compulsory for DS; optional for CS, ACS, and AST.

Lecturer: Dr DW Archambault

Remark: This module will require a significant level of competence in programming to complete satisfactorily

Assessment: Courswork 1,2 (15 % each); presentation (25 %); Report (45 %)

Synopsis: The course will provide an introduction to visual analytics and supporting necessary concepts in information visualisation. We will discuss the analytics process, data processing, and visualisation methods for many types of data. We will also cover supporting the human analytics process for this data.

Syllabus:

- History and goals of visual analytics.
- Types of data and encodings.
- Data processing and clustering.
- Information visualisation techniques.
- The analytics process and pipeline.

CSCM35 Big Data and Data Mining

Semester: 1

Availability: Compulsory for DS; optional for CS, ACS, and AST.

Lecturer: Dr AT Paiement

Remark: This module will require a significant level of competence in programming to complete satisfactorily
Number of places in this are restricted, priority is given to students of the MSc in Data Science.

Assessment: Assignment 1,2 (5 % each); Participation (10 %); Coursework 1 (20 %); Coursework 2 (60 %).

Synopsis: This course is an introductory course on data mining and its role in science and engineering. Data mining refers to the computational process of discovering patterns in large data sets. The main goal of the course is for students to gain practical data mining experience. The module is aimed at students with previous experience in programming and statistics, and preferably basic knowledge of the Python language.

Syllabus:

- Goals of data mining
- Data and challenges
- Methodology
- Data preparation
- Pattern analysis
- Python for data mining
- Big data requirements: efficiency and scalability
- Deep learning
- Hadoop

CSCM41 Programming in Java

Semester: 1

Availability: Compulsory for CS; unavailable for ACS, AST, and DS.

Lecturer: Dr O Kullmann

Assessment: Courswork 1 (20 %); Courswork 2 (20 %); Laboratory Work (10 %); Exam (50 %).

Synopsis: This intensive course provides a solid introduction to the Java programming language and development process.

Syllabus:

- Introduction to Programming. Introduction to Java.
- Basic structures of programming. Functions (static methods).
- The Object-Oriented paradigm: objects, classes and methods.
- Various I/O methods, and input/output via command-line.
- Basic correctness for programming (precision, finding and correcting errors).
- Aspects of exception handling and defensive programming.

CSCM48 Web Application Development

Semester: 1

Availability: Optional for CS, ACS, AST; unavailable for DS.

Lecturer: Dr. S. Walton

Remark: This module will require a significant level of competence in programming to complete satisfactorily

Assessment: 30% Individual 1000 word report planning the web application.
40% Implementation of the web application.
30% Individual 1000 word report evaluating the submitted web application.

Synopsis:
The module will develop the principles and technologies used for building web-based systems with particular reference to advanced techniques and principles using the .NET Framework and other technologies. Practical experience of building web systems will be gained via laboratories and coursework.

Syllabus:
The history of web application development.
HTML and CSS: Introduction and Good Practices.
Web Application Design.
Introduction to php.
Working with Visual Studio 2015
ASP.NET MVC 5
The C# programming language
Security and identity in web applications
Web development using Javascript
AJAX
ASP.NET Web API

CSCM53 Computer Systems Concepts

Semester: 1

Availability: Compulsory for CS; unavailable for ACS, AST, and DS.

Lecturer: Mrs MS Abuhmida

Assessment: Coursework 1,2 (10 % each); Exam (80 %)

Synopsis: This module gives an overview of some of the main principles underlying computers and computing from both a theoretical and an applied point of view. It includes a brief history of computers and software, an introduction to the representation of data and the basic components of a computer, the basic features of operating systems, file systems, computer networks, the world wide web, and some basic issues of computer security. A brief discussion on the use of formal methods in specifying computer systems is also given.

Syllabus:
Brief history of computers and software
Binary values and number systems
Data representation
Logic, gates and circuits
Computing components
Low level programming
Operating system concepts
File systems and directories
Computer networks
The world wide web
Overview of computer security
Introduction to formal methods

CSCM58 High-Performance Computing in C/C++

Semester: 1

Availability: Optional for CS, ACS, AST and DS.

Lecturer: Dr AT Paiement

Remark: This module will require a significant level of competence in programming to complete satisfactorily
Number of places in this are restricted, priority is given to students of the MSc in Data Science.

Assessment: 30 % Coursework 1, practical programming assignment; 65 % Coursework 2, practical programming assignment; 5 % Assignment 1, in class practical assignment

Synopsis:
This course is an introductory course on high-performance computing (HPC) and its role in science and engineering. High-performance computing refers to a specialized use and programming of supercomputers, computer clusters, and related architectures and software to speed up computations.
The main goal of the class is for students to gain practical HPC experience.
The module is aimed at students with previous experience in programming in a high-level programming language and preferably basic knowledge of the C/C++ language.

Syllabus:
Introduction to High Performance Computing
Fundamentals: architectures, parallel vs. distributed computing, types of parallelisms.
Parallel Application Design.
Building and Running Parallel Applications.
Distributed Memory Programming Using MPI
Computing Parallel Performances.
Practical applications development on HPCWales clusters.
HPC Network Topologies
Multi-threaded programming/ OpenMP.

CSCM59 Relational and Object-Oriented Database Systems

Semester: 1

Availability: Compulsory for CS; unavailable for ACS, AST, and DS.

Lecturer: Mr CJ Whyley

Assessment: Coursework (20 %); Exam (80 %)

Synopsis: This module gives an appreciation of of the complexity of real-world databases. It considers some of the problems that can occor in multi-user, multi-transactions situations. It discusses relational and object-oriented databases and covers their design and implementation. Distributed databases and databases linked to the web will also be discussed, as will data warehousing and data mining. Students will gain practical experience in designing and implementing a database.

Syllabus: A review of the nature of data and databases and an overview of database management and database system architecture.
Data models: relational databases, object databases.
Relational databases: the structure of the relational model, integrity constraints, relational algebra and calculus, normalisation.
Transaction management, data security and recovery, optimisation, distributed databases, concurrency control.
Object-oriented databases, type inheritance, active databases, temporal databases, logic-based databases.
Data warehouses and data mining, data visualisation.
Web technology and databases.

CSCM70 Mathematical Skills for Data Scientists

Semester: 1

Availability: Compulsory for DS; unavailable for CS, ACS, and AST.

Lecturer: Dr A Pauli

Assessment: Class Test 1 + 2 (30 %); Coureswork 1 (20 %); Laboratory Work (20 %)

Synopsis:
This course is an introductory course to the mathematical methods needed by a data scientist. It covers the basics of algebra, optimisation techniques, statistics, and Fourier analysis. The main goal of the class is for students to gain practical experience of the mathematical methods and tools that are essential in data science and that will be used in the other modules of this programme. The module is aimed at students with basic experience in mathematics.

Syllabus:
Vectors and matrices
Derivatives and partial derivatives
Variational calculus (fundamentals)
Gradient descent
Least Squares
Fundamentals of probability
Standard deviation, Variance and covariance
Bayesian Theorem
Eigenvalues and eigenvectors, PCA
Gaussian distribution, T-distribution
Cross correlation, Chi-square, mahalanobis distance
Fourier analysis

CSCM75 Logic in Computer Science

Semester: 1

Availability: Optional for CS, ACS, and AST; unavailable for DS.

Lecturer: Dr. U. Berger

Assessment: 10% Coursework 1 Syntax and semantics of propositional logic; 10 % Coursework 2: Predicate logic and automated proof search (Resolution); 10 % Lab: Formal proofs in natural deduction using an interactive proof tool; 70 % Written Examination

Synopsis: This module provides an in-depth introduction to logic and its applications to computer science, as a sound basis for the formal specification and verification of computer programs. Student will also learn how to use an interactive proof tool and carry out interactive proofs themselves.

Syllabus:

- Propositional logic (syntax, semantics, proof systems of natural deduction and resolution)
- Predicate logic (syntax, semantics, proof system)
- Applications of logic to program specification and verification
- Specialised logics e.g. for security protocols, reactive systems and credit card systems

CSCM77 Computer Vision and Pattern Recognition

Semester: 1

Availability: Optional for CS, ACS, AST, and DS.

Lecturer: Dr. X. Xie

Remark: This module will require a significant level of competence in programming to complete satisfactorily

Assessment: 10 % coursework 1; 20% coursework 2; 70% Written Examination

Synopsis:
This module introduces students to the important and modern topics and concepts of computer vision and pattern recognition, including image processing, segmentation, feature extraction, camera calibration, stereo vision, motion analysis, object tracking, recognition, data clustering, and dimensionality reduction. It teaches techniques that are used to understand and interpret the contents of images and videos and dissects state-of-the-art vision systems, such as Microsoft Kinect. Practical examples in C++ with OpenCV library and Matlab are provided throughout the lectures.

Syllabus: This course is composed of four parts: Introduction, Image Processing, Video Analysis and Pattern Recognition & Applications.
Introduction: The first part of the lectures gives an overview of Computer Vision and Pattern Recognition (CVPR) and a road show of this course. It also provides a brief revision of basic and important mathematical techniques frequently used in CVPR.
Image processing: filtering, registration, object extraction, shape recognition, segmentation, texture analysis.
Video analysis: camera models and calibration, stereo vision, depth estimation, motion estimation and tracking, local features for tracking.
Pattern Recognition: data clustering and K-means, Gaussian Mixture Modelling, dimensionality reduction, and applications.
The module is also accompanied with practical examples in both C++ (with OpenCV library) and Matlab.

CSCM98 Operating Systems and Architectures

Semester: 1

Availability: Optional for CS, ACS, AST, and DS.

Lecturer: Dr. B. Mora

Remark: This module will require a significant level of competence in programming to complete satisfactorily

Assessment: 30% Practical Assignment; 70% Written Examination

Synopsis:
This module gives an overview of current and future processor architectures, operating systems and basic concurrency problems. It intends to teach most details of the developing environment that must be taken into consideration when developing efficient software
Syllabus:
Operating Systems in general (Scheduler, Virtual Memory, Multi-tasking).

- Kernel calls.
- Resource management.
- Memory management.
- Paging and virtual memory.
- File Systems
- Processes and threads management

Architectures

- Registers+ALU
- Caches, cache lines and cache levels.
- Cache trashing.
- MMU
- TLB
- RAM Latency and throughput
- SIMD units
- SIMD Programming SSE,AVX, AVX-512
- Dedicated processor instructions.

Concurrency and issues

- Definition of core concepts including race conditions, deadlocks, starvation, critical sections.
- Standard concurrency problems and solutions
- Some standard techniques including software based locks, mutexes and semaphores, atomic instructions, barriers.

Distributed systems

- Distributed locks.
- Distributed file systems.
- Distributed clocks and time stamping.
- Cloud computing.
- Map/reduce algorithm.

11.2 Modules Semester 1 + 2

CSCM04 Software Team Project

Semester: 1 + 2

Availability: Compulsory for AST, unavailable for CS, ACS, and DS.

Lecturer: Dr T Owen

Assessment: Coursework 1 (30%), Coursework 2 (10 %), Coursework 3 (60%)

Synopsis:
The aim of this module is to provide students with the opportunity to apply their specialised knowledge to a realistic problem, and gain practical experience of the processes involved in the team-based production of software.

Syllabus:
Project planning, tools and techniques for planning.
Agile and traditional software development methodologies, including SCRUM.
Project conduct, time management, risk analysis management, and team working.
Application of legal, ethical, social and professional issues applicable to the computer industry, in the specific context of software development projects.

Students will specify, develop, test and document a substantial software system under the supervision of an academic staff member.

Significant emphasis will be placed on delivery - that is, meeting stated project goals

Somewhat less emphasis will be placed on ambitious technological solutions.

CSCM10 Computer Science Project Research Methods

Semester: 1 + 2

Availability: Compulsory for CS, ACS, AST, and DS.

Coordinator: Dr. AG Setzer

Lecturers: Dr A Denisova, Dr SC Lindsay, DR MJ Roach, Dr AG Setzer, Mr CJ Whyley, DR X Xie.
In addition tutorials and individual supervision by project supervisor

Assessment: Report surveying a research area of interest to you (35%), Presentation on Project and Initial Plan (15%), Project Specification and Design Document (50%).

Synopsis:
This module will introduce students to some fundamental research methodologies and good practice in research. They will undertake background research including a literature review and specify the aims of their MSc project.

Syllabus:
Seminars about selected scientific texts and research projects
Lectures on

- fundamental research methodologies
- good practice in research
- formulation of research questions and hypotheses
- logical reasoning
- literature research
- proper acknowledgement of sources
- principles of carrying out experimental research including ethical issues
- presentation of results

Individual guidance from project supervisors on

- identifying a research topic
- finding and reading related work
- report writing, citations and references
- using (digital) library services and search tools

CSCM94 Software Engineering Principles

Semester: 1 + 2

Availability: Compulsory for CS; unavailable for ACS, AST, and DS.

Lecturer: Dr T Owen

Assessment: Coursework (50 %); Exam (50 %)

Synopsis: Students will be introduced to the principles of software development and the main professional issues associated with its practice. They will also develop a significant piece of software in teams.

Syllabus: Introduction to the principles of professional software development.
Software process models (Waterfall etc).
Object-oriented software design (UML) and implementation
Review of legal, ethical, social, preliminary security, and professional issues applicable to the computer industry.
Software project management, planning, and risks
Software Testing.
Team work and time management.

11.3 Modules Semester 2

CSCM12 Software Concepts and Efficiency

Semester: 2

Availability: Compulsory for CS; unavailable for ACS, AST, and DS.

Lecturer: Dr KL Tam

Assessment: Coursework 1,2 (15 % each); Exam (70 %)

Synopsis:
This module provides a solid introduction to algorithm design, complexity analysis and data structure for efficient algorithm development. The module will involve students developing and evaluating their own algorithms. Java will be used as the main programming language.

Syllabus:
Introduction to the concept of algorithm and program efficiency.
Sorting and searching algorithms.
Top-down and bottom-up approach to algorithm development.
Recursion.
Complexity analysis, comparison and implementation of algorithms and various data structures.
The concept of data abstraction with particular reference to the object-oriented paradigm.
The abstract specification, implementation and complexity of various standard data types, which include array, linked list, stack, queue, tree, heap, graph and hashtable.
Importances of algorithms and data structures, and their applications (e.g. database - indexing, security - hashing and message digest).
The object-oriented language Java will be used for practical implementations.

CSCM18 IT-Security: Theory and Practice

Semester: 2

Availability: Optional for ACS and AST; unavailable for CS and DS.

Lecturer: Dr P James

Assessment: 10 % Coursework 1; 10 % Coursework 2; 10 % Laboratory work; 70 % Written Examination.
Synopsis:
The aim of this course is to examine theoretical and practical aspects of computer and network security.
Syllabus:
Security threats and their causes.
Security criteria and models.
Cryptography: including basic encryption, DES, AES, hash functions.
Access Control.
Security tools and frameworks: including IPSec, TLS, SSL, SSH and related tools.
Vulnerabilities and attacks: including port scanning, packet sniffing, SQL injection.
Security issues in wireless networks.
Security on the cloud.

CSCM37 Data Visualization

Semester: 2

Availability: Optional for CS, ACS, AST, and DS.

Lecturer: Dr. R. Laramee

Assessment: 50% Practical Coursework (15%, 20%, 15%); 50% Written Examination

Synopsis:
Data Visualization is concerned with the automatic or semi-automatic generation of digital images that depict data in a meaningful way(s). It is a relatively new field of computer science that is rapidly evolving and expanding. It is also very application oriented, i.e., real tools are built in order to help scientists from other disciplines.

Syllabus:
Introductory topics include: purposes and goals of visualisation, applications, challenges, the visualisation pipeline, sources of data: data dimensionality, data types, and grid types.
Information visualisation topics include: abstract data, hierarchical data, tree maps, cone trees, focus and context techniques, hyperbolic trees graphs and graph layouts, multi-dimensional data, scatter plots, scatter plot matrices, icons, parallel coordinates, interaction techniques, linking and brushing.
Volume visualisation topics include: slicing, surface vs. volume rendering, transfer functions, interpolation schemes, direct volume visualisation, ray casting, shear-warp factorisation, image order vs. object order algorithms, gradients, filtering, interpolation, iso surfacing.
Flow visualisation topics include: simulation, measured, and analytical data, steady and time-dependent (unsteady) flow, direct and indirect flow visualisation, applications, hedge hog plots, vector glyphs, numerical integration schemes, streamlines, streamline placement, geometric flow visualisation techniques, line integral convolution (LIC), texture-based techniques, feature-based flow visualisation.

CSCM39 Human Computer Interaction

Semester: 2

Availability: Optional for CS, ACS, AST, and DS.

Lecturer: Prof. H. Thimbleby

Assessment: 5 % Coursework 1; 45 % Coursework 2; 50 % Coursework 3.

Synopsis:
This module gives an overview on the main topics in Human Computer Interaction and helps students understand research and research processes in Human Computer Interaction. Students explore the advanced literature and research results underpinning the field of HCI. Classic papers and controversies are covered, as well as recent work from the leading figures. Students achieve a clear view of the 'cutting edge' and issues in the field and where things are happening. The module is very interactive, and students will be expected to give presentations.

Syllabus:
Advanced topics in Human Computer Interaction including:

- Interface design.
- Usability.
- Evaluation.
- Human factors.
- Human error.
- Cognitive science, and their role in the field.
- User models.
- User experience.
- Larger systems, ethics, design principles.

Research topics: Classic literature and personalities
Research Methods in HCI.

CSCM45 Big Data and Machine Learning

Semester: 2

Availability: Compulsory for DS; optional for CS, ACS, and AST.

Lecturer: Dr. X. Xie

Assessment: 20% Laboratory work; 20 % Coursework; 60% Written examination

Synopsis:
This module will discuss in-depth some of the most widely used and state-of-the-art artificical intelligence and machine learning techniques and their applications to big data problems. The students will gain both theoretical understanding of learning and practical know-how in applying those theories to real world problems. Topics include big data concept, data mining, learning theories, supervised and unsupervised learning, and reinforcement learning.

Syllabus:
This module covers three parts: introduction to big data and learning, data analysis techniques, and learning concepts and methods.
Introduction to big data and data mining;
Data clustering;
Dimensionality reduction: linear techniques;
Dimensionality reduction: nonlinear techniques;
Discriminative analysis;
Learning theory, including bias and variance theory, innovation process in machine learning;
Expert systems;
Unsupervised learning;
Supervised learning, including parametric and nonparametric methods, neural network, kernels, support vector machine, randomised decision trees;
Reinforcement and adaptive control;
Example applications to bioinformatics, health informatics, and web data processing.

CSCM64 Software Testing

Semester: 2

Availability: Compulsory for AST, optional for CS, ACS; unavailable for DS.

Lecturer: Professor M Roggenbach

Remark: This module will require a basic level of competence in programming to complete satisfactorily.
Awareness of propositional and predicate logic will be helpful for this module. Students should have a good understanding of programming and software architecture. Though there will be no programming required in this module, a number of the discussed testing approaches are based on program analysis.

Assessment: 10% Coursework 1; 10 % Coursework 2; 10 % Reflective Report; 10 % Laboratory work; 60% Written Examination

Synopsis:
Testing is the process of systematically experimenting with an object (the SUT = System Under Test) in order to establish its quality, where quality means the degree of accordance to the intention or specification. This module will provide an in-depth introduction to various test scenarios and enable students to gain hands-on experience by means of a number of practical exercises.

Syllabus:
The module provides a profound overview on industrially relevant methods in software testing and points out current research directions.

- Functional Testing: Boundary Value Testing, Equivalence Class Testing, Decision Table- Based Testing.
- Structural Testing: Path Testing, Data Flow Testing.
- Integration and System Testing: Levels of Testing, Approaches to Integration Testing.
- Object-Oriented Testing: Issues, Class Testing, Object-Oriented Integration Testing.
- Possibly selected Research Topics: e.g. Testing Hybrid Systems.

CSCM67 Graphics Processor Programming

Semester: 2

Availability: Optional for CS, ACS, AST, and DS.

Lecturer: Dr. M. Mora

Remark: This module will require significant level of competence in programming to complete satisfactorily, and content will be delivered in C and C++.

Assessment: Three practical graphics processor programming exercises (30%, 30%, 40%)

Synopsis:
This module introduces the different programming paradigms for Graphics processors to students. Programming of Graphics-oriented applications using OpenGL applications is investigated first. In the second part of the module, GP-GPU (General Purpose programming on the GPU) techniques with languages like CUDA or OpenCL are discussed. The programming concepts are put into practice through several lab classes using the C/C++ OpenCL and OpenGL libraries. Prior knowledge of programming concepts is required (for instance, pointers in C, data structures, classes, and basic sorting algorithms).

Syllabus:
3D evolution and Standard Graphics Hardware Pipeline.
The fundamentals of Rasterization, OpenGL and DirectX.
Graphics Hardware.

- Specific 3D architectures including NVidia, AMD and the new Intel Xeon PHI architecture will be detailed.

Graphics extensions and Shading languages.

- Vertex, Geometry and Fragments shaders as extensions to OpenGL.
- Introduction to DirectX.
- Introduction to OpenGL ES and WebGL.

Advanced Graphics techniques.

- Ambient Occlusions, Multi-pass rendering and Global Illumination on the GPU.

GP-GPU.

- The GPU as a co-processor. New technologies like OpenCL, NVidia CUDA for general purpose computing.

- Standard programming patterns and algorithms for GP_GPU (e.g., parallel reduction and prefix sum).
- GP_GPU case studies of high profile scientific problems.

Future of Graphics technology.

- Possible evolutions in the near future of Graphics technology.

Numerous practical labs to get familiar with the core programming concepts using OpenCL and OpenGL.

CSCM68 Embedded Systems Design

Semester: 2

Availability: Optional for CS, ACS, AST; unavailable for DS.

Lecturer: Prof. A. Beckmann and Dr A Pauli
Remark: Due to the lab, the number of places available for this module is limited. Places will be allocated during the first week of teaching; the allocation criteria will be announced in the first lecture.

This module will require a significant level of competence in programming to complete satisfactorily

Assessment: 10% Coursework 1; 40% Laboratory work; 50% Written Examination. Two lab exercises use Lego Mindstorms; each exercise has group component (building and demonstrating a robot) and individual lab book.

Synopsis:
Embedded systems are information processing systems embedded into enclosing products such as cars, telecommunication or fabrication equipment. They are essential for providing ubiquitous information, one of the key goals of modern information technology.
The aim of this module is to provide an overview of embedded system design, to relate the most important topics in embedded system design to each other, and to obtain an appreciation of the model based approach to embedded systems design.
The lab provides hands-on experience in the design of embedded systems, based on the Lego-Mindstorms kit.
Awareness of logical concepts (propositional logic, first order logic) will help the understanding of this module.

Syllabus:
The lectures discuss selected techniques in their specialisation to the design of embedded systems such as:

- Common characteristics, Requirements, Specification and Modelling
- Programming-language-level description techniques
- Hardware (Sensors, actuators, processors)
- Operating systems, middleware, scheduling
- Model driven design process
- Hardware/software partitioning and codesign
- Simulation, testing and verification techniques

The labs consist of a series of experiments using the Lego Mindstorms Kit that give the students hands-on experience in typical design challenges in embedded systems. Possible topics include examples from

- control theory
- real time systems
- discrete control
- fault tolerance
- distributed algorithms.

CSCM79 Hardware and Devices

. **Semester:** 2

Availability: Optional for CS, ACS, and AST; unavailable for DS.

Lecturer: Dr T Owen, Dr DR Sahoo

Remark: This module can only accommodate a limited students in the lab and the enrolment is on a first-come-first-enrolled basis. The precise number of places is defined in the 'Student Capacity' field of this catalogue entry.
Students who are enrolled in this module MUST have strong programming skills specially in OO programming.

Synopsis:
This module encourages students to explore the advanced literature and research results underpinning the field of interaction technologies and ubiquitous user-interface development. Students are expected to achieve a clear view of the cutting edge and issues in the field.

Syllabus:

- Ubiquitous computing and tangible user interfaces
- Interfacing with the real world using sensors and actuators with Arduino/Phidgets/Kinect
- Mobile phone sensing, e.g. camera/location/orientation
- Processing sensor data

CSCM85 Modelling and Verification Techniques

Semester: 2

Availability: Optional for CS, ACS, and AST; unavailable for DS.

Lecturer: Dr. U. Berger

Assessment: 30% Continuous Assessment; 70% Written Examination

Synopsis:
This module will give an overview of the landscape and the state of the art of current modelling and verification techniques. One particular tool for software verification will be studied in depth. Students will gain hands-on experience in using that tool.

Syllabus:
Overview of techniques for formal verification.
Interactive theorem proving, automated theorem proving and model checking.
Introduction to one specific logic for modelling and verification.
Techniques for modelling of software using verification tools.
Practical verification of software examples.

11.4 Summer Project

CS-M20 MSc Project (60 credits)

Semester: Summer

Availability: Compulsory for CS, ACS, AST, and DS.

Coordinator: Mr. C. Whyley

Assessment: Dissertation and Viva

Synopsis:
This module will provide students with the opportunity of exploring a particular topic in computer science in some considerable depth.

Syllabus:
The student will carry out independent project under the guidance of their supervisor.
The dissertation may include the following topics:

- Discussion of the subject area and its history;
- A literature survey;
- Formulation of scientific questions and the answers to them;
- Theoretical background;
- Description of the approach taken;
- Discussion of issues arising in the undertaking of the project;
- Evaluation of results;
- Progress and achievements of the project;
- Suggestions for further work.

12 Using the Online Timetable

- We recommend referring to the online timetable, the following timetables are only provided for convenience.
- By default it shows the timetable for the current week - especially during enrolment week it won't contain any entries for modules.
- You can switch to the next week to see the modules.
- If you want to see all MSc modules you can select modules from Computer Science, level M. You can choose this week, semester 1 or semester 2.
- Note that CSCM10 has two lectures (Monday and Friday at 9:00). All slots if CSCM10 taking place in the Digital Technium, and you need to attend only one slot biweekly. You will be allocated a tutorial slot which doesn't clash with your lectures.
- Some of the optional modules for MSc in Computer Science, which are marked as requiring substantial programming background may result in timetable clashes. If you are affected please contact the lecturer of that module whether it is possible to find an arrangements to avoid that clash.
- The list of modules for each of the 4 courses contains now a list of modules not be taken. If you select all level M modules, you need to ignore the modules from that list.

13 Timetable (Semester 1)

- For your convenience the timetable as of 28 Sept 2017 will be provided below for first orientation. Note that the timetable will change, especially during the first week of teaching.
- Please check on `https://science.swansea.ac.uk/intranet/attendance/timetable` for your **personalised timetable** and **updates.**
- Some lecturers will inform you about cancelled events during their lectures.
- Tutorial allocation will be carried out later, you will need to attend only the tutorial group allocated to you – tutorial slots have been marked as CSCM10 (tutorial).
- **Note on weeks:** week 1 is enrolment week, week 2 is first week of teaching etc. Modules announced for weeks 3 - 12 will not take place in the first week of teaching.

"Changes may still be made to the timetable, so please check regularly online."

Timetable MSc (All Modules), Semester 1, 2017/18

	Monday	Tuesday	Wednesday	Thursday	Friday
9:00	**CSCM10** AD, SCL, MJR, AGS, CJW, XX Faraday M	**CSCM58** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM48** SPW Fulton Lecture Rm A Weeks: 2-7	**CSCM58** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM10** AD, SCL, MJR, AGS, CJW, XX Faraday K Weeks: 3-7
10:00	**CSCM35** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM58** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab) **CSCM59** CJW Faraday Rm F	**CSCM35** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab) **CSCM48** SPW Talbot PC Lab 043 Weeks: 4-7	**CSCM58** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab) **CSCM01** AD Talbot Rm 38	**CSCM77** XX Vivian 731/2
11:00	**CSCM35** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab) **CSCM53** MSMA Faraday Rm F	**CSCM27** DWA 2 hours Faraday C Weeks: 3-12 **CSCM53** MSMA Faraday Rm D	**CSCM35** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab) **CSCM53** MSMA Faraday Rm F	**CSCM75** UB 2 hours Faraday Rm D	**CSCM01** AD Talbot Rm 38

	Monday	Tuesday	Wednesday	Thursday	Friday
12:00	**CSCM13** AGS 2 hours Glyndwr Rm B	**CSCM27** DWA 2 hours Faraday C Weeks: 3-12	**CSCM10 (tutorial group 6)** (TBC) Digi Tech Rm 213 Weeks: 4, 6, 8, 10, 12, 21	**CSCM75** UB 2 hours Faraday Rm D	**CSCM98** BM 2 hours Wallace Rm 219B (PC Lab)
		CSCM59 CJW Faraday Rm F	**CSCM10 (tutorial group 7)** (TBC) Digi Tech Rm 213 Weeks: 3, 5, 7, 9, 11, 20		
			CSCM48 SPW Faraday M Weeks: 2-7		
			CSCM70 MS Digi Tech Rm 212		
13:00	**CSCM13** AGS 2 hours Glyndwr Rm B	**CSCM59** CJW Wallace Rm 217 (PC Lab)		**CSCM77** XX Glyndwr Rm A	**CSCM98** BM 2 hours Wallace Rm 219B (PC Lab)
	CSCM70 MS 2 hours Glyndwr Rm K	**CSCM77** XX Glyndwr Rm B			
14:00	**CSCM70** MS 2 hours Glyndwr Rm K	**CSCM04** TO Keir Hardie 021 Weeks: 2, 4, 6, 8, 10, 12		**CSCM98** BM 2 hours Glyndwr Rm M	**CSCM41** OK 2 hours Faraday Tower Rm 500 (Linux Lab) Weeks: 3-12
	CSCM10 (tutorial group 1) (TBC) Digi Tech Rm 211 Weeks: 4, 6, 8, 10, 12, 21	**Computer Science Research Seminar** Talbot 909			
	CSCM10 (tutorial group 2) (TBC) Digi Tech Rm 212 Weeks: 3, 5, 7, 9, 11, 20				
	CSCM94 TO Faraday Rm F Weeks: 2-12, 20-26, 30-33				

	Monday	Tuesday	Wednesday	Thursday	Friday
15:00	**CSCM27** DWA Faraday Rm B Weeks: 3-12 **CSCM41** OK Faraday Rm F	**CSCM75** UB Faraday Rm D		**CSCM98** BM 2 hours Glyndwr Rm M **CSCM10 (tutorial group 8)** (TBC) Digi Tech Rm 215 Weeks: 3, 5, 7, 9, 11, 20	**CSCM41** OK 2 hours Faraday Tower Rm 500 (Linux Lab) Weeks: 3-12
16:00	**CSCM10 (tutorial group 3)** (TBC) Digi Tech Rm 215 Weeks: 3, 5, 7, 9, 11, 20 **CSCM10 (tutorial group 4)** (TBC) Digi Tech Rm 215 Weeks: 4, 6, 8, 10, 12, 21 **CSCM41** OK 2 hours Faraday Tower Rm 500 (Linux Lab) Weeks: 3-12 **CSCM48** SPW 2 hours Talbot PC Lab 043 Weeks: 8-12	**CSCM48** SPW 2 hours Talbot PC Lab 043 Weeks: 8-12		**CSCM13** AGS 2 hours Faraday Tower Rm 500 (Linux Lab) Weeks: 3-7	**CSCM41** OK Faraday Tower Rm 500 (Linux Lab) Weeks: 3-12

	Monday	Tuesday	Wednesday	Thursday	Friday
17:00	**CSCM41** OK 2 hours Faraday Tower Rm 500 (Linux Lab) Weeks: 3-12	**CSCM48** SPW 2 hours Talbot PC Lab 043 Weeks: 8-12		**CSCM13** AGS 2 hours Faraday Tower Rm 500 (Linux Lab) Weeks: 3-7	
	CSCM48 SPW 2 hours Talbot PC Lab 043 Weeks: 8-12	**CSCM48** SPW Talbot PC Lab 043 Weeks: 4-7			
	CSCM10 (tutorial group 5) (TBC) Digi Tech Rm 215 Weeks: 4, 6, 8, 10, 12, 21				
	CSCM48 SPW Faraday M Weeks: 3-7				

"Changes may still be made to the timetable, so please check regularly online."

MSc Computer Science, Compulsory Modules, Semester 1, 2017/18

	Monday	Tuesday	Wednesday	Thursday	Friday
9:00	**CSCM10** AD, SCL, MJR, AGS, CJW, XX Faraday M		Add activity...		**CSCM10** AD, SCL, MJR, AGS, CJW, XX Faraday K Weeks: 3-7
10:00		**CSCM59** CJW Faraday Rm F			
11:00	**CSCM53** MSMA Faraday Rm F	**CSCM53** MSMA Faraday Rm D	**CSCM53** MSMA Faraday Rm F		
12:00		**CSCM59** CJW Faraday Rm F	**CSCM10 (tutorial group 6)** (TBC) Digi Tech Rm 213 Weeks: 4, 6, 8, 10, 12, 21 **CSCM10 (tutorial group 7)** (TBC) Digi Tech Rm 213 Weeks: 3, 5, 7, 9, 11, 20		
13:00		**CSCM59** CJW Wallace Rm 217 (PC Lab)			

	Monday	Tuesday	Wednesday	Thursday	Friday
14:00	**CSCM10 (tutorial group 1)** (TBC) Digi Tech Rm 211 Weeks: 4, 6, 8, 10, 12, 21 **CSCM10 (tutorial group 2)** (TBC) Digi Tech Rm 212 Weeks: 3, 5, 7, 9, 11, 20	**Computer Science Research Seminar** Talbot 909			**CSCM41** OK 2 hours Faraday Tower Rm 500 (Linux Lab) Weeks: 3-12
15:00	**CSCM41** OK Faraday Rm F			**CSCM10 (tutorial group 8)** (TBC) Digi Tech Rm 215 Weeks: 3, 5, 7, 9, 11, 20	**CSCM41** OK 2 hours Faraday Tower Rm 500 (Linux Lab) Weeks: 3-12
16:00	**CSCM10 (tutorial group 3)** (TBC) Digi Tech Rm 215 Weeks: 3, 5, 7, 9, 11, 20 **CSCM10 (tutorial group 4)** (TBC) Digi Tech Rm 215 Weeks: 4, 6, 8, 10, 12, 21 **CSCM41** OK 2 hours Faraday Tower Rm 500 (Linux Lab) Weeks: 3-12				**CSCM41** OK Faraday Tower Rm 500 (Linux Lab) Weeks: 3-12
17:00	**CSCM41** OK 2 hours Faraday Tower Rm 500 (Linux Lab) Weeks: 3-12 **CSCM10 (tutorial group 5)** (TBC) Digi Tech Rm 215 Weeks: 4, 6, 8, 10, 12, 21				

"Changes may still be made to the timetable, so please check regularly online."

Timetable MSc Adv Computer Science, Semester 1, 2017/18

	Monday	Tuesday	Wednesday	Thursday	Friday
9:00	**CSCM10** AD, SCL, MJR, AGS, CJW, XX Faraday M	**CSCM58** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM48** SPW Fulton Lecture Rm A Weeks: 2-7	**CSCM58** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM10** AD, SCL, MJR, AGS, CJW, XX Faraday K Weeks: 3-7
10:00	**CSCM35** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM58** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM35** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab) **CSCM48** SPW Talbot PC Lab 043 Weeks: 4-7	**CSCM58** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM77** XX Vivian 731/2
11:00	**CSCM35** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM27** DWA 2 hours Faraday C Weeks: 3-12	**CSCM35** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM75** UB 2 hours Faraday Rm D	
12:00	**CSCM13** AGS 2 hours Glyndwr Rm B	**CSCM27** DWA 2 hours Faraday C Weeks: 3-12	**CSCM10 (tutorial group 6)** (TBC) Digi Tech Rm 213 Weeks: 4, 6, 8, 10, 12, 21 **CSCM10 (tutorial group 7)** (TBC) Digi Tech Rm 213 Weeks: 3, 5, 7, 9, 11, 20 **CSCM48** SPW Faraday M Weeks: 2-7	**CSCM75** UB 2 hours Faraday Rm D	**CSCM98** BM 2 hours Wallace Rm 219B (PC Lab)

	Monday	Tuesday	Wednesday	Thursday	Friday
13:00	**CSCM13** AGS 2 hours Glyndwr Rm B	**CSCM77** XX Glyndwr Rm B		**CSCM77** XX Glyndwr Rm A	**CSCM98** BM 2 hours Wallace Rm 219B (PC Lab)
14:00	**CSCM10 (tutorial group 1)** (TBC) Digi Tech Rm 211 Weeks: 4, 6, 8, 10, 12, 21 **CSCM10 (tutorial group 2)** (TBC) Digi Tech Rm 212 Weeks: 3, 5, 7, 9, 11, 20	**Computer Science Research Seminar** Talbot 909		**CSCM98** BM 2 hours Glyndwr Rm M	Add activity...
15:00	**CSCM27** DWA Faraday Rm B Weeks: 3-12	**CSCM75** UB Faraday Rm D		**CSCM98** BM 2 hours Glyndwr Rm M **CSCM10 (tutorial group 8)** (TBC) Digi Tech Rm 215 Weeks: 3, 5, 7, 9, 11, 20	
16:00	**CSCM10 (tutorial group 3)** (TBC) Digi Tech Rm 215 Weeks: 3, 5, 7, 9, 11, 20 **CSCM10 (tutorial group 4)** (TBC) Digi Tech Rm 215 Weeks: 4, 6, 8, 10, 12, 21 **CSCM48** SPW 2 hours Talbot PC Lab 043 Weeks: 8-12	**CSCM48** SPW 2 hours Talbot PC Lab 043 Weeks: 8-12		**CSCM13** AGS 2 hours Faraday Tower Rm 500 (Linux Lab) Weeks: 3-7	

	Monday	Tuesday	Wednesday	Thursday	Friday
17:00	**CSCM48** SPW 2 hours Talbot PC Lab 043 Weeks: 8-12	**CSCM48** SPW 2 hours Talbot PC Lab 043 Weeks: 8-12		**CSCM13** AGS 2 hours Faraday Tower Rm 500 (Linux Lab) Weeks: 3-7	
	CSCM10 (tutorial group 5) (TBC) Digi Tech Rm 215 Weeks: 4, 6, 8, 10, 12, 21	**CSCM48** SPW Talbot PC Lab 043 Weeks: 4-7			
	CSCM48 SPW Faraday M Weeks: 3-7				

This page is intentionally left blank.

"Changes may still be made to the timetable, so please check regularly online."

Timetable MSc Adv. Software Technology, Semester 1, 2017/18

	Monday	Tuesday	Wednesday	Thursday	Friday
9:00	**CSCM10** AD, SCL, MJR, AGS, CJW, XX Faraday M	**CSCM58** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM48** SPW Fulton Lecture Rm A Weeks: 2-7	**CSCM58** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM10** AD, SCL, MJR, AGS, CJW, XX Faraday K Weeks: 3-7
10:00	**CSCM35** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM58** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM35** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab) **CSCM48** SPW Talbot PC Lab 043 Weeks: 4-7	**CSCM58** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab) **CSCM01** AD Talbot Rm 38	**CSCM77** XX Vivian 731/2
11:00	**CSCM35** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM27** DWA 2 hours Faraday C Weeks: 3-12	**CSCM35** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM75** UB 2 hours Faraday Rm D	**CSCM01** AD Talbot Rm 38
12:00	**CSCM13** AGS 2 hours Glyndwr Rm B	**CSCM27** DWA 2 hours Faraday C Weeks: 3-12	**CSCM10 (tutorial group 6)** (TBC) Digi Tech Rm 213 Weeks: 4, 6, 8, 10, 12, 21 **CSCM10 (tutorial group 7)** (TBC) Digi Tech Rm 213 Weeks: 3, 5, 7, 9, 11, 20 **CSCM48** SPW Faraday M Weeks: 2-7	**CSCM75** UB 2 hours Faraday Rm D	**CSCM98** BM 2 hours Wallace Rm 219B (PC Lab)

	Monday	Tuesday	Wednesday	Thursday	Friday
13:00	**CSCM13** AGS 2 hours Glyndwr Rm B	**CSCM77** XX Glyndwr Rm B		**CSCM77** XX Glyndwr Rm A	**CSCM98** BM 2 hours Wallace Rm 219B (PC Lab)
14:00	**CSCM10 (tutorial group 1)** (TBC) Digi Tech Rm 211 Weeks: 4, 6, 8, 10, 12, 21 **CSCM10 (tutorial group 2)** (TBC) Digi Tech Rm 212 Weeks: 3, 5, 7, 9, 11, 20	**CSCM04** TO Keir Hardie 021 Weeks: 2, 4, 6, 8, 10, 12 **Computer Science Research Seminar** Talbot 909		**CSCM98** BM 2 hours Glyndwr Rm M	Add activity...
15:00	**CSCM27** DWA Faraday Rm B Weeks: 3-12	**CSCM75** UB Faraday Rm D		**CSCM98** BM 2 hours Glyndwr Rm M **CSCM10 (tutorial group 8)** (TBC) Digi Tech Rm 215 Weeks: 3, 5, 7, 9, 11, 20	
16:00	**CSCM10 (tutorial group 3)** (TBC) Digi Tech Rm 215 Weeks: 3, 5, 7, 9, 11, 20 **CSCM10 (tutorial group 4)** (TBC) Digi Tech Rm 215 Weeks: 4, 6, 8, 10, 12, 21 **CSCM48** SPW 2 hours Talbot PC Lab 043 Weeks: 8-12	**CSCM48** SPW 2 hours Talbot PC Lab 043 Weeks: 8-12		**CSCM13** AGS 2 hours Faraday Tower Rm 500 (Linux Lab) Weeks: 3-7	

	Monday	Tuesday	Wednesday	Thursday	Friday
17:00	**CSCM48** SPW 2 hours Talbot PC Lab 043 Weeks: 8-12	**CSCM48** SPW 2 hours Talbot PC Lab 043 Weeks: 8-12		**CSCM13** AGS 2 hours Faraday Tower Rm 500 (Linux Lab) Weeks: 3-7	
	CSCM10 (tutorial group 5) (TBC) Digi Tech Rm 215 Weeks: 4, 6, 8, 10, 12, 21	**CSCM48** SPW Talbot PC Lab 043 Weeks: 4-7			
	CSCM48 SPW Faraday M Weeks: 3-7				

This page is intentionally left blank.

"Changes may still be made to the timetable, so please check regularly online."

Timetable MSc in Data Science, Semester 1, 2017/18

	Monday	Tuesday	Wednesday	Thursday	Friday
9:00	**CSCM10** AD, SCL, MJR, AGS, CJW, XX Faraday M	**CSCM58** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	Add activity...	**CSCM58** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM10** AD, SCL, MJR, AGS, CJW, XX Faraday K Weeks: 3-7
10:00	**CSCM35** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM58** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM35** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM58** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM77** XX Vivian 731/2
11:00	**CSCM35** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)	**CSCM27** DWA 2 hours Faraday C Weeks: 3-12	**CSCM35** ATMP 2 hours Faraday Tower Rm 500 (Linux Lab)		
12:00		**CSCM27** DWA 2 hours Faraday C Weeks: 3-12	**CSCM10 (tutorial group 6)** (TBC) Digi Tech Rm 213 Weeks: 4, 6, 8, 10, 12, 21 **CSCM10 (tutorial group 7)** (TBC) Digi Tech Rm 213 Weeks: 3, 5, 7, 9, 11, 20 **CSCM70** MS Digi Tech Rm 212		**CSCM98** BM 2 hours Wallace Rm 219B (PC Lab)
13:00	**CSCM70** MS 2 hours Glyndwr Rm K	**CSCM77** XX Glyndwr Rm B		**CSCM77** XX Glyndwr Rm A	**CSCM98** BM 2 hours Wallace Rm 219B (PC Lab)

	Monday	Tuesday	Wednesday	Thursday	Friday
14:00	**CSCM70** MS 2 hours Glyndwr Rm K	**CSCM04** TO Keir Hardie 021 Weeks: 2, 4, 6, 8, 10, 12		**CSCM98** BM 2 hours Glyndwr Rm M	Add activity...
	CSCM10 (tutorial group 1) (TBC) Digi Tech Rm 211 Weeks: 4, 6, 8, 10, 12, 21	**Computer Science Research Seminar** Talbot 909			
	CSCM10 (tutorial group 2) (TBC) Digi Tech Rm 212 Weeks: 3, 5, 7, 9, 11, 20				
15:00	**CSCM27** DWA Faraday Rm B Weeks: 3-12			**CSCM98** BM 2 hours Glyndwr Rm M	
				CSCM10 (tutorial group 8) (TBC) Digi Tech Rm 215 Weeks: 3, 5, 7, 9, 11, 20	
16:00	**CSCM10 (tutorial group 3)** (TBC) Digi Tech Rm 215 Weeks: 3, 5, 7, 9, 11, 20				
	CSCM10 (tutorial group 4) (TBC) Digi Tech Rm 215 Weeks: 4, 6, 8, 10, 12, 21				
17:00	**CSCM10 (tutorial group 5)** (TBC) Digi Tech Rm 215 Weeks: 4, 6, 8, 10, 12, 21				