Programme Information & PLOs

Title of the new programme – including any year abroad/ in industry variants

MMath in Mathematics and Computer Science

Level of qualification

Please select:	Level 7	

Please indicate if the programme is offered with any year abroad / in industry variants	**Year in Industry** Please select Y/N	Yes
	Year Abroad Please select Y/N	Yes

Department(s):
Where more than one department is involved, indicate the lead department

Lead Department	Computer Science
Other contributing Departments:	Mathematics

Programme leader

Dr Detlef Plump

Purpose and learning outcomes of the programme

Statement of purpose for applicants to the programme

Computer Science and Mathematics are naturally congruent, and the MMath combined degree programme provides pathways to study each subject at masters level, with opportunities to specialise within and across both disciplines, allowing you to acquire a spectrum of expertise and range of skills that are not only in demand by key employers but also bring you to the gateway into current research.

The programme will provide you with a solid foundation in the principles and practices of computer science, including coding and basic engineering, leading to advanced training in focussed areas of your choice. You will also develop your core mathematical skills, particularly in calculus and algebra, and your reasoning and analytical prowess will be sharpened as you are guided to use mathematics in deeper and more interesting ways, some of which are directly related to computation. You will experience didactic teaching delivered by computer scientists and mathematicians at the forefront of their areas of expertise, and collaborative learning through group work and team-based problem solving. The final two years of the programme enable you to tailor your learning from a wide range of optional modules, of which those in the final year will bring you to the frontiers of research and state of the art methodology, culminating with a substantial and significant independent investigation in an area of your choosing, which may be in Computer Science or Mathematics.

The variant with a year in industry provides a placement after completing two full years at university, which presents you with opportunities to apply your expertise in a commercial environment, increasing your awareness of the power and potential of applied computer science, and providing you with a keener sense of the directions you want to pursue when you return to complete the final two years of the programme.

Programme Learning Outcomes

Please provide six to eight statements of what a graduate of the programme can be expected to do.

Taken together, these outcomes should capture the distinctive features of the programme. They should also be outcomes for which progressive achievement through the course of the programme can be articulated, and which will therefore be reflected in the design of the whole programme.

PLO	On successful completion of the programme, graduates will be able to:
1	Confidently and competently apply computational and mathematical thinking to problems, using skills in problem analysis, representation and abstraction, and the application of standard and higher level mathematical and computational techniques, including the theory and practice of programming and software engineering. [Computational and mathematical competence]
2	Critically analyse statements, arguments or conjectures that underpin the theory of Mathematics and Computer Science, justifying the principles chosen for such critiques, and developing their own lines of well-founded reasoning. [Computational and mathematical reasoning]
3	Adapt to new and unfamiliar challenges in Computer Science and Mathematics, recognising appropriate ideas and approaches drawn from a range of technologies, languages, paradigms, models and mathematical theories, and informed by current research and scholarship. [Adaptability]
4	Conduct an independent investigation into a specialised area of Mathematics or Computer Science, at a level which engages with current research or cutting edge developments, by gathering material from a variety of sources, and synthesising this material into a well-organised and coherent account, or effective solution to a user-specified need or commercial imperative. [Independence]

5	Work effectively in a team, formulating and fulfilling obligations towards achieving goals by managing workloads, setting and meeting deadlines, and optimising resources , and taking leadership and responsibility for aspects of the work planned. [Team work]
6	Communicate complex ideas in Computer Science and Mathematics in a clear, unambiguous and organised manner, at a level appropriate for the intended recipients, and also present an effective summary of these ideas for an expert audience. [Communication]
7	Appreciate the wider context of Mathematics and Computer Science and their component disciplines, understand how these can contribute to and impact on society, develop an awareness of key legal and ethical issues, and operate as responsible professionals. [Awareness and professionalism]

Programme Learning Outcome for year in industry (where applicable) For programmes which lead to the title 'with a Year in Industry' – typically involving an additional year – please provide either a) amended versions of some (at least one, but not necessarily all) of the standard PLOs listed above, showing how these are changed and enhanced by the additional year in industry b) an additional PLO, if and only if it is not possible to capture a key ability developed by the year in industry by alteration of the standard PLOs.
Apply acquired computational and mathematical expertise in the modern workplace, and understand, experience and appreciate the potential of such ideas and skills in an industrial context, with an awareness of relevant legislation, procedures and commercial imperatives. [Year in industry]
Programme Learning Outcome for year abroad programmes (where applicable) For programmes which lead to the title 'with a Year Abroad' – typically involving an additional year – please provide either a) amended versions of some (at least one, but not necessarily all) of the standard PLOs listed above, showing how these are changed and enhanced by the additional year abroad or b) an additional PLO, if and only if it is not possible to capture a key ability developed by the year abroad by alteration of the standard PLOs.
Explanation of the choice of Programme Learning Outcomes Please explain your rationale for choosing these PLOs in a statement that can be used for students (such as in a student handbook). Please include brief reference to:
i) Why the PLOs are considered ambitious or stretching?
With the possible exception of PLO4, all PLOs are partially fulfilled in each Stage of the programme. The distinctly hierarchical nature of the Stages requires students to have mastered the knowledge and skills of one Stage before progressing to the next, allowing the PLOs to be addressed at deeper levels as the programme progresses. PLO4 requires a panoramic view of the programme, and places the responsibility for learning firmly in the hands of the student.

ii) The ways in which these outcomes are distinctive or particularly advantageous to the student:
Each outcome is distinctive and advantageous for students, as follows: computational and mathematical competence and problem solving (PLO1); computational and mathematical reasoning (PLO2); adaptability (PLO3); independence (PLO4); team work (PLO5); communication (PLO6); awareness and professionalism (PLO7); industrial experience (PLOYI).
iii) How the programme learning outcomes develop students' digital literacy and will make appropriate use of technology-enhanced learning (such as lecture recordings, online resources, simulations, online assessment, 'flipped classrooms' etc)?
Bearing in mind the title of the programme, the PLOs necessarily develop digital literacy at the highest level, including programming skills. The PLOs also necessitate use of appropriate software for writing technical documents (eg LaTeX) and delivering presentations, and in some cases require the use of specialised mathematical software (eg Maple, Matlab). More routine use of digital technology certainly includes lecture recordings (currently audio, maybe video in future), interaction with university and departmental VLEs to access bespoke online resources, elements of online assessment (for some, but not all modules).
iv) How the PLOs support and enhance the students' employability (for example, opportunities for students to apply their learning in a real world setting)? The programme's employablity objectives should be informed by the University's Employability Strategy:
http://www.york.ac.uk/about/departments/support-and-admin/careers/staff/
Each PLO provides a distinct area of expertise, listed in ii) above, which when combined provide students with a skill set that is highly relevant to a wide range of careers and valued by key employers. In particular, PLO7 directly addresses responsible practice, and PLOYI provides direct experience of the commercial and industrial workplace, and in many cases provides a direct link to future employment.
vi) How will students who need additional support for academic and transferable skills be identified and supported by the Department?
Students are assessed for disability and special needs via the university's Disability Support Unit, and appropriate support provided centrally. Small group support teaching is provided to all students on a module by module basis, allowing individual students to access the academic support they require. Personal supervision is provided to each student by one of the Departments.
vii) How is teaching informed and led by research in the department/ centre/ University?
All didactic teaching is delivered by computer scientists and mathematicians who work at the forefront of their areas of expertise, and are therefore engaged in and aware of current research, cutting edge methodology etc. Many modules in the final two years of the programme are designed around the research interests of staff (for example, quantum computation, cryptography, computer vision, formal languages and automata, and semigroup theory). Modules in the final year are categorised as masters level, and bring students to the gateway of research.
Stage-level progression Please complete the table below, to summarise students' progressive development towards the achievement of PLOs, in terms of the characteristics that you expect students to demonstrate at the end of each year. This summary may be particularly helpful to students and the programme team where there is a high proportion of option modules. Note: it is not expected that a position statement is written for each PLO, but this can be done if preferred (please add information in the 'individual statement' boxes). For a statement that applies across all PLOs in the stage fill in the 'Global statement' box.
Stage 0 (if your programme has a Foundation year, use the toggles to the left to show the hidden rows)

Stage 1							
On progression from the first year (Stage 1), students will be able to:				*Understand and apply the mathematical principles underlying computing.* *Understand and apply the principles of mathematical logic and reasoning in the context of proving elementary theorems.* *Understand the foundations of systems architecture and programming as used in computer systems.* *Competently apply foundational computational and mathematical techniques and thinking to unfamiliar but straightforward problems.* *Work as an individual and in a team.* *Produce short reports and presentations that communicate elementary mathematical and computational ideas clearly and concisely.*			
PLO 1	PLO 2	PLO 3	PLO 4	PLO 5	PLO 6	PLO 7	PLO 8
Individual statements							
Stage 2							
On progression from the second year (Stage 2), students will be able to:				*Apply mathematical logic and reasoning to the proof of more advanced and sophisticated theorems in the context of developing well-defined mathematical theories.* *Competently apply more sophisticated computational and mathematical thinking to larger problems that require the comparison of techniques and paradigms from across a broad range and selection of those that are most appropriate.* *Understand engineering tradeoffs in computer system development.* *Work effectively both independently and in teams.* *Communicate clearly and concisely with a variety of audiences in a range of formats.*			
PLO 1	PLO 2	PLO 3	PLO 4	PLO 5	PLO 6	PLO 7	PLO 8
Individual statements							
Stage 3							
(For Integrated Masters) On progression from the third year (Stage 3), students will be able to:				*Adapt to new high-level mathematical concepts and applications, and new computing technologies and languages, by transferring understanding of previously-studied mathematical and computational principles.* *Use specialised knowledge from a variety of option modules in computer science and mathematics to deconstruct and solve mathematical problems and engineer solutions to problems in which computation forms a significant part.*			

PLO 1	PLO 2	PLO 3	PLO 4	PLO 5	PLO 6	PLO 7	PLO 8
Individual statements							

Programme Structure

Module Structure and Summative Assessment Map
Please complete the summary table below which shows the module structure and the pattern of summative assessment through the programme.

‘Option module’ can be used in place of a specific named option. If the programme requires students to select option modules from specific lists these lists should be provided in the next section.

From the drop-down select 'S' to indicate the start of the module, 'A' to indicate the timing of each distinct summative assessment point (eg. essay submission/ exam), and 'E' to indicate the end of the module (if the end of the module coincides with the summative assessment select 'EA') . It is not expected that each summative task will be listed where an overall module might be assessed cumulatively (for example weekly problem sheets).

If summative assessment by exams will be scheduled in the summer Common Assessment period (weeks 5-7) a single ‘A’ can be used within the shaded cells as it is understood that you will not know in which week of the CAP the examination will take place.

Stage 0 (if you have modules for Stage 0, use the toggles to the left to show the hidden rows)

Stage 1

Credits	Module		Autumn Term										Spring Term										Summer Term									
	Code	Title	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10	1	2	3	4	5	6	7	8	9	10
30	MAT00001C	Calculus		S									A														EA					
20	MAT00010C	Algebra		S									A														EA					
10	MAT00011C	Mathematical Skills 1: Reasoning and Communication		S									A									EA		A								
15	COM00001C	Introduction to Computer Architecture (ICAR)		S										A												E			A			
20	COM00005C	Mathematical Foundations of Computer Science (MFCS)		S									A													E			A			
5	COM00008C	Skills, Knowledge and Independent Learning (SKIL)	S								A									E	A											

20	COM00007C	Theory and Practice of Programming (TPOP)		S																						E	A		A			
Stage 2																																
Credits	**Module**		**Autumn Term**										**Spring Term**										**Summer Term**									
	Code	**Title**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
40	MAT00032I	Pure Mathematics		S									A														EA					
10	MAT00026I	Linear Algebra		S								E	A																			
10	MAT00030I	Vector Calculus		S								E	A																			
10	COM00002I	Computability and Complexity (COCO)											S									E					A					
20	COM00005I	Principles of Programming Languages (POPL)	S																	E		A					A					
20	COM00001I	Artificial Intelligence (ARIN)											S												E		A					
or																																
20	COM00014I	Systems (SYST)											S												E		A					
10	COM00013I	Implementation of Programming Languages (IMPL)	S									E	A																			
or																																
10	COM00009I	Vision and Graphics (VIGR)	S									E	A																			
Stage 3																																
Credits	**Module**		**Autumn Term**										**Spring Term**										**Summer Term**									
	Code	**Title**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
10		Autumn (Maths), List A		S								E	A																			
10		Spring/Summer (Maths), List B												S								E					A					
20		Autumn/Spring (Maths), List C																														
20	MAT00043H	Maths Group Project (optional)		S																						EA				A		

10	COM00030H (option)	Project Management for Computer Scientists (PMCS)		S					A			E																				
10	COM00022H (option)	Analysable Real-Time Systems (ART1)		S					E			E	A																			
10	COM00031H (option)	Design of Analysable Real-Time Systems (DART)												S												E			A			
10	COM00027H (option)	Computer Vision (CVIS)												S												E			A			
10	COM00026H (option)	Computing by Graph Transformation (GRAT)												S												E			A			
10	COM00029H (option)	Introduction to Neural Networks (INNS)												S								A				E						
20	COM00009H (option)	Multi-Agent Interaction and Games (MAIG)		S																		E							A			
10	COM00028H (option)	Fundamentals of Machine Learning (FUML)		S								E	A																			
10	COM00032H (option)	Machine Learning & Probabilistic Graphical Models (MLPG)												S									A			E						
10	COM00023H (option)	Data-Oriented Specifications & their Analysis (DOSA)		S								E	A																			
10	COM00020H (option)	Concurrent System Analysis & Verification (CSAV)												S												E			A			
Stage 4																																
Credits	**Module**		**Autumn Term**										**Spring Term**										**Summer Term**									
	Code	**Title**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
40	MAT00006M	Final Year Mathematics Project		S								A														EA				A		
or																																
40	COM00079M	Final Year CS Project (PRIN)	S																						EA							

10		Autumn (Maths), List D		S								E	A																			
10		Spring (Maths), List E												S								E					A					
10	MAT00004M	Directed Learning in Maths (Autumn)		S								EA																				
10	MAT00004M	Directed Learning in Maths (Spring)												S								EA										
10	COM00042M (option)	Quantum Information Processing (QIPR)												S			E						A									
10	COM00045M (option)	Quantum Computation (QUCO)							S			E	A																			
10	COM00063M (option)	Static Analysis and Verification (SAVE)																	S			EA	A									
10	COM00082M (option)	Topics in Privacy and Security (PSEC)												S			E					A										
10	COM00122M (option)	Software Testing (SOTE)																	S			EA	A									
10	COM00066M (option)	Adaptive and Learning Agents (ALAS)																	S			E	A									
10	COM00069M (option)	Critical Systems (CRSY)		S			A		E				A																			
10	COM00071M (option)	Evolutionary Computation (EVCO)		S			E								A																	
10	COM00123M (option)	Functional Programming Technology (FUNC)																	S			E	A									

Optional module lists - these lists are indicative

If the programme requires students to select option modules from specific lists these lists should be provided below. If you need more space, use the toggles on the left to reveal ten further hidden rows.

Option List A	Option List B	Option List C	Option List D	Option List E	Option List F	Option List G	Option List H
Dynamical Systems MAT00011H	Differential Geometry MAT00006H	Numerical Analysis MAT00041H	Algebraic Geometry MAT00001M	Algebraic Groups MAT00003M	Maths Group Project MAT00043H		

Galois Theory MAT00008H	Topology MAT00044H		Lie Algebras and Lie Groups MAT00065M	Representation Theory of the Symmetric Group MAT00047M			
Number Theory MAT00023H	Cryptography MAT00034H		Semigroup Theory MAT00050M	Riemannian Geometry MAT00052M			
Algebraic Number Theory MAT00029H (compulsory)			Metric Number Theory MAT00049M				
Groups and Actions MAT00056H							

<table>
<tr><th colspan="9">Management and Admissions Information</th></tr>
<tr><td colspan="5">This document applies to students who commenced the programme(s) in:</td><td colspan="4">2018/19</td></tr>
<tr><td colspan="9">Interim awards available Interim awards available on undergraduate programmes (subject to programme regulations) will normally be: Certificate of Higher Education (Level 4/Certificate), Diploma of Higher Education (Level 5/Intermediate), Ordinary Degree and in the case of Integrated Masters the Bachelors with honours. Please specify any proposed exceptions to this norm.</td></tr>
<tr><td colspan="9">Certificate of Higher Education Generic Level 4/Certificate
Diploma of Higher Education Generic Level 5/Intermediate
BSc (Hons.) Ordinary Degree Generic Level 6/Honours
BSc (Hons) Computer Science and Mathematics Level 6/Honours</td></tr>
<tr><td colspan="9">Admissions Criteria</td></tr>
<tr><td colspan="9">TYPICAL OFFERS
A levels AAA/AAB
IB Diploma Programme
36/35 points including HL 6
in essential subjects
BTEC Extended Diploma
DDD (may vary for
combined programmes)</td></tr>
<tr><td colspan="9">Length and status of the programme(s) and mode(s) of study</td></tr>
<tr><td rowspan="2">Programme</td><td rowspan="2">Length (years)</td><td rowspan="2">Status (full-time/part-time) Please select</td><td rowspan="2">Start dates/months (if applicable – for programmes that have multiple intakes or start dates that differ from the usual academic year)</td><td colspan="5">Mode</td></tr>
<tr><td colspan="2">Face-to-face, campus-based</td><td colspan="2">Distance learning</td><td>Other</td></tr>
<tr><td>MMath in Mathematics and Computer Science</td><td>3</td><td>Full-time</td><td>n/a</td><td>Please select Y/N</td><td>Yes</td><td>Please select Y/N</td><td>No</td><td>n/a</td></tr>
<tr><td colspan="9">Language(s) of study</td></tr>
<tr><td colspan="9">English.</td></tr>
<tr><td colspan="9">Language(s) of assessment</td></tr>
</table>

English.		
Programme accreditation by Professional, Statutory or Regulatory Bodies (PSRB)		
Is the programme recognised or accredited by a PSRB		
Please Select Y/N:	No	if No move to next Section if Yes complete the following questions
Additional Professional or Vocational Standards		
Are there any additional requirements of accrediting bodies or PSRB or pre-requisite professional experience needed to study this programme?		
Please Select Y/N:	No	if Yes, provide details
University award regulations		
The University's award and assessment regulations apply to all programmes: any exceptions that relate to this programme are approved by University Teaching Committee and are recorded at the end of this document.		
Are students on the programme permitted to take elective modules? (See: https://www.york.ac.uk/media/staffhome/learningandteaching/documents/policies/Framework%20for%20Programme%20Design%20-%20UG.pdf)		
Please Select Y/N:	Yes	
Careers & Placements - 'With Placement Year' programmes		
Students on all undergraduate and integrated masters programmes may apply to spend their third year on a work-based placement facilitated by Careers & Placements. Such students would return to their studies at Stage 3 in the following year, thus lengthening their programme by a year. Successful completion of the placement year and associated assessment allows this to be recognised in programme title, which is amended to include 'with Placement Year' (e.g. BA in XYZ with Placement Year'). The Placement Year also adds a Programme Learning Outcome, concerning employability. (See Careers & Placements for details). In exceptional circumstances, UTC may approve an exemption from the 'Placement Year' initiative. This is usually granted only for compelling reasons concerning accreditation; if the Department already has a Year in Industry with criteria sufficiently generic so as to allow the same range of placements; or if the programme is less than three years in length.		
Programme **excluded** from Placement Year **?**	No	If yes, what are the reasons for this exemption:
Study Abroad (including Year Abroad as an additional year and replacement year)		

Students on all programmes may apply to spend Stage 2 on the University-wide North America/ Asia/ Australia student exchange programme. Acceptance onto the programme is on a competitive basis. Marks from modules taken on replacement years count toward progression and classification.

Does the programme include the opportunity to undertake other formally agreed study abroad activities? All such programmes must comply with the Policy on Study Abroad

https://www.york.ac.uk/staff/teaching/procedure/programmes/design/

Please Select Y/N:	No	

Additional information

Transfers out of or into the programme

ii) Transfers into the programme will be possible? (please select Y/N)		

Exceptions to University Award Regulations approved by University Teaching Committee

Exception Please detail any exceptions to University Award Regulations approved by UTC	**Date approved**

Date on which this programme information was updated:

27/03/2018

Please note:

The information above provides a concise summary of the main features of the programme and the learning outcomes that a typical student might reasonably be expected to achieve and demonstrate if they take full advantage of the learning opportunities that are provided.

Detailed information on the learning outcomes, content, delivery and assessment of modules can be found in the module descriptions.

The University reserves the right to modify this overview in unforeseen circumstances, or where the process of academic development, based on feedback from staff, students, external examiners or professional bodies, requires a change to be made. Students will be notified of any substantive changes at the first available opportunity.

Programme Map

Please note: the programme map below is in interim format pending the development of a University Programme Catalogue.

Programme Map: Module Contribution to Programme Learning Outcomes

Please complete the summary table below which shows how individual modules contribute to the achievement of programme learning outcomes.

Core modules should be mapped individually. If the programme offers multiple options that contribute to exactly the same PLOs you can group these, providing a statement that articulates how all of these contribute to the achievement of the programme learning outcomes. All modules, both core and optional, should be accounted for in the map.

The table maps the contribution to programme learning outcomes made by each module, in terms of the advance in understanding/ expertise acquired or reinforced in the module, the work by which students achieve this advance and the assessments that test it. This enables the programme rationale to be understood:
· Reading the table vertically illustrates how the programme has been designed to deepen knowledge, concepts and skills progressively. It shows how the progressive achievement of PLOs is supported by formative work and evaluated by summative assessment. In turn this should help students to understand and articulate their development of transferable skills and to relate this to other resources, such as the Employability Tutorial and York Award;
· Reading the table horizontally explains how the experience of a student at a particular time includes a balance of activities appropriate to that stage, through the design of modules.

Note: it is not expected that every module contributes directly to all PLOs, but every module should advance some of them.

Stage	Module		Programme Learning Outcomes							
			PLO1	PLO2	PLO3	PLO4	PLO5	PLO6	PLO7	PLO8

			Confidently and competently apply computational and mathematical thinking to problems, using skills in problem analysis, representation and abstraction, and the application of standard and higher level mathematical and computational techniques, including the theory and practice of programming and software engineering. [Computational and mathematical competence]	**Critically analyse statements, arguments or conjectures that underpin the theory of Mathematics and Computer Science, justifying the principles chosen for such critiques, and developing their own lines of well-founded reasoning. [Computational and mathematical reasoning]**	**Adapt to new and unfamiliar challenges in Computer Science and Mathematics, recognising appropriate ideas and approaches drawn from a range of technologies, languages, paradigms, models and mathematical theories, and informed by current research and scholarship. [Adaptability]**	**Conduct an independent investigation into a specialised area of Mathematics or Computer Science, at a level which engages with current research or cutting edge developments, by gathering material from a variety of sources, and synthesising this material into a well-organised and coherent account, or effective solution to a user-specified need or commercial imperative. [Independence]**	**Work effectively in a team, formulating and fulfilling obligations towards achieving goals by managing workloads, setting and meeting deadlines, and optimising resources , and taking leadership and responsibility for aspects of the work planned. [Team work]**	**Communicate complex ideas in Computer Science and Mathematics in a clear, unambiguous and organised manner, at a level appropriate for the intended recipients, and also present an effective summary of these ideas for an expert audience. [Communication]**	**Appreciate the wider context of Mathematics and Computer Science and their component disciplines, understand how these can contribute to and impact on society, develop an awareness of key legal and ethical issues, and operate as responsible professionals. [Awareness and professionalism]**	

Stage 1	Calculus	Progress towards PLO	Competently use the standard methods of differential and integral calculus	Justify the steps in the solution of calculus problems, or their application	Adapt standard calculus tools to problems slightly outside the standard format			Present clear and concise solutions to exercises	Understand how The Calculus has developed to enable the solution of a variety of mathematical problems related to geometry and the physical sciences	
		By working on (and if applicable, assessed through)	lecture material and exercises, with the support of seminars and formative feedback through marked work, and assessed by examination.	lecture material and exercises, with the support of seminars and formative feedback through marked work, and assessed by examination.	exercises, with formative feedback through marked work and seminars, and assessed by examination.			exercises, with the support of seminars and formative feedback through marked work.	lecture material, exercises and with the support of seminars.	

Stage 1	Algebra	Progress towards PLO	Competently use the standard algebra of vectors, matrices and related objects	Justify the steps and methods used in algebraic arguments	Adapt the standard algebraic tools to problems slightly outside the standard format			Present clear and concise solutions to exercises	Understand how algebraic methods have developed to allow the solution of a variety of mathematical problems related to symmetry, geometry, combinatorics and the physical sciences	
		By working on (and if applicable, assessed through)	lecture material and exercises, with the support of seminars and formative feedback through marked work, and assessed by examination.	lecture material and exercises, with the support of seminars and formative feedback through marked work, and assessed by examination.	exercises, with formative feedback through marked work and seminars, and assessed by examination.			exercises, with the support of seminars and formative feedback through marked work.	lecture material, exercises and with the support of seminars.	

Stage 1	Mathematical Skills 1: Reasoning and Communication	Progress towards PLO	Competence in working with sets, functions, logic and methods of proof	Practice different methods of mathematical reasoning	Adapt the standard concepts of set theory and logic to problems slightly outside the standard format	Find relevant resources and understand their content	Contribute towards the group report as a collaborative effort in exposition. This is part of the summative assessment.	Practice and develop written and oral communication skills, by preparing written group report and giving short oral presentation.	Understand how mathematics is used to solve a variety of interesting problems	
		By working on (and if applicable, assessed through)	lecture material and exercises, with formative feedback through marked work and the tutorials, and assessed by examination.	lecture material and exercises, with formative feedback through marked work and tutorials, and assessed by examination.	exercises, with formative feedback through marked work and tutorials, and assessed by examination.	the group project.	the group project.	exercises, with the support of tutorials and formative feedback through marked work, and the production of the group project and group presentation.	lecture material, exercises, and the topic of the group project, with support of the tutorials and as assessed by the group project/presentation.	

Stage 1	ICAR	Progress towards PLO	Design simple computer architectures from basic building blocks (CPU, memory, peripheral devices, systems buses) and then assess their performance for a given problem		Adapt to new instruction sets and future technologies		Work cooperatively in order to design, implement and test a program for a given problem	Explain the thought processes in solving complex computational problems	An initial consideration of the importance of security in system design	
		By working on (and if applicable, assessed through)	solving a series of exercises.		writing assembly language programs on a wide range of processor architectures.		*working with a partner during practical sessions. (This is a compulsory module.)*	working with a partner during practical sessions.	designing software which considers security.	

Stage 1	MFCS	Progress towards PLO		Acquire skills in abstract representation, problem analysis and formal reasoning, and a practical grasp of foundational ideas and methods	Capacity to acquire new terminologies, notations and conceptual models		Capacity to appreciate and combine different views	Explain the thinking about technical issues		
		By working on (and if applicable, assessed through)		solving a series of problems involving concepts of discrete maths and formal languages and automata.	working with unfamiliar notations and layered ideas in discrete mathematics and formal languages and automata.		*working in small groups to solve problems. (This is a compulsory module.)*	working in small groups to solve problems.		

Stage 1	TPOP	Progress towards PLO	(a) Develop skills including problem solving, abstract representation, ability to select or develop an appropriate algorithm/data structure and to develop appropriate software testing strategies; (b) Obtain the basic ability to build and maintain software systems, enabling larger software engineering projects	Familiarity with the theoretical tools used to understand algorithms and their complexity	Gain the ability to develop algorithms and data structures independent of platform, and the ability to transfer skills learnt on one programming paradigm to another one		Appreciate the issues of how to communicate, argue and assess the proposed analysis of the problem, and the choice of design implementation			

		By working on (and if applicable, assessed through)	(a) implementing a series of solutions to problems (well known and new) in a specific programming language and paradigm; (b) developing small pieces of software, and modifying code written by another programmer.	analysing well-known algorithms and data structures, in addition to solving a series of theoretical problems.	practising analysis of programs using different theoretical techniques, and implementing algorithms and data structures using two different languages from distinct paradigms.		designing and implementing a solution to a larger problem in a small group of students over a period of two weeks. (This is a compulsory module.)			
Stage 1	SKIL	Progress towards PLO				Investigate a topic of choice, and construct a critical analysis of a small number of items of relevant literature		Appreciate some of the possible different communication methods, and consider different possible audiences	Start to learn about the wider (legal and ethical) implications of Computer Science, and set personal goals for achievement by graduation	

		By working on (and if applicable, assessed through)				preparing a critical analysis of a paper in the area, and using this as a basis for other communication activities.		engaging with a number of different communication methods (written report, oral presentation to tutorial group, poster) for a number of different audiences (peers, employer, academic report).	analysing computing job adverts to discern the skills and competencies required for the post, and drafting a prospective CV.	
Stage 2	Pure Mathematics	Progress towards PLO	Understand the language of abstract mathematics, and work confidently with the ideas which form the basis of abstract algebra, number theory and geometry	Reproduce, with understanding, central arguments used in algebra, number theory and geometry, and be able to adapt these to similar situations	Recognise and be able to put into practice the principles of abstract mathematics in unfamiliar settings			Present coherent, clear and concise solutions to exercises	Appreciate, and be able to explain, how the fundamental ideas of algebra, number theory and geometry have arisen from, and enabled the solution of, some important problems in science	

		By working on (and if applicable, assessed through)	lecture material and exercises, with the support of seminars and formative feedback through marked work, and assessed by examination.	lecture material and exercises, with the support of seminars and formative feedback through marked work, and assessed by examination.	exercises and with formative feedback through marked work and the seminars, and assessed by examination.			exercises, with the support of seminars.	lecture material, exercises and with the support of seminars.	
Stage 2	Linear Algebra	Progress towards PLO	Use the standard methods of basic linear algebra and matrix theory, and their theoretical justification through abstract algebra	Prove standard results in abstract linear algebra	Apply basic linear algebra and matrix theory to a range of unfamiliar situations			Present clear and concise solutions to exercises	Appreciate the power of the abstract approach to linear algebra and the variety of uses of linear algebra	

		By working on (and if applicable, assessed through)	lecture material and exercises, with the support of seminars and formative feedback through marked work, and assessed by examination.	lecture material and exercises, with the support of seminars and formative feedback through marked work, and assessed by examination.	exercises, and with formative feedback through marked work and the seminars, and assessed by examination.			exercises, with the support of seminars.	lecture material, exercises and with the support of seminars.	
Stage 2	Vector Calculus	Progress towards PLO	Use the standard methods of multi-variable differential and integral calculus to work with functions of many variables and vector fields		Apply the standard methods to problems which require a level of interpretation to set up the application.			Present clear and concise solutions to exercises	Understand and appreciate how the methods of vector calculus arise from important problems in the study of the physical world	

		By working on (and if applicable, assessed through)	lecture material and exercises, with the support of seminars and formative feedback through marked work, and assessed by examination.		exercises and with formative feedback through marked work and the seminars, and assessed by examination.			exercises, with the support of seminars.	lecture material, exercises and with the support of seminars.	
Stage 2	COCO	Progress towards PLO	Understand the difference between solvable and unsolvable problems, and be able to analyse the computational complexity of algorithms	Appreciate the relevance of formal methods and be able to apply them to reason about software and hardware systems	Adapt to the properties of new languages and paradigms					
		By working on (and if applicable, assessed through)	studying (semi-) decidable languages, Turing-computable functions and the time and space complexity of Turing machines.	formally analysing correctness, termination and complexity properties of Turing machines.	studying computability and complexity in a basic computational model.					

Stage 2	POPL	Progress towards PLO	Ability to judge the most effective programming techniques for a particular computational requirement	Ability to make effective use of current and future programming language implementations	Adapt to changes in language fashions, and new technologies as they occur during their careers			Communicate the choice of principles and technical rationales		
		By working on (and if applicable, assessed through)	characterising different programming principles, including concurrency.	implementing a series of simple programming languages displaying the abstract principles, and solving similar classic problems in several different languages.	understanding and applying the fundamentals of different programming languages.			solving formative and summative problems in a variety of languages, and writing concise and focussed explanations of the solutions.		
Stage 2	ARIN	Progress towards PLO	Ability to apply computational thinking to problems that can be solved using core AI techniques	Acquisition of core AI techniques	Ability to apply knowledge of AI as part of a larger problem, and to transfer acquired skills to solving unseen problems				Exposure to wider applications of AI across engineering	

		By working on (and if applicable, assessed through)	practising the key principles underlying search algorithms, machine learning algorithms and approaches to and formalisms for problem and knowledge representation.	learning the key principles underlying search algorithms, machine learning algorithms and approaches to and formalisms for problem and knowledge representation.	using industrial-strength tools for specific problems in AI, and considering a range of problems that can be addressed using AI techniques.				working on a variety of problems across problem domains.	
Stage 2	SYST	Progress towards PLO	Ability to build systems that exhibit required non-functional properties including data consistency, process separation and (aspects of) security	Ability to apply the principles of resource management, networks, concurrency and databases	Adapt to new systems programming approaches			Gain experience of communicating with stakeholders	Develop engineering and problem-solving skills for building systems that can be applied to current and future industrial problems	

		By working on (and if applicable, assessed through)	understanding how hardware supports an Operating System's provision of resource management.	understanding these principles and the characteristics of these topics.	experiencing the principles of and different approaches to systems programming (including networks and databases).			solving formative and summative problems, together with a variety of laboratory problems, requiring writing concise and focussed explanations of the solutions.	solving realistic problems posed in laboratory sessions.	

Stage 2	IMPL	Progress towards PLO	(a) Develop and be able to recognise situations in which a pipeline architecture can be applied, including its associated techniques, to represent sentences of formal languages; (b) Generally improve acquired software engineering skills	Build understanding of the relationship between high and low level expression of computation	Improve adaptability to new programming languages and paradigms					

		By working on (and if applicable, assessed through)	(a) implementing appropriate algorithms for each phase of the compiler pipeline, drawing on foundations such as formal language theory and Natural Deduction presentations of types and semantics; (b) developing all the components of a compiler.	exploring the relationship between source code and machine-level code.	experiencing a new programming language paradigm, lazy functional programming.					

Stage 2	VIGR	Progress towards PLO	Process visual and graphical information, and develop appropriate algorithms and programs	Understand the requirements of visual information processing, and implement computational thinking into software for analysing images and for creating computer graphics	Adapt to any programming language and library used for processing visual information and in computer graphics			Communicate with technical and non-technical people about the solutions for and suitable approaches to complex computational problems of visual information processing, in a clear and organised manner		

		By working on (and if applicable, assessed through)	applying the theory of visual information processing and computer graphics into programs, and testing them in processing visual representation data, using physical sciences understanding and computing skills.	applying computational modelling of visual information, using specific algorithms for image analysis (computer vision) and for creating images (computer graphics).	learning the principles of visual information analysis, including the physics and geometry of scene information in visual systems.			learning and understanding how to represent and process visual information and its underlying principles.		
Stage 3	Differential Geometry	Progress towards PLO	Understand and be able to calculate the standard geometric properties of curves and surfaces	Justify the steps made in differential geometric arguments	Decide which geometric properties can be evaluated given different representations of a curve or surface			Present clear and concise solutions to exercises	Comprehend the power and central importance of coordinate invariance in geometr	

		By working on (and if applicable, assessed through)	lecture material and exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	lecture material and exercises, with the guidance and support of seminars, and as assessed through examination.	lecture material and exercises, with the guidance and support of seminars, and as assessed through examination.			exercises, with the support of seminars and formative feedback through marked work.	lecture material and seminar discussion.	
Stage 3	Dynamical Systems	Progress towards PLO	Analyse the qualitative features of simple dynamical systems	Justify the conclusions of a qualitative analysis of a nonlinear system	Adapt standard techniques to unfamiliar nonlinear dynamical systems			Present clear and concise solutions to exercises	Comprehend the value of qualitative analysis in the context of dynamical systems	
		By working on (and if applicable, assessed through)	lecture material and exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	lecture material and exercises, with the guidance and support of seminars, and as assessed through examination.	exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.			exercises, with the support of seminars and formative feedback through marked work.	lecture material, exercises and seminar discussion.	

Stage 3	Number Theory	Progress towards PLO	Understand and be able to use a wide range of methods from analytic number theory, Diophantine equations and Diophantine approximation	Comprehend and produce mathematical arguments to support claims concerning fundamental properties of numbers	Apply analytic/number theoretic foundations to solve specific problems (e. g.. counting primes, Waring's problem) and develop new areas (Diophantine approximation)			Present clear and concise solutions to exercises	Comprehend the power and central importance of Number Theory to solve deep concrete problems	
		By working on (and if applicable, assessed through)	lecture material and exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	lecture material and exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	lecture material and exercises, with the guidance and support of seminars, and as assessed through examination.			exercises, with the support of seminars and formative feedback through marked work.	lecture and problems class material, seminar discussion, and exercises.	

Stage 3	Algebraic Number Theory	Progress towards PLO	Understand what is meant by “Algebraic Number Theory”, and be well-versed in many of the standard techniques	Justify the steps made in algebraic and number-theoretic arguments	Recognise various problems in algebraic number theory and apply appropriate techniques to solve them (e.g. factorisation of algebraic integers or ideals; identification of prime and irreducible elements in rings of integers)			Present clear and concise solutions to exercises	Understand the genesis of algebraic number theory through attempts to prove Fermat’s Last Theorem, appreciate how algebraic ideas can influence number theory and vice versa, and witness that mathematics is a developing subject by exposure to open problems	

		By working on (and if applicable, assessed through)	lecture material and exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	lecture material and exercises, with the guidance and support of seminars, and as assessed through examination.	lecture material and exercises, with the guidance and support of seminars, and as assessed through examination.			exercises, with the support of seminars and formative feedback through marked work.	lecture material, problem sheets and lecture/seminar discussion.	
Stage 3	Galois Theory	Progress towards PLO	Understand and be able use symmetry in the solution of polynomial equations, and the correspondence that reconstructs fields and their subfields inside groups of symmetry	Follow the reasoning behind the construction of the Galois group of a field extension, and the correspondence between its subgroups and intermediate fields	See how acquired algebraic foundations can be applied to solve specific problems (in particular, the algebraic solutions of equations and the construction using ruler and compass of specific objects)			Present clear and concise solutions to exercises	See how the abstract algebra learned in the first two years of the degree can be used in a non-trivial way to solve seemingly intractable but nevertheless quite concrete problems	

		By working on (and if applicable, assessed through)	lecture material and exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	lecture material and exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	lecture material and exercises, with the guidance and support of seminars, and as assessed through examination.			exercises, with the support of seminars and formative feedback through marked work.	lecture material, problem sheets and lecture/seminar discussion.	
Stage 3	Cryptography	Progress towards PLO	Understand and be able to work with some of the mathematical underpinnings of modern cryptography	Follow the reasoning as to why a primality test or a factorisation algorithm works	Apply acquired mathematical knowledge to new areas; namely, certain cryptographic systems			Present clear and concise solutions to exercises	See how the mathematics seen in previous modules is used by modern cryptography to ensure privacy in a modern web-based world	

		By working on (and if applicable, assessed through)	lecture material and exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	lecture material and exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	lecture material and exercises, with the guidance and support of seminars, and as assessed through examination.			exercises, with the support of seminars and formative feedback through marked work.	lecture and problems class material, seminar discussion, and exercises.	
Stage 3	Topology	Progress towards PLO	Understand the notion of topological spaces, invariants and fundamental groups and be able to apply the ideas in an abstract setting.	Answer questions and solve problems about topological spaces that require reasoned, solid mathematical arguments	Determine when a given space is a topological space, be able to determine when two spaces are essentially the same and be able to determine what, if any, topological invariants the spaces possess			Present clear and concise solutions to exercises	Appreciate and understand the central role that topology plays in mathematics and the wider world	

		By working on (and if applicable, assessed through)	lecture material and exercises, with the guidance and support of seminars, and as assessed through examination.	lecture material and exercises, with the guidance and support of seminars, and as assessed through examination.	lecture material and exercises, with the guidance and support of seminars, and as assessed through examination.			exercises, with the support of seminars and formative feedback through marked work.	lecture material and exercises, with the guidance and support of seminars, and as assessed through examination.	
Stage 3	Groups and Actions	Progress towards PLO		Follow logical steps in arguments and justify those steps				Present clear and concise solutions to exercises	Appreciate and understand the central role that topology plays in mathematics and the wider world	
		By working on (and if applicable, assessed through)	lecture notes and exercise sheet material.	exercises from exercise sheets and past exam papers.	lecture notes, exercise sheets and previous exams (and ideally making up small exercises of one's own).			exercises, with the support of seminars and formative feedback through marked work.	lecture notes, seminar participation, viewing the course as a whole, exercises and exam papers.	

Stage 3	Numerical Analysis	Progress towards PLO	Ability to apply numerical approximation techniques to a range of standard mathematical problems	Justify which particular numerical method is appropriate in a given context, and in which sense the approximation error is small	Decide which of a range of approximation techniques can be used in unfamiliar application problems			Communicate mathematical arguments in Numerical Analysis in writing	Comprehend the value and power of numerical approximation techniques, and their applicability to the modern world	
		By working on (and if applicable, assessed through)	lecture materials, computer practicals, assessed computer-based coursework, as well as being assessed in the examination.	lecture materials, computer practicals, written coursework, and as assessed through examination.	lecture materials, computer practicals.			assessed written coursework.	lecture material, computer practicals, coursework.	

Stage 3	Maths Group Project	Progress towards PLO		Provide a clear critical analysis of the mathematical principles under investigation	Apply methods from other modules, as appropriate, to the topic of the project	Make an individual contribution to the study of background material by the group and be able to properly reference sources of information for the written project, and prepare an individual poster	Work effectively as part of a group tasked with the preparation of a 30 page (approx.) dissertation	Present a clear written account of the topic under investigation, as well as a concise summary in poster form		

		By working on (and if applicable, assessed through)		the background material relevant to the project, with the support of peer discussion and with the guidance of the project supervision meetings.	the development of the project material, with the guidance of the project supervisor.	the background for the group project and the written report, with support on proper referencing from the lecture, and a session on poster preparation.	the project material, collectively or individually, as agreed amongst members of the group, and under the overall guidance of the project supervisor.	the written report (approx. 30 pages in total), in collaboration with the peer group, and the individually prepared poster. Formative assessment: two short individual assignments during term. Summative assessment: the group project and the poster presentation.		

Stage 3	Analysable Real-Time Systems (ARTS I)	Progress towards PLO	Apply computational thinking in order to abstract the relevant application timing requirements and computing platform characteristics, so that predictions can be made as to whether real-time requirements will be met when the system is exhibiting its worst-case timing behaviour	Appreciate the need to use software engineering techniques that help to deal with large and complex systems (threads and modules), and also appreciate the pros and cons of writing low-level software in a high-level language	Adapt to new languages, whether they are domain-specific or generic		Increase capacity to appreciate and combine different views		Ability to apply various approaches to fault-tolerant computing	

		By working on (and if applicable, assessed through)	doing schedulability analysis problems using response time analysis on various application use cases and for different execution platforms.	producing both high and low level software components for a simple embedded system (the Ball Sorter) which consists of multiple threads of execution.	understanding how the computational model needed to support schedulability analysis can be supported in Ada, and by focussing on the underlying principles that Ada supports.		working in pairs to develop software.		understanding how to build resilient systems.	

Stage 3	Dependable Analysable Real-Time Systems (DARTS or ARTS II)	Progress towards PLO	Apply computational thinking in order to abstract the relevant application timing requirements and computing platform characteristics, so that predictions can be made as to whether real-time requirements will be met when the system is exhibiting its worst-case timing behaviour	Appreciate the need to use software engineering techniques that help to deal with large and complex systems (threads and modules), and also appreciate the pros and cons of writing low-level software in a high-level language	Adapt to new languages, whether they are domain-specific or generic		Increase capacity to appreciate and combine different views		Ability to apply various approaches to fault-tolerant computing	

		By working on (and if applicable, assessed through)	doing schedulability analysis problems using response time analysis on various application use cases and for different execution platforms.	producing both high and low level software components for a simple embedded system (the Ball Sorter) which consists of multiple threads of execution.	understanding how the computational model needed to support schedulability analysis can be supported in Ada, and by focussing on the underlying principles that Ada supports.		working in pairs to develop software.		understanding how to build resilient systems.	
Stage 3	Computer Vision (CVIS)	Progress towards PLO		Understand the complexities of algorithm design in an interdisciplinary context constrained by the underlying science of human vision, and apply this to real world problems	Increase capacity to address problems in an interdisciplinary way, not necessarily confined to CS			Develop critical writing skills		

		By working on (and if applicable, assessed through)		studying the principles underlying computer vision algorithms, both those based on algorithmic and those based on the underlying science (often physics, geometry or the biology of vision).	solving problems of algorithm design using models provided by a diverse set of disciplines.			undertaking a reading exercise and answering critical questions about a selected scientific paper describing a computer vision algorithm.		
Stage 3	Computing by Graph Transformatio n (GRAT)	Progress towards PLO	Ability to write graph problems for solving problems in graph-like domains and reason about program correctness and complexity	Develop an appreciation for problem solving and formal reasoning in rule-based systems and domain-specific languages	Ability to adapt to the properties of new domain-specific languages					

		By working on (and if applicable, assessed through)	developing small rule-based programs for manipulating graph structures and analysing the properties of these programs.	studying the properties of rule-based systems in the domain of graphs, and the semantics and use of a non-deterministic programming language on graphs.	studying the properties of a rule-based and non-deterministic domain-specific language.					
Stage 3	Information & Coding Theory - General Aspects	Progress towards PLO	Apply computational thinking to modern issues associated with data storage and transmission, such as protecting information from loss, and from other adverse effects associated with limited and incomplete forms of transmission	Understand the mathematical principles and difficulties which are behind the protection of confidential and private information	Adapt and extend acquired knowledge to other mathematical models, such as quantum information and computation, and network design			Ability to communicate with both mathematicians and computer scientists		

		By working on (and if applicable, assessed through)	analysing and applying methods and algorithms for data compression, and applying the principles of error correction and channel coding.	learning and practising the basic tools of cryptography.	studying and understanding the fundamental notions of information, coding and network theory.			learning the most basic definitions and theorems in information theory and also applying these tools to practical examples.		
Stage 3	Information & Coding Theory - General Aspects	Progress towards PLO	Apply computational thinking to modern issues associated with data storage and transmission, such as protecting information from loss, and from other adverse effects associated with limited and incomplete forms of transmission	Understand the mathematical principles and difficulties which are behind the protection of confidential and private information	Adapt and extend acquired knowledge to other mathematical models, such as quantum information and computation, and network design			Ability to communicate with both mathematicians and computer scientists		

		By working on (and if applicable, assessed through)	analysing and applying methods and algorithms for data compression, and applying the principles of error correction and channel coding.	learning and practising the basic tools of cryptography.	studying and understanding the fundamental notions of information, coding and network theory.			learning the most basic definitions and theorems in information theory and also applying these tools to practical examples.		
Stage 3	Introduction to Neural Network (INCA)	Progress towards PLO	Ability to select the appropriate tools and paradigms to solve specific problems	Apply computational thinking to develop solutions to a broad range of complex problems	Adapt more readily to new technologies and paradigms		Communicate with technical stakeholders about complex issues	Communicate with technical stakeholders about complex issues		
		By working on (and if applicable, assessed through)	implementing neural network training algorithms, understanding their characteristics and analysing their performance.	implementing and using different neuron models and neural network architectures.	applying different types of neural networks to a range of real problems.		working in small groups to analyse problems, and by giving presentations about solutions.	working in small groups to analyse problems, and by giving presentations about solutions.		

Stage 3	Multi-Agent Interaction and Games (MAIG)	Progress towards PLO	Ability to solve practical problems by applying abstract interaction models and to perform a precise analysis of complex multi-agent situations	Ability to define optimal individual and group behaviours and the impact of interaction environment designs on these						
		By working on (and if applicable, assessed through)	working with mathematical abstractions and applying them to problem solving.	modelling and analysing agent interactions as mathematical games.						
Stage 3	Fundamentals of Machine Learning	Progress towards PLO	Students will be able to develop their own software solutions to novel data analysis problems	Students will be able to apply computational thinking to develop Bayesian learning algorithms for complex learning problems	Students will be able to adapt existing machine learning algorithms to new domains and new problems			Students gain experience in communicating their analysis and conclusions on moderately complex datasets	Students will be able to analyse and interpret different types of data across disciplines	

		By working on (and if applicable, assessed through)	implementing a range of different machine learning algorithms.	learning the statistical and probabilistic principles underlying Bayesian machine learning.	understanding how a range of data analysis problems can be solved.			writing a coursework report on specific problem domains.	performing predictive analysis tasks on a variety of data coming from different application domains.	
Stage 3	Machine Learning and Probabilistic Graphical Models	Progress towards PLO	Students will be able to develop their own software solutions to novel data analysis problems	Students will be able to apply computational thinking to develop Bayesian learning algorithms for complex learning problems	Students will be able to adapt existing machine learning algorithms to new domains and new problems			Students gain experience in communicating their analysis and conclusions on moderately complex datasets	Students will be able to analyse and interpret different types of data across disciplines	
		By working on (and if applicable, assessed through)	implementing a range of different machine learning algorithms.	learning the statistical and probabilistic principles underlying Bayesian machine learning.	understanding how a range of data analysis problems can be solved.			writing a coursework report on specific problem domains.	performing predictive analysis tasks on a variety of data coming from different application domains.	

Stage 3	Data-oriented specifications and their analysis	Progress towards PLO	Students will become able to carry out problem analysis using the mathematical foundations of computer science	Students will understand programming as part of an engineering discipline with solid mathematical foundations	Students will be able to handle a variety of modelling and analysis techniques to deal with systems descriptions			Students will have an awareness of the issues of ambiguity and incompleteness in informal descriptions	Students will understand how to specify and develop alternative software designs and meet the users' needs for reliability	
		By working on (and if applicable, assessed through)	writing formal models using a data modelling language and a process algebra.	learning the mathematical principles of correctness.	learning to write models using mathematical notations.			writing formal descriptions of systems.	learning formal characterisations of the notion of correctness.	
Stage 3	Concurrent System Analysis and Verification	Progress towards PLO	Students will become able to carry out problem analysis using the mathematical foundations of computer science	Students will understand programming as part of an engineering discipline with solid mathematical foundations	Students will be able to handle a variety of modelling and analysis techniques to deal with systems descriptions			Students will have an awareness of the issues of ambiguity and incompleteness in informal descriptions	Students will understand how to specify and develop alternative software designs and meet the users' needs for reliability	
		By working on (and if applicable, assessed through)	writing formal models using a data modelling language and a process algebra.	learning the mathematical principles of correctness.	learning to write models using mathematical notations.			writing formal descriptions of systems.	learning formal characterisations of the notion of correctness.	

Stage 3	Project Management for CS (PMCS)	Progress towards PLO		Students learn how to critically analyse statements that underpin theory	Students learn how to apply management principles to deliver working systems, on time	Students learn how to engineer solutions to problems in which computation forms a significant part of their independent project	Students develop their leadership and effectively teamwork skills.		Students learn to recognise the wider context of CS/Maths such as professional, legal, social, ethical and other issues and address them appropriately	
		By working on (and if applicable, assessed through)		by developing skills in project management and risk assessment. Assessed in the report.	by developing skills in project management and risk assessment. Assessed in the report.	By considering how projects are developed withing regulatory frameworks, including cybersecurity management.	By working as a team to analyse different aspects of case studies and synthesize a single report.		by considering such issues in relation to case studies, and in relation to their selected individual project Assessed in the report.	

Stage 4	Final Year Maths Project	Progress towards PLO		Justify the reasoning and/or choice of methods used in the mathematics relevant to the project topic	Adapt and apply the mathematics learned during the degree to some challenging topic outside the MMath degree syllabus	Conduct an independent study into a specialised area of mathematics, by researching material from a variety of sources, and be able to verify independently some of the results described in the literature		Communicate advanced mathematical ideas clearly in writing at M-level, and also be able to present an effective summary of these ideas for non-experts in a presentation	Understand and be able to explain the context and/or role of the mathematics presented in the dissertation, both within mathematics and more widely in the sciences to which the project topic is relevant	

		By working on (and if applicable, assessed through)		the project dissertation, with the support of the project supervisor and as assessed by the dissertation.	material found in the literature, with the support of the project supervisor and as assessed by the dissertation.	the project dissertation, with the support of the project supervisor and as assessed by the dissertation.		the project dissertation (30-40 pages) and the presentation talk (10 minutes), with the support of the project supervisor, lectures and demonstration on writing and presenting mathematics, as assessed by the writing assignments, the dissertation and the presentation talk.	the introduction and conclusion of the dissertation, and the writing assignment which addresses that aspect.	
Stage 4	Algebraic Geometry, Algebraic Groups	Progress towards PLO	Work with the algebraic methods which reflect the geometry of algebraic sets or groups	Produce their own lines of reasoning to prove statements about algebraic sets or groups	Apply the ideas of algebraic geometry/groups to the analysis of unfamiliar concrete examples			Present clear written or seminar presentations of worked exercises		

		By working on (and if applicable, assessed through)	lecture material and exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.			exercises, with the support of seminars and formative feedback on marked work and presentations.		
Stage 4	Lie Algebras and Lie Groups	Progress towards PLO	Use the standard tools of Lie algebra and matrix Lie group theory, particularly those relevant to the classification of finite dimensional Lie algebras	Produce their own lines of reasoning to prove statements about Lie algebras and matrix Lie groups	Apply the ideas of Lie theory to the analysis of unfamiliar concrete examples			Present clear written or seminar presentations of worked exercises		

		By working on (and if applicable, assessed through)	lecture material and exercises , with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.			exercises, with the support of seminars and formative feedback on marked work and presentations.		
Stage 4	Semigroup Theory	Progress towards PLO	Develop an understanding of the algebraic theory of semigroups; an example of a class of algebras where not every congruence is determined by a subalgebra	Produce their own lines of reasoning to prove statements, both general and specific, about properties of semigroups	Apply the ideas of semigroup theory to the analysis of unfamiliar concrete examples			Present clear written or seminar presentations of worked exercises		

		By working on (and if applicable, assessed through)	lecture material and exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	lecture material and exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.			exercises, with the support of seminars and formative feedback on marked work and presentations, and interaction with peers.		
Stage 4	Metric Number Theory	Progress towards PLO	Work with a range of techniques and ideas in Diophantine approximatio n, geometry of numbers, probability theory and fractals	Produce their own lines of reasoning to prove statements concerning systems of Diophantine inequalities	Apply the techniques and ideas of metric number theory to the analysis of unfamiliar concrete examples			Present clear written or seminar presentations of worked exercises		

		By working on (and if applicable, assessed through)	lecture material and exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.			exercises, with the support of seminars and formative feedback on marked work and presentations.		
Stage 4	Representation Theory of the Symmetric Group	Progress towards PLO	Use the standard tools of the representation theory of finite groups to understand the representations of the symmetric group	Produce their own lines of reasoning to prove statements about representations of the symmetric group	Apply the ideas of representation theory to the analysis of unfamiliar concrete examples			Present clear written or seminar presentations of worked exercises		

		By working on (and if applicable, assessed through)	lecture material and exercises , with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.			exercises, with the support of seminars and formative feedback on marked work and presentations.		
Stage 4	Riemannian Geometry	Progress towards PLO	Work with the standard tools required for understandin g the geometry of Riemannian manifolds	Produce their own lines of reasoning to prove statements, both general and specific, about the geometry of Riemannian manifolds	Apply the ideas of Riemannian geometry to the analysis of unfamiliar concrete examples			Present clear written or seminar presentations of worked exercises		

		By working on (and if applicable, assessed through)	lecture material and exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.	exercises, with the guidance and support of seminars, and through feedback on marked work, and as assessed through examination.			exercises, with the support of seminars and formative feedback on marked work and presentations, and interaction with peers.		
Stage 4	Directed Learning in Mathematics	Progress towards PLO	Understand and be able to use methods relevant to the area of specialism of the DLM	Critically analyse the literature to obtain a clear understanding of the topic under discussion			Take responsibility for certain aspects of the group's learning	Write clear and concise work as required by the assessment of the DLM		
		By working on (and if applicable, assessed through)	recommended reading and seminars.	recommended reading and seminars.			presenting material in seminars, under the direction of the module leader.	coursework, with the support of the seminars.		

Stage 4	Quantum Computation (QUCO)	Progress towards PLO	Students will be able to engineer solutions to simple computational problems with limited information access	Students will develop new paradigms for dealing with complex problems through novel representations	Students will be prepared for the introduction of new quantum languages, algorithms and protocols			Students will be able to communicate and advise stakeholders whether a particular problem would be suitable for a quantum solution	Students will be able to understand and participate in future trends in cryptosystems	
		By working on (and if applicable, assessed through)	by learning about computation on superpositions.	by analysing quantum algorithms.	by solving benchmark quantum computational algorithms.			by understanding the basics of quantum computation.	by studying Shor's algorithm for breaking public key cryptography.	

Stage 4	Static Analysis and Verification (SAVE)	Progress towards PLO	Students will understand how to specify and develop alternative algorithms	Students will become able to analyse and test their programs using mathematical representations and abstractions to describe interfaces, and will understand programming as part of an engineering discipline with solid mathematical foundations	Students will learn to deal with a variety of testing techniques and a variety of techniques to ensure reliability				Students will learn about the issues related to ambiguity and incompleteness in informal descriptions	
		By working on (and if applicable, assessed through)	by learning formal characterisations of the notion of correctness.	by learning to write formal assertions within code, and by learning the mathematical principles of correctness.	by learning to write assertions using mathematical notations.				by learning about formal modelling.	

Stage 4	Topics in Privacy and Security (PSEC)	Progress towards PLO	Students will be able to employ rigorous software engineering processes, capturing security requirements as an integral part of the requirements of a system, and thus designing, implementing and testing software with security in mind	Students will be able to analyse the effectiveness of existing and new cybersecurity solutions within specific scenarios, to contribute to the rigorous development of new security controls, and to analyse the security risks of systems they are responsible for	Students will be able to judge the challenges associated with new classes of threats and vulnerabilities, and the relative merits of new types of security controls that will emerge in the future		Students will enhance their ability to make effective contributions as part of project teams	Students will enhance their ability to describe their team's work to a knowledgeable audience	Students will be in a better position to lead responsible professional careers, aware of key legal and ethical issues associated with their professional roles and with those of the computer systems they develop and use	

		By working on (and if applicable, assessed through)	by understanding the principles underlying secure software development, through ethical hacking and security protocol modelling and analysis during practical sessions, and by software tool development for the open assessment.	by learning the formal models underpinning a wide range of access control mechanisms, a broad spectrum of cryptosystems and security protocols, and approaches to security risk analysis.	by studying recent research papers and industry reports to identify the assumptions, limitations, benefits and tradeoffs of different security controls and the characteristics of the threats they prevent or mitigate.		by working on a team project on biometrics for the formative assessment, and working in small groups during the security risk management practicals.	by presenting to the entire cohort their team project.	by learning about the influence of the social, legal and ethical context on the use of encryption, and about legal and ethical aspects of security risk management.	

Stage 4	Software Testing (SOTE)	Progress towards PLO	Students will learn how to test systems and to evaluate and justify their testing, and how to carry out testing under realistic conditions at all stages of the lifecycle	Students will be able to create testing plans that are both tractable and justifiably adequate	Students will become able to test systems implemented with a range of technologies, languages and paradigms, and gain some ability to move bleeding-edge technologies into the space of those we trust			Students will be able to communicate test plans and results in a clear and unambiguous form	Students will become more aware of the possible consequences of unethical and unprofessional behaviour	

		By working on (and if applicable, assessed through)	by design testing approaches taking account of stakeholder needs and the implementation details of the system under test, and by testing a variety of systems, including those for which specifications are inadequately defined.	by analysing systems in terms of behaviours and properties.	by designing testing approaches for those technologies, languages and paradigms, including systems built with technologies at the edge of our understanding.			by writing testing reports.	by considering a variety of historical cases where inadequate testing cause significant risk, loss or harm.	

Stage 4	Adaptive and Learning Agents (ALAS)	Progress towards PLO	Students will be capable of applying their machine learning skills effectively in an industrial setup with a minimum of preparation	Students become able to combine the multi-agent paradigm with machine learning and evolutionary techniques, to develop intelligent autonomous software agents capable of optimising their performance	Students will become capable of building on various AI skills and combining them effectively	Students will work individually	Students will work in teams		Students become able to incorporate elements of cutting-edge research in their work	

		By working on (and if applicable, assessed through)	by studying and using an industrial-strength machine learning and data mining tool.	by acquiring hands-on skills with the encoding of agent behaviour in a way that is suited to the application of machine learning and evolutionary algorithms, and practising the use of selected examples of such algorithms.	by responding to explicit engineering challenges which require combined skills from several areas of AI.	by acquiring hands-on skills with the encoding of agent behaviour in a way that is suited to the application of machine learning and evolutionary algorithms, and practising the use of selected examples of such algorithms.	by acquiring hands-on skills with the encoding of agent behaviour in a way that is suited to the application of machine learning and evolutionary algorithms, and practising the use of selected examples of such algorithms.		by studying and implementing ideas from recent publications and patents in the practicals and open book hands-on assessment.	

Stage 4	Critical Systems (CRSY)	Progress towards PLO	Students will be able to choose between different hardware- and software-based solutions to achieve the right balance between predictability and fault tolerance, and make pragmatic decisions over the whole development and maintenance lifecycle	Students will be able to critically assess a range of complex scenarios at different levels of abstraction and determine how these can be mitigated through process and design	Students will be able to select and apply appropriate solutions to future safety-critical problems		Students will be able to assess how teams should be managed to support the development and maintenance of systems	Students will be able to comprehend, distill and explain complex scenarios and development challenges	Students will be able to comprehend the motivation and impact of cutting-edge research, and identify legal, ethical and societal responsibilities	

		By working on (and if applicable, assessed through)	by studying different design solutions for given problems, and analysing how individual decisions affect other parts of the engineering lifecycle for a system.	by analysing how accidents have occurred in the past and how they might occur in the future.	by applying a range of techniques in a variety of systems and contexts.		by studying how accidents have occurred due to inappropriate teamwork and management, and what certification standards state.	by taking part in lecture discussions and through the seminars given as part of the assessment.	by practising and delivering solutions through all activities in the learning design, and considering the potential impact of how systems are developed and operated.	
Stage 4	EVCO	Progress towards PLO	Solve will be able to solve ill-understood problems	Students become able to work at a higher level of computational abstraction in search of solutions	Students will be able to adapt to the new paradigm of metaheuristic search	Students will use their theoretical and practical knowledge of evolutionary search				
		By working on (and if applicable, assessed through)	by applying evolutionary search.	by applying computational thinking to biological processes.	by solving problems using evolutionary search.	by developing a solution to a problem in a challenging area for the assessment.				

Stage 4	Functional Programming Technology (FUNC)	Progress towards PLO	Students extend their ability to express and to apply theory in practice, and to make effective use of new software tools	Students increase their capacity for effective abstraction and fluency of thought when reasoning about programs and computations	Students become more able to assess and to adopt alternative views and methods in software composition	Students become more able to recognise and to achieve potential applications of novel software technologies				
		By working on (and if applicable, assessed through)	by making practical use of state-of-the-art tools for the evaluation and verification of functional programs.	by solving a series of problems requiring techniques of abstraction and reasoning in the context of recursive structures and functions.	by making practical use of state-of-the-art tools for the evaluation and verification of functional programs.	by carrying out application-based exercises in functional programming.				

Stage 4	Final Year CS Project (PRIN)	Progress towards PLO	Students will be able to contribute in an original way to an established area of research or development, demonstrating a practical understanding of how established techniques of research and enquiry are used to create and interpret knowledge		Students will learn how to apply software and/or hardware engineering principles to deliver working systems in time to answer a project brief, and to ask questions of the project brief or refine it as needed	Students will learn (mainly independently) how to learn, evaluate and apply new techniques and ideas and will learn critical and experimental skills		Students will be able to synthesise and critically expound existing approaches to computational problems, and explain their own approach to such problems and how they have evaluated their own approach, and will be able to tailor their writing and presentation to a general, informed audience succinctly and consistently	Students will gain awareness of issues of ethics and academic integrity in computer science	

		By working on (and if applicable, assessed through)	by conducting supervised individual research which is relevant to the project, where the project topic is either proposed by a supervisor (and often arising from ongoing research work), or self-proposed (perhaps arising from year-in-industry experiences, or individual students' interest).		by working out how to engineer an artefact which meets the requirements of the project brief within the given time frame, and by focussing on usage scenarios and defining a clear sense of the requirements for and application of the product.	by independently defining and tackling a problem in their project that will not be entirely 'covered' in other modules, and by undertaking a targeted search for and review of literature in a given area and considering how to apply/extend it.		by writing a substantial project report, and preparing and delivering a presentation, with both formats being subject to strict length constraints.	by explicitly considering ethical issues when conducting their project work and when writing their project report, and also by undertaking explicit training in academic integrity and the use of plagiarism detection software as a writing / research aid.	