Imperial College London

DEPARTMENTS OF MATHEMATICS AND COMPUTING

MATHEMATICS AND COMPUTER SCIENCE (MSci GG41)

GUIDE TO COURSES

FOURTH/FINAL YEAR 2012—2013

Notes and syllabus details on Fourth Year/level courses

Prof. Jonathan Mestel
Director of Undergraduate Studies
Mathematics

Dr. Jeremy Bradley
Course Director <jmc@doc.ic.ac.uk>
Computing

June 2012 (v2)

The information given is current at this date and may be subject to alteration

This guide gives course information about all courses which have been validated at this level. A selection is made at each session to give students a suitable set of course from which to make their choice.

All Departmental course documents may be viewed via the Departmental undergraduate webpage at:

www3.imperial.ac.uk/mathematics/students/undergraduate

MSci DEGREE PROGRAMME

Each Fourth Year Mathematics and each Fourth Year Computing course carries a half unit of credit. MSci students may normally choose *new courses up to a value of four course units* from the available courses in their Fourth Year of study.

Please note that the choice of options of all students is subject to the approval of the Director of Studies (Mathematics) and Course Director (Computing).

PASSES WILL NORMALLY BE REQUIRED IN ALL COURSE ELEMENTS (HERE MAINLY HALF UNITS) FOR GRADUATION.

MSci students may be awarded an Honours degree classified as either First, Second (upper and lower divisions) or Third. There is no Pass degree category.

In general, applications for postponement of the award of a degree will be granted by the Departments only in special cases; such as absence through illness.

FOURTH YEAR MSci

Students will take 7 half unit courses, including at least 2 from Mathematics and at least 2 from Computing. Students also submit and present their (Third Year) industrial placement work at the beginning of the Autumn Term and submit, at the end of their Summer Term, a compulsory individual project MC4R, either in Mathematics or Computing. The industrial placement and the individual project form together one half-unit with enhanced weighting for Honours.

(M)

The full programme of courses to be available to Third and Fourth Year students in future sessions is under continuous review. Many courses are intended to run every session, but others do tend to run in alternate sessions.

Some courses are available in Fourth Year in M4 "...with Advanced Study" form. The enhancement is in the form of an additional masters level exam question for each Advanced Study option taken on top of the base M3 exam. This enhancement is not part of the M3 assessment, only part of that for M4. Naturally students may not offer such an M4 course if they have previously taken the M3 version.

Some courses are only available at the M4 level. These M4 courses have a strong overlap with the PG taught course MSc courses in Applied Mathematics, Pure Mathematics.

MSci Fourth Year students are normally expected to offer all of their course choices in M4 form. It is expected that, with Departmental permission, there may be dispensation with respect to one M3 course, but only if the corresponding M4 course is not available. Fourth Year students also complete an additional research project in Maths or Computing.

M4 COURSES

Below is a list of suggested Mathematics courses for joint Mathematics and Computing students. If however, a JMC student wishes to take another M4 course and the timetable allows it, they may take that course as one of their half units (please let the JMC Course Director know first). In taking one of these unlisted M4 courses, you may have to do extra preparatory work to cover material that may not have been in J1—J3.

NUMERICAL/APPLIED ANALYSIS	
M4N7	Numerical Solution of Ordinary Differential Equations with Advanced Study
M4N10	Computational Partial Differential Equations I with Advanced Study

PURE MATHEMATICS	
GEOMETRY	
M4P5	Geometry of Curves and Surfaces with Advanced Study
M4P20	Geometry I: Algebraic Curves with Advanced Study
M4P21	Geometry II: Algebraic Topology with Advanced Study
M4P51	Riemannian Geometry
M4P52	Manifolds
M4P54	Differential Topology
M4P57	Complex Manifolds
ALGEBRA AND DISCRETE MATHEMATICS	
M4P8	Algebra III with Advanced Study
M4P10	Group Theory with Advanced Study
M4P11	Galois Theory with Advanced Study
M4P12	Group Representation Theory with Advanced Study
M4P17	Algebraic Combinatorics with Advanced Study
M4P23	Computational Algebra and Geometry with Advanced Study*
M4P34	Groups and Representations
M4P46	Lie Algebras
M4P55	Commutative Algebra
NUMBER THEORY	
M4P14	Elementary Number Theory with Advanced Study
M4P15	Algebraic Number Theory with Advanced Study
M4P16	Analytic Number Theory with Advanced Study
M4P32	Number Theory: Elliptic Curves
OTHER	
M4P6	Probability Theory with Advanced Study*
M4P7	Functional Analysis with Advanced Study*
M4P18	Fourier Analysis and Theory of Distributions with Advanced Study*
M4P19	Measure and Integration with Advanced Study*
M4SC	Scientific Computation
M4T	Communicating Mathematics

STATISTICS	
M4S1	Statistical Theory I with Advanced Study
M4S2	Statistical Modelling II with Advanced Study
M4S4	Applied Probability with Advanced Study
M4S7	Statistical Pattern Recognition with Advanced Study
M4S8	Time Series with Advanced Study
M4S9	Stochastic Simulation with Advanced Study
M4S11	Games, Risks and Decisions
M4S14	Survival Models and Actuarial Applications with Advanced Study
M4S16	Credit Scoring I with Advanced Study
M4S17	Credit Scoring II with Advanced Study
M4A42	Applied Stochastic Processes

C4 COURSES

Students should select Computing courses from the following list. Recommended prerequisites are indicated with the syllabus details in Appendix C below; see Note (1) below.

C320	Complex Systems		Term 1
C330	Network Security	Recommended: Distributed Systems (C335)	Term 1
C332	Advanced Computer Architecture		Term 2
C382	Type Systems for Programming Languages		Term 1
C417	Advanced Graphics and Visualisation	Recommended: Graphics (C317)	Term 2
C418	Computer Vision	Recommended: Graphics (C317)	Term 1
C421	Computational Neurodynamics		Term 1
C422	Computational Finance	Recommended: Operations Research (C343)	Term 2
C424	Machine Learning and Neural Computation		Term 1
C429	Parallel Algorithms	Recommended: Distributed Systems (C335) or Simulation & Modelling (C337)	Term 2
C436	Performance Analysis	Recommended: Simulation & Modelling (C337). Exclusive with M3S4 Applied Probability I	Term 2
C437	Distributed Algorithms	Recommended: Concurrency (C223), Distributed Systems (C335)	Term 2
C438	Complexity	Recommended: Models of Computation (C240) or (M20D)	Term 2
C440	Software Reliability		Term 1
C470	Program Analysis		Term 2
C471	Advanced Issues in Object Oriented Programming	Recommended: Type Systems for Programming Languages (C382)	Term 1
C474	Multi-agent Systems	Recommended: Introduction to AI (C231) & Prolog	Term 2
C475	Software Engineering for Industry		Term 1
C477	Computing for Optimal Decisions	Recommended: Operations Research (C343) or (M3N3)	Term 1
C480	Automated Reasoning	Recommended: Prolog	Term 1
C481	Models of Concurrent Computation	Recommended: Concurrency (C223)	Term 1
C484	Quantum Computing		Term 2
C491	Knowledge Representation		Term 2
C493	Intelligent Data and Probabilistic Inference		Term 2
C499	Modal and Temporal Logic		Term 1

NOTE:

(1) Recommended prerequisites in Computing are given as a guide to the background material required. Without these, in some cases you may need to do some extra background reading in order to fully understand the material. However you may still take a course without sitting the recommended prerequisite.

(2) C317, C395, C436 and C480 may not be taken in both Year 3 and Year 4

(3) C382 cannot be taken in Year 4 if C482 has been taken in Year 3

Some further information is given below. Students may well also find it useful to consult overall and year-specific course documents for other degree codings within the two Departments and the College Calendar.

Students should certainly feel free to seek advice from Personal Tutors and Senior Tutors, together with the Course Directors and Directors of Studies in the two Departments.

OTHER COURSES

Up to ONE course in Humanities or Management may normally be offered in Fourth Year from the list of courses approved by the Departments. For syllabi see the Humanities Programme and Imperial College Business School websites.

H01	Philosophy I	1 + seminars	Terms 1 & 2
H04	Controversies and Ethical Dilemmas in Science and Technology	1 + seminars	Terms 1 & 2
H05	European History : 1870 – 1989	1 + seminars	Terms 1 & 2
H06	Politics	1 + seminars	Terms 1 & 2
H07	History of Science	1 + seminars	Terms 1 & 2
H08	Global History of Twentieth Century Things	1 + seminars	Terms 1 & 2
H09	History of Medicine	1 + seminars	Terms 1 & 2
H12	Music and Western Civilisation	1 + seminars	Terms 1 & 2
H13	Communicating Science: The Public and the Media	1 + seminars	Terms 1 & 2
H15	Humanities Essay	Tutorials + private study	Terms 1 & 2
H19	Creative Writing	1 + tutorials	Terms 1 & 2
H21	Music Technology	2 + private study	Terms 1 & 2
H22	Philosophies of Science: Theory, Society and Communications	1 + seminars	Terms 1 & 2
BS0808	Finance and Financial Management	1 + seminars/tutorials	Term 2
BS0820	Innovation Management	1 + seminars/tutorials	Term 1
BS0815	Managerial Economics	1 + seminars/tutorials	Term 1
BS0821	Project Management	1 + seminars/tutorials	Term 2

NOTE:

- **Humanities courses extend throughout Terms 1 and 2**
- **H15 Humanities Essay normally requires permission from the Humanities Programme and approval by the Department of Mathematics**

STUDENTS ARE ENCOURAGED TO CONSIDER BROADENING THEIR STUDY PROGRAMME BY TAKING ADVANTAGE OF THIS HUMANITIES/MANAGEMENT PROVISION.

It is important to note that places in Humanities/Management courses are normally limited – individual bookings for them should be done separately for these courses on the forms at the Humanities and Business School websites before the end of the previous session.

The Department always endeavours to avoid timetabling Mathematics courses during the allocated 12-2 College slots reserved for the above courses. However, timetabling by the

Humanities or Business School outside these periods cannot be so protected by the Departments.

APPENDIX (M)

Please note that the Department reserves the right to cancel a particular course if the number of students attending that course does not make it viable. Similarly some courses are occasionally run as 'Reading Courses' (with an examination of the usual kind).

NUMERICAL/APPLIED ANALYSIS

M4N7 NUMERICAL SOLUTION OF ORDINARY DIFFERENTIAL EQUATIONS WITH ADVANCED STUDY
Dr. D.R. Moore **Term 1**

An analysis of methods for solving ordinary differential equations. Totally examined by project.

Runge-Kutta, extrapolation and linear multistep methods. Analysis of stability and convergence.
Error estimation and automatic step control. Introduction to stiffness.
Boundary and eigenvalue problems. Solution by shooting and finite difference methods.
Introduction to deferred and defect correction.

Additional material: Contained within the scope of the projects.

M4N10 COMPUTATIONAL PARTIAL DIFFERENTIAL EQUATIONS I WITH ADVANCED STUDY

Dr. Lushi **Term 2**

An introduction to the finite element method for approximating PDEs.

Topics: Variational principles, weak formulation. Essential and natural boundary conditions.
Galerkin method. Conforming elements in one and higher space dimensions.
Approximation and error analysis. Functional analysis setting. Computational aspects.
Variational crimes.
Non-self adjoint problems. Time dependent problems. Stability analysis

Additional material: From topics within the above syllabus.

PURE MATHEMATICS

GEOMETRY

M4P5 GEOMETRY OF CURVES AND SURFACES WITH ADVANCED STUDY

Dr. A. Neves **Term 2**

The main object of this course is to understand what is the curvature of a surface in 3-dimensional space.

Topological Surfaces: Definition of an atlas; the prototype definition of a surface; examples.

The topology of a surface; the Hausdorff condition, the genuine definition of a surface. Orientability, compactness. Subdivisions and the Euler characteristic.
Cut-and-paste technique. The classification of compact surfaces. Connected sums of surfaces.

Smooth Surfaces: Definition of a smooth atlas, a smooth surface and of smooth maps into and out of smooth surfaces. Surfaces in IR^3**,** tangents, normals and orientability.
The first fundamental form, lengths and areas, isometries.
The second fundamental form, principal curvatures and directions.
The definition of a geodesic, existence and uniqueness, geodesics and coordinates.
Gaussian curvature, definition and geometric interpretation, Gauss curvature is intrinsic, surfaces with constant Gauss curvature. The Gauss-Bonnet theorem
(Not examinable and in brief) Abstract Riemannian surfaces, metrics.

Additional material: Mean curvature and minimal surfaces, including the definition of mean curvature, its geometric interpretation, the definition of minimal surfaces and some examples.

M4P20 GEOMETRY I: ALGEBRAIC CURVES WITH ADVANCED STUDY

Dr. Hein **Term 1**

Plane algebraic curves including inflection points, singular and non-singular points, rational parametrisation, Weierstrass form and the Group Law on non-singular cubics. Abstract complex manifolds of dimension 1 (Riemann surfaces); elliptic curves as quotients of C by a lattice. Elliptic integrals and Abel’s theorem.

M4P21 GEOMETRY II: ALGEBRAIC TOPOLOGY WITH ADVANCED STUDY

Dr. Kaloghiros **Term 2**

Homotopies of maps and spaces. Fundamental group. Covering spaces, Van Kampen (only sketch of proof). Homology: singular and simplicial (following Hatcher’s notion of Delta-complex). Mayer-Vietoris (sketch proof) and long exact sequence of a pair. Calculations on topological surfaces. Brouwer fixed point theorem.

M4P51 RIEMANNIAN GEOMETRY

Dr. Muller **Term 2**

Riemannian manifolds, (embedded) surfaces, Gauss map, Gauss-Bonnet.
Riemann curvature. Gauss Theorem Egregium. Levi-Civita. Connections and a little Chern-Weil theory.

M4P52 MANIFOLDS

Dr. M. Kool **Term 1**

Smooth manifolds, tangent spaces, differential forms. Stokes’ theorem, de Rham cohomology, Mayer-Vietoris for de Rham cohomology, de Rham pairing between singular homology and de Rham cohomology.
(Emphasis on examples: e.g. Hopf fibration, linking numbers of curves via differential forms)

M4P54 DIFFERENTIAL TOPOLOGY
[Prerequisites: Algebraic Topology (M3P21) and M4P52 Manifolds (M4P52)]

Dr. J. Nordstrom **Term 2**

Advanced topics in algebraic topology and manifold theory, from: Mayer-Vietoris exact sequence, Hodge theory, Morse theory, Poincare duality.

M4P57 COMPLEX MANIFOLDS

Dr. J. Nordstrom **Term 2**

Advanced topics in algebraic topology and manifold theory, from: Mayer-Vietoris exact sequence, Hodge theory, Morse theory, Poincare duality.

ALGEBRA AND DISCRETE MATHEMATICS

M4P8 ALGEBRA III WITH ADVANCED STUDY

Professor K. Buzzard **Term 1**

Rings, integral domains, unique factorization domains.
Modules, ideals homomorphisms, quotient rings, submodules quotient modules
Fields, maximal ideals, prime ideals, principal ideal domains
Euclidean domains, rings of polynomials, Gauss's lemma, Eisenstein's criterion
Field extensions
Noetherian rings and Hilbert's basis theorem
Dual vector space, tensor algebra and Hom
Basics of homological algebra, complexes and exact sequences

Additional material: From topics within the above syllabus.

M4P10 GROUP THEORY WITH ADVANCED STUDY

Dr. Britnell **Term 1**

An introduction to some of the more advanced topics in the theory of groups.

Composition series, Jordan-Hölder theorem, Sylow's theorems, nilpotent and soluble groups.
Permutation groups. Types of simple groups.
Additional material: Automorphisms. Free groups, Generators and relations. Free products.

M4P11 GALOIS THEORY WITH ADVANCED STUDY

Professor M.W. Liebeck **Term 1**

The formula for the solution to a quadratic equation is well-known. There are similar formulae for cubic and quartic equations, but no formula is possible for quintics. The course explains why this happens.

Irreducible polynomials. Field extensions, degrees and the tower law. Extending isomorphisms.
Normal field extensions, splitting fields, separable extensions. The theorem of the primitive element.
Groups of automorphisms, fixed fields. The fundamental theorem of Galois theory.
The solubility of polynomials of degree at most 4. The insolubility of quintic equations.

Additional material: Kummer theory, solubility of polynomials with soluble Galois groups. Construction of a radical solution for a soluble polynomial. Tests for solubility of quintics.

M4P12 GROUP REPRESENTATION THEORY WITH ADVANCED STUDY

Dr. E. Segal **Term 2**

Although the underlying theory in this course concerns the ways in which a group can be written as a set of matrices, the main results involve group characters.

Some elementary group theory; examples of groups; conjugacy classes; Cayley's theorem.
Representations of groups: definitions and basic properties. Maschke's theorem, Schur's lemma.
Representations of Abelian groups. The character of a group representation. The group algebra and the regular character. Orthogonality relations.
Character tables of various interesting groups.
Construction of representations by using tensor products.

Additional material: Induced representations and characters. The Frobenius Reciprocity theorem.

M4P17 ALGEBRAIC COMBINATORICS WITH ADVANCED STUDY

Professor A.A. Ivanov **Term 1**

An introduction to a variety of combinatorial techniques which have wide applications to other areas of mathematics.

Elementary coding theory. The Hamming metric, linear codes and Hamming codes.
Combinatorial structures: block designs, affine and projective planes. Construction of examples using finite fields and vector spaces. Steiner systems from the Golay code. Basic theory of incidence matrices.
Strongly regular graphs: examples, basic theory, and relationship with codes and designs.

Additional material: The Mathieu group and their relationship with codes and strongly regular graphs.

M4P23 COMPUTATIONAL ALGEBRA AND GEOMETRY WITH ADVANCED STUDY

Dr. Kasprzyk **Term 1**

Topics in abstract algebra, approached from the perspective of Gröbner bases and computational algebra software (Macaulay 2, Magma, GAP, SAGE). Hilbert Basis Theorem. Buchberger's algorithm. Elimination theory and resultants. Invariant theory of finite groups.

M4P34 GROUPS AND REPRESENTATIONS

Professor A.A. Ivanov **Term 1**

Variational calculus on tangent spaces of Lie groups is explained in the context of familiar concrete examples. Through these examples, the student will develop skills in performing computational manipulations, starting from vectors and matrices, working through the theory of quaternions to understand rotations, then transferring these skills to the computation of more abstract adjoint and coadjoint motions, formulations of Euler-Poincare dynamics on the Lagrangian side and Lie-Poisson formulations on the Hamiltonian side, momentum maps and finally the dynamics with nonholonomic contraints, such as the off-centre cue ball and the falling, rolling, spinning coin.

M4P46 LIE ALGEBRAS

Prof. Liebeck **Term 2**

The semisimple complex Lie algebras: root systems, Weyl groups, Dynkin diagrams, classification. Cartan and Borel subalgebras. Classification of irreducible representations.

M4P55 COMMUTATIVE ALGEBRA

Dr. A. Pal **Term 1**

Prime and maximal ideals, nilradical, Jacobson radical, localization. Modules. Primary decomposition of ideals. Applications to rings of regular functions of affine algebraic varieties. Artinian and Noetherian rings, discrete valuation rings, Dedekind domains. Krull dimension, transcendence degree. Completions and local rings. Graded rings and their Poincare series.

NUMBER THEORY

M4P14 NUMBER THEORY WITH ADVANCED STUDY

Dr. P. Cascini **Term 1**

The course is concerned with properties of natural numbers, and in particular of prime numbers, which can be proved by elementary methods.

Division and factorisation, fundamental theorem of arithmetic, Euclid's algorithm.
Arithmetic functions, multiplicative functions, perfect numbers, Möbius inversion, Dirichlet convolution.
Congruences, Fermat-Euler theorem, Chinese remainder theorem, Lagrange's theorem. Wilson's theorem.
Primitive roots, Gauss's theorem, indices.
Quadratic residues, Euler's criterion , Gauss's lemma, law of quadratic reciprocity, Jacobi symbol.
Sums of squares.
Distribution of quadratic residues and non-residues.
Irrationality, Liouville's theorem, construction of a transcendental number.

Additional material: Order of magnitude of arithmetic functions, square-free numbers, lattice points on a disc. Farey sequences, continued fractions, approximation by convergents, Hurwitz's theorem.

M4P15 ALGEBRAIC NUMBER THEORY WITH ADVANCED STUDY

Dr. Pal **Term 2**

An introduction to algebraic number theory, with emphasis on quadratic fields. In such fields the familiar unique factorisation enjoyed by the integers may fail, but the extent of the failure is measured by the class group.
The following topics will be treated with an emphasis on quadratic fields :
Field extensions, minimum polynomial, algebraic numbers, conjugates and discriminants, Gaussian integers, algebraic integers, integral basis, quadratic fields, cyclotomic fields, norm of an algebraic number, existence of factorisation.
Factorisation in quadratic field Ideals, Z-basis, maximal ideals, prime ideals, unique factorisation theorem of ideals and consequences, relationship between factorisation of numbers and of ideals, norm of an ideal.
Ideal classes, finiteness of class number, computations of class number.

Additional material: The topics of M3P15, treated for general number fields, together with Fractional ideals, Minkowski's theorem on linear forms, Ramification, characterisation of units of cyclotomic fields, a special case of Fermat's last theorem.

M4P16 ANALYTIC NUMBER THEORY

Professor N. Bingham **Term 2**

Uniform distribution, Weyl's criterion, the sequence $\{na\}$.
Euler's proof that there are infinitely many primes. Von Mangoldt's function, Chebychev's Inequality, Merten's theorem.
Dirichlet series, generating functions.
Gauss sums, second proof of law of quadratic reciprocity.
Dirichlet characters. Dirichlet's theorem on primes in arithmetic progression.
Selberg's upper bound sieve. Prime number theorem.

Additional material: Kronecke's theorem, Bertrand's postulate, Number of representations as sum of squares, proof of three square theorem.

M4P32 NUMBER THEORY: ELLIPTIC CURVES

Dr. T. Gee **Term 1**

The p-adic numbers. Curves of genus 0 over $\mathbf{Q}$. Cubic curves and curves of genus 1. The group law on a cubic curve. Elliptic curves over p-adic fields and over $\mathbf{Q}$. Torsion points and reduction mod p. The weak Mordell-Weil theorem. The Schaferevich-Tate group. Heights. The (full) Mordell-Weil theorem. Relation of elliptic curves to Fermat's Last Theorem.

OTHER

M4P6 PROBABILITY WITH ADVANCED STUDY

Prof. B. Zegarlinski **Term 2**

A rigorous approach to the fundamental properties of probability.
Probability measures. Random variables Independence. Sums of independent random variables; weak and strong laws of large numbers. Weak convergence, characteristic functions, central limit theorem. Elements of Brownian motion.
Additional material: From Brownian motion and Martingales.

M4P7 FUNCTIONAL ANALYSIS WITH ADVANCED STUDY

Prof. A. Laptev **Term 1**

This course brings together ideas of continuity and linear algebra. It concerns vector spaces with a distance, and involves linear maps; the vector spaces are often spaces of functions.
Vector spaces. Existence of a Hamel basis. Normed vector spaces. Banach spaces. Finite dimensional spaces. Isomorphism. Separability. The Hilbert space. The Riesz-Fisher Theorem. The Hahn-Banach Theorem. Principle of Uniform Boundedness. Dual spaces. Operators, compact operators. Hermitian operators and the Spectral Theorem.
Additional material: Extension of M3P7 topics. Banach algebras.

M4P18 FOURIER ANALYSIS AND THEORY OF DISTRIBUTIONS WITH ADV. STUDY

Dr. J.-C. Cuenin **Term 2**

Spaces of test functions and distributions, Fourier Transform (discrete and continuous), Bessel's, Parseval's Theorems, Laplace transform of a distribution, Solution of classical PDE's via Fourier transform, Basic Sobolev Inequalities, Sobolev spaces. Additional material: Extension of topics in above syllabus.

M4P19 MEASURE AND INTEGRATION

Dr. I. Krasovsky Term 1

Lebesgue theory of measure of integration. Outer measure, Basic theorems of integration theory. Integration versus differentiation. Fubini-Tonelli Theorem and applications, Convergence almost everywhere. Borel-Cantelli Lemma, L_p spaces. Ergodic Theorem. Additional material: Entension of topics in above syllabus.

M4T COMMUNICATING MATHEMATICS

Dr. E. McCoy/Dr. L.V. White Term 2/3

This course will give students the opportunity to observe and assist with teaching of Mathematics in local schools. Entry to the course is by interview in the preceding June and numbers will be limited. For those selected there will follow a one-day training course in presentation skills and other aspects of teaching. Students will be assigned to a school where they will spend 10 half-days in the Spring Term, under the supervision of a teacher. Assessment will be based on a portfolio of activities in the school, a special project, evaluation by the school teacher and an oral presentation.

M4SC SCIENTIFIC COMPUTATION

Dr D.R. Moore Term 2

Computer programming in an important tool for any mathematician. The aim of this course is to develop proficiency in a scientific programming language and to solve mathematical problems using computation. The objectives are that by the end of the course all students should be familiar with the essential elements of the C programming language, and be able to undertake substantial programming tasks in C, examples of which are given below.

The elements of the language to be covered will include:

Running a C program on the computers in Mathematics. Basic C data types and data declarations. Arithmetic with C data types. Simple I/O in C. C control structures. Arrays and matrices. Pointers and dynamic allocation. Structures and linked lists. File handling.
Examples of programming tasks to be undertaken:

Simple numerical tasks (e.g. recursion, numerical integration, differential equations). Data processing (e.g. sorting algorithms, Fast Fourier Transforms). More computational tasks (e.g. iterative methods for linear and non-linear equations). More data processing (e.g. voting algorithms, triangulation). Mastery Material: Extension within the scope of the projects.

STATISTICS

M4S1 STATISTICAL THEORY I WITH ADVANCED STUDY

Dr. R. Coleman Term 1

Theories Of Estimation And Hypothesis-Testing, Including Sufficiency, Completeness, Exponential Families, Cramer-Rao, Blackwell-Rao And Neyman-Pearson Results. Elementary Decision Theory.
Additional material: Extending some of the above topics.

M4S2 STATISTICAL MODELLING II WITH ADVANCED STUDY

Dr. C. Anagnostopoulos **Term 2**

This course leads on from the linear models covered in M2S2 and Probability and Statistics covered in M2S1. The Generalised Linear Model is introduced from a theoretical and practical viewpoint and various aspects are explained.

Generalised Linear Model, as a unifying statistical framework – linear models and quantitative responses. Nonlinear regression. Generalised Additive Models, Kernel Regression. Fixed and random effect models.

The R statistical package will be used to expose how the different models can be applied on example data.

Additional material: Within the above syllabus and assignments.

M4S3 STATISTICAL THEORY II WITH ADVANCED STUDY

Extension and enhancement of statistical theory (Year 2 course) to more advanced level.

Asymptotic and Large Sample Theory in Statistics
Central Limit Theorems and Asymptotic Normality
Advanced Hypothesis Testing and Methods of Inference
Extensions to Likelihood Theory: Profile, Adjusted Profile, Partial, Quasi
Expansions. Non-parametric statistics, Robust Statistics. Influence.
Empirical Processes. Advanced Bayesian Theory.
Additional material : To study an area/a topic identified by the lecturer of technical interest relating to, but extending the M3 syllabus material (e.g. infinitely divisible distributions.
Non-parametrics/Bayesian Non-Parametrics, Empirical Processes).

M4S4 APPLIED PROBABILITY WITH ADVANCED STUDY

Dr. A. Veraart **Term 1**

This course aims to give students an understanding of the basics of stochastic processes. The theory of different kinds of processes will be described, and will be illustrated by applications in several areas. The groundwork will be laid for further deep work, especially in such areas as genetics, finance, industrial applications, and medicine.

Revision of basic ideas of probability. Important discrete and continuous probability distributions. Random processes: Bernoulli processes, simple birth processes, branching processing. Point processes. Stationarity. Non-homogeneous, compund, and doubly stochastic Poisson processes. Autocorrelation functions. Perturbed regular processes. Probability generating functions and how to use them. Branching processes and their properties. Galton-Watson model. Random walks. Gambler's ruin. Absorbing and reflecting barriers. Markov chains. Chapman-Kolmogorov equations. Recurrent, transient, periodic, aperiodic chains. Returning probabilities and times. The basic limit theorem. Birth processes.
Differential difference equations and pgfs. Finding pgfs. Birth, death, immigration processes. Embedded processes. Time to extinction. Queues. Specification. Long term behaviour. Traffic intensity. Waiting times. Steady states.

Additional material: e.g. Martingales.

M4S7 STATISTICAL PATTERN RECOGNITION WITH ADVANCED STUDY

Dr. Cohen **Term 1**

A fundamental aim in statistics is classifying things. For example, this occurs in medicine, where the aim is to assign people to diseases, speech recognition, where the aim is to assign words to meanings, banking, where the aim is to assign customers to risk classes, and in a vast number of other areas. This course provides a modern view of methods for performing such so-called pattern recognition tasks.

Totally examined by projects.

Discriminant analysis, generalised linear models, nearest neighbour methods, recursive partitioning methods, trees, ensemble classifiers, and other tools, The areas of pre-processing, feature selection and classification performance assessment will also be explored, and an introduction to unsupervised pattern recognition methods will be given.

Additional material: Will focus on an area of classification not covered in detail in the course. Advanced study will be assessed by extra project material.

M4S8 TIME SERIES WITH ADVANCED STUDY

Professor A.T. Walden **Term 1**

An introduction to the analysis of time series (series of observations, usually evolving in time) is given which gives weight to both the time domain and frequency domain viewpoints. Important structural features (e.g. reversibility) are discussed, and useful computational algorithms and approaches are introduced. The course is self-contained.

Discrete time stochastic processes and examples. ARMA processes. Trend removal and seasonal adjustment. General linear process. Invertibility. Directionality and reversibility in time series. Spectral representation. Aliasing. Generating functions. Estimation of mean and autocovariance sequence. The periodogram. Tapering for bias reduction. Parametric model fitting. Forecasting.

Additional material: From long-memory processes, Autoregressive parametric spectrum estimation, Harmonic analysis, Multichannel time series modelling and analysis.

M4S9 STOCHASTIC SIMULATION WITH ADVANCED STUDY

Professor Bottolo **Term 2**

Computational techniques are becoming increasingly important in statistics, especially in the simulation of stochastic systems (for example for testing new estimation methods). This course provides a modern view of such simulation methods covering areas from basic random variate generation to ways of describing images. Material from M2S1 would form a firm foundation.

Random numbers. Congruential and other generators. Techniques of random variate generation. Computer experiments. Monte Carlo integration. Variance reduction techniques. Markov random fields. Gibbs sampling. Applications in statistics, optimisation and operations research.

Additional material: From Proofs for period lengths of congruential generators, Theory of Wichmann-Hill generator, Markov chain Monte Carlo methods, Applications: Monte Carlo tests, bootstrapping , appropriateness of asymptotics; queues.

M4S11 GAMES, RISKS AND DECISIONS WITH ADVANCED STUDY

Dr. L. White **Term 2**

Simple probabilistic and mathematical tools are used to study the theory of games where two opponents are in conflict, including the celebrated Prisoners' Dilemma problem. Utilities, based on (apparently) reasonable axioms are introduced, leading to a study of decision theory.

Game Theory. Randomized strategies. Admissibility. Bayes strategies. Utility theory. Risks and risk aversion. Decision Theory. Sequential Decision-making. Applications to problems in Economics, Biology, Medicine, Insurance and Investment.

Additional material: More on Non-zero sum two-person games, Sequences of decisions and decision trees; the theory of preferences and utilities, Uninformative Bayes priors.

M4S14 SURVIVAL MODELS AND ACTUARIAL APPLICATIONS WITH ADVANCED STUDY

Dr. N.A. Heard **Term 2**

Survival models are fundamental to actuarial work, as well as being a key concept in medical statistics. This course will introduce the ideas, placing particular emphasis on actuarial applications.

Explain concepts of survival models, right and left censored and randomly censored data. Describe estimation procedures for lifetime distributions: empirical survival functions, Kaplan-Meier estimates, Cox model.

Statistical models of transfers between multiple states, maximum likelihood estimators. Binomial model of mortality. Estimation of transition intensities which depend on age. Testing crude estimates for consistency. Discussion of simple assurance and annuity contracts. Computation of expected present values and variances of simple benefits. Description and calculation of net premiums and net premium policy values of simple

Additional material: Extension of one or more of the above topics.

M4S16 CREDIT SCORING I WITH ADVANCED STUDY

Dr. T. Bellotti **Term 1**

Introduction and background: Aims and objectives of scoring, legislative and commercial aspects.
Consumer credit data: characteristics, transformations, data quality, transaction types, challenges. Notions of statistical scorecards
Basic models: application and behavioural model types and characteristics, including segmented, dynamic, and stochastic process models
Estimation and model fitting: basic principles and model selection. Issues of model search space and selectivity bias.
Special models and methods: tailored methods for specific problems, such as existing model favouritism and the needle in the haystack problem in fraud detection
Performance assessment and monitoring: measures and estimation, method comparison

Additional material: Within the above topics and topics from fraud scoring, trigger and change analysis, reject inference.

M4S17 CREDIT SCORING II WITH ADVANCED STUDY

Dr. T. Bellotti **Term 2**

Syllabus not available.

M4A42 APPLIED STOCHASTIC PROCESSES

Dr. Pavliotis **Term 2**

Elements of probability theory.

1. Stochastic processes: basic definitions, examples.
2. Discrete time Markov processes: Markov chains.
3. Continuous time Markov processes.
    - Example: Brownian motion.
    - Diffusion processes: basic definitions, the generator.
    - Backward Kolmogorov and the Fokker-Planck (forward Kolmogorov) equations.
    - Stochastic differential equations (SDEs); Ito calculus, Ito and Stratonovich stochastic integrals, connection between SDEs and the Fokker-Planck equation.
4. Applications.
    - Bistability, metastability and exit problems.
    - Stochastic mode elimination.
    - Spatially extended systems.
    - Markov Chain Monte Carlo (MCMC).
    - Numerical methods for SDEs.
    - Statistical mechanics (Mori-Zwanzig formalism, convergence to equilibrium etc).

APPENDIX (C)

C317 GRAPHICS

Dr. P. Aljabar/Professor D. Gillies/Professor D. Ruckert Term 2

Syllabus
This course covers the fundamental principles of computer graphics and their use in prominent applications. The lectures include:
Device independent graphics: Raster and vector devices, world coordinates, the normalisation transformation, output primitives, input primitives. Polygon rendering: 3D data base representation, projection onto a viewing surface, transformation of graphical scenes, homogenous co-ordinates, affine transformations for animation. 3D geometry:
clipping and containment in 3D convex objects, splitting concave objects. Texture mapping and anti-aliasing. Shading planar polygons:
Gouraud shading, Phong shading. Representing Colours: tri-stimulus model, RGB model, CIE definition space, perceptual colour spaces. Ray Tracing: Ray/object intersection calculations, secondary rays, shadows, reflection and refraction, object space coherence, ray space coherence.
Radiosity: Modelling ambient light, form factors, specular effects, shooting patches, computational efficiency. Geometric Warping and Morphing. Special Visual Effects: particle systems for fire smoke and water, inverse kinematics in animation, non-photorealistic rendering.

C320 COMPLEX SYSTEMS

Professor A. Edalat Term 1

Contents
In the course of the past decade complexity has become a major scientific field in its own right, which is sometimes considered the ultimate in interdisciplinary fields (EPSRC has named complexity among the five priority research areas along with energy, information communication technologies, nano-technology and health care). It relies on concepts in mathematics, physics and computer science on the one hand, and the rapid growth in computational power and information processing capabilities on the other.

This course aims to introduce students to this exciting new field, underlining its theoretical underpinnings, while engaging them in hands-on laboratory exercises in order to develop their practical skills and emphasise the applicability of this subject in the modern world.

Learning Outcomes
By the end of the course the students will be able to:

- Understand the theoretical foundations of complexity: mathematical, computational and information-theoretic;
- Reason about complexity with fluency and using scientific terminology; appreciate it as a post-Newtonian paradigm for approaching a large body of phenomena;
- Apply the terminology and methodology of nonlinear dynamics and chaos to study the time-evolution of complex systems;
- Describe some of the well-known models of complex systems, such as artificial life and Self-Organised Criticality (SOC);
- Construct new variations on models for complex systems using the modelling techniques taught in this course;
- Implement these models elegantly and efficiently using Java or MATLAB;
- Appreciate the interdisciplinary nature of complexity and the pivotal role that computer scientists play in its study.

Syllabus

- Complexity theory: definition of complex systems, models, and dynamical systems; examples of complex systems (e.g. neural networks; traffic flow; artificial life; financial markets and human societies).
- Fundamental theoretical concepts: information, computational complexity, Kolmogorov complexity.
- Discrete dynamical systems: iterative maps; orbits; fixed and periodic points; graphical analysis; phase portraits; bifurcations; chaos. The quadratic family as an illustration of the phenomenology of dynamical systems.
- Cellular automata: cellular automaton rules; number-conserving cellular automata; generalised cellular automata. Cellular automata as models of complex systems: artificial life; stock market dynamics (e.g. Sornette's model of market fluctuations); sandpile model; percolation (e.g. forest fires).
- Complex networks: the small-world phenomenon; graphs; random networks; small-world networks; scale-free networks. WWW as a complex network.
- Hopfield model: associative memories; discrete Hopfield model; applications.
- Game theory: theory of choice; representations of games; repeated games. Economic applications of game theory. Competition and cooperation: stag hunt and prisoner's dilemma; repeated prisoner's dilemma.

C330 NETWORK SECURITY
[Recommended: Distributed Systems (C335)]

Dr. M. Huth/ Dr. E. Lupu ***Term 1***

Aims
To survey the principles and practice of network security. The emphasis of the course is on the underlying principles and techniques of network security with examples of how they are applied in practice.

Learning Outcomes
At the end of the course, a student will have an understanding of: the themes and challenges of network security, the role of cryptography, the techniques for access control and intrusion detection, the current state of the art. The student will have developed a critical approach to the analysis of network security, and will be able to bring this approach to bear on future decisions regarding network security. Practical skills will include the implementation of a security protocol.

Outline Syllabus

Introduction: assets, threats, countermeasures; network security models, security functions: confidentiality, authentication, integrity, nonrepudiation, access control, availability, passive and active attacks, end-to-end vs link-to-link encryption.

Classical Cryptography: key ideas, steganography, codes, one-time pad, substitution and transposition ciphers, cryptanalysis, cryptographic strength.

Symmetric-Key cryptography: Feistel cipher; DES: basics, rounds, e-box, s-box, p-box, key box; Modes of Operation: ECB, CBC, CFB, OFB; Double DES, Triple DES, IDEA, RC5, AES, problems with symmetric key cryptography.

Public-Key cryptography: requirements, confidentiality, authentication, modular arithmetic, Diffie-Hellman key exchange, RSA, attacks against RSA, hybrid cryptosystems, Elliptical Curve, Quantum Cryptography.

Digital Signatures: characteristics, MAC's, one-way hash functions, signing and verification, birthday attack, public-key certificates, disavowed signatures, arbitrated digital signatures, chaffing & winnowing.

Mutual Authentication: basics, replay attacks, man-in-the-middle, interlock protocol, Andrew Secure RPC, Needham Schroeder, Wide-Mouth Frog, Neuman-Stubblebine, Woo-Lam.

BAN-Logic

Key Management: distribution, KDC, announcements and directories, public key certificates, X509 certification authorities, PGP web of trust, control vectors, key generation and destruction, key backup.

Intruders and Programmed Threats: host access, password systems and attacks, one-time passwords, token cards, biometrics, trapdoors, programmed threats: trapdoors, logic bombs, trojan horses, viruses, worms, countermeasures, intrusion-direction.

Firewalls: internet security policies, firewall design goals, firewall controls, TCP/IP, packet filtering routers, application-level gateways, circuit-level gateways, firewall architectures, VPNs.

Web Security: WWW, web servers, CGI, active content, Java applets, Java security model: sandbox, class loaders, bytecode verification, security manager, Java attacks, bypassing Java, mobile code cryptography.

C332 ADVANCED COMPUTER ARCHITECTURE

Professor P. Kelly ***Term 2***

This is a third-level course that aims to develop a thorough understanding of high-performance and energy-efficient computer architecture, as a basis for informed software performance engineering and as a foundation for advanced work in computer architecture, compiler design, operating systems and parallel processing.

Topics include:

- Pipelined CPU architecture. Instruction set design and pipeline structure. Dynamic scheduling using scoreboarding and Tomasulo's algorithm; register renaming. Software instruction scheduling and software pipelining. Superscalar and long-instruction-word architectures (VLIW, EPIC and Itanium). Branch prediction and speculative execution.
- Simultaneous multithreading ('hyperthreading'), and vector instruction sets (such as SSE and AVX).
- Caches: associativity, allocation and replacement policies, sub-block placement. Multilevel caches, multilevel inclusion. Cache performance issues. Uniprocessor cache coherency issues: self-modifying code, peripherals, address translation.
- Dependence in loop-based programs; dependence analysis, and iteration-space transformations - to enable automatic parallelisation, vectorisation (eg for SSE), and for memory hierarchy optimisations such as tiling.
- Implementations of shared memory: the cache coherency problem. Update vs invalidation. The bus-based 'snooping' protocol design space.
- Scalable shared memory using directory-based cache coherency. How shared memory supports programmability; OpenMP and MPI.

- Graphics processors and 'manycore' architectures: SIMT ('single instruction multiple thread), and the CUDA and OpenCL programming models. Decoupling, latency tolerance, throughput-intensive memory system architecture.

Further details are available from the course web site: http://tinyurl.com/kelly-c332

C382 TYPE SYSTEMS FOR PROGRAMMING LANGUAGES

Dr. S. van Bakel ***Term 1***

Contents

The course sets of with the presentation of the Lambda Calculus, on which most functional languages are based. We will introduce the Curry Type Assignment System for this calculus, for which we show that types are respected by reduction, and there exists a notion of principality on types, i.e. all types for a program can be constructed from the 'principal' type.

We will then focus on how to extend the language and the system of types in order to deal with Polymorphism. To this end, Combinator Systems are introduced as a slight extension of LC. In order to study how to deal with Recursion, the language of CS is extended to allow for recursive definitions, and we will discuss ways of typing this extension.

We will then look at Milner's basic ML system, and show that, in this calculus, a solution is present for all the issues discussed separately before. A disadvantage of ML is that it is, syntactically, very distant from actual programming languages, and it is not easy to see that all properties shown for ML translate to the 'real' languages. In particular, definitions of functions by 'cases' (Pattern Matching) are not present in ML. In order to come to a formal notion of types that relates more closely to actual programming languages, we introduce Term Rewriting Systems, which are much closer to actual programs, and show that we can obtain much of the desired properties. To conclude, we will have a brief visit to Intersection Type Assignment, which allows to prove strong properties. The aim of this course is to lay out in detail the design of type assignment systems for programming languages and focus on the importance of a sound theoretical framework, in order to be able to reason about the properties of a typed program. Students will study and compare a variety of systems and languages.

Lambda Calculus: terms, abstraction, application, reduction, normal form, normalisation, head normal form, head normalisation. The Curry type assignment System.

Types, type assignment rules, type substitution, unification, subject reduction. The Principle type for Curry's System.

Recursion and Polymorphism: the need for a recursor, how to type recursion, Milner's let, the basic ML system, the principal type property for the ML system.

Church's typed lambda calculus: type checking versus type inference.

The Polymorphic Lambda calculus. Strong normalisation for the polymorphic Lambda Calculus.

Intersection type assignment: types, type assignment rules, subject expansion, undecidability, filter semantics.

Patterns: term rewriting systems, weak reduction, normal forms, strong normalisation. Dealing with patterns, recursion, subject expansion and reduction.

A decidable restriction of the intersection system.

C395 MACHINE LEARNING

[Recommended: Introduction to Artificial Intelligence (C231)]

Professor S. Muggleton/Mr. L. Dickens/Professor M. Pantic Term 2

Machine Learning (course 395) is envisioned to be an introductory course for several groups of students including MSc Advanced Computing students, Fourth Year Information Systems Engineering students and Third/Fourth Year Mathematics and Computer Science students

Course Aims

Students should be familiar with some of the foundations of the Machine Learning (ML), students should have an understanding of the basic ML concepts and techniques:

- Concept Learning (to be taught by Maja Pantic)
- Decision Trees (to be taught by Maja Pantic)
- Artificial Neural Networks (to be taught by Maja Pantic)
- Instance Based Learning (to be taught by Maja Pantic)
- A Genetic Algorithms (to be taught by Maja Pantic)
- Evaluating Hypotheses (to be taught by Maja Pantic)
- Inductive Logic Programming (to be taught by Mr. L. Dickens)

Students should gain programming skills using Matlab with an emphasis on ML and they should learn how to design, implement and test ML systems. Students should enhance their skills in project planning, working with dead-lines, and reflecting on their own involvement in the teamwork

Course Material

Book: Machine Learning by Tom Mitchell, McGraw-Hill Press, 1997 (chapters: 1-5, 8, 9) Computer Based Coursework (CBC) Manual

Syllabus on Case-based Reasoning

Lecture notes on Inductive Logic Programming (pdf file)

Course Schedule

The curriculum schedules 18 class meetings of one hour each. The CBC for this course will mainly be devoted to coursework (+/- 80 hours per group of 4-5 students).

Attendance to the CBC is mandatory and accounts for 10% of the final grade for the Machine Learning Course. Students are allowed to miss at most 2 hours. Failure to attend due to medical reasons requires a letter from your GP.

Computer Based Coursework (CBC)

CBC CONTENT:

The project is designed to build on lectures by teaching students how to apply ML techniques about which they have been lectured to real-world problems.

The project will consist of five assignments. The first four assignments will focus on emotion recognition from data on displayed facial expression using decision trees, neural networks, case-based reasoning, and genetic algorithms. The last assignment will focus on evaluating (by means of paired t tests and ANOVA test) which of these ML techniques is more suitable for the problem in question in the case of clean data and in the case of noisy data.

CBC ASSESSMENT:

Assessment of the project work will be conducted based upon the following:

- the quality of the delivered code as measured by the clarity, effectiveness and efficacy of the delivered code when tested on real (previously unseen) data,
- the quality of the delivered project report (+/- 2 pages) as measured by the correctness, depth and breadth of the provided discussion on the evaluation of the performance of the developed ML systems for emotion recognition,
- personal involvement and contribution to the group’s results (to be judged based

upon a final interview with each of the groups).

C417 ADVANCED GRAPHICS AND VISUALISATION
[Recommended: Graphics (C317)]

Dr. E. Edwards/Professor D. Rueckert ***Term 2***

Aims
To introduce the students to modern techniques in virtual reality, augmented reality and visualisation and their application to medicine and engineering.

Learning Outcomes
At the end of the course the successful student will: (1) understand the mathematical foundations of visualisation; (2) Understand the perceptual aspects of visualisation (3) understand the principles of visual simulation, in particular in medicine (4) understand the properties and limitation of existing graphics software and systems (5) understand the current research issues in visualisation (6) program new applications for virtual and augmented reality (7) apply visualisation techniques to scientific data (8) make effective use of current software and hardware systems (9) reading and understanding research literature (10) programming scientific applications (11) use of mathematical techniques.

Syllabus
1) Principals of visualisation: Fundamentals and concepts, scalar, vector and tensor data 2) Principles of visualisation: Scalar, vector and tensor data, applications such flow visualisation 3) Volume Rendering: Image-order rendering, Object-order rendering 4) Volume Rendering: scalar and ray transfer functions 5) Volume Rendering: Isosurface generation, marching cubes algorithm 6) Surface reconstruction 1: Surface decimation, surface smoothing, surface normal generation 7) Surface reconstruction 2: Surface triangulation, Voronoi diagrams, Delaunay triangulation 8) Spline Curves 1: Parametric and non-parametric splines, cubic spline patches. 9) Spline Curves 2: Bezier Curves, B-Spline formulation 10) Surface modelling 1: Coons patches, bi-cubic surfaces 11) Surface modelling 2: B spline surfaces, NURBS, rendering spline surfaces 12) Implicit Surface models, soft objects 13) Image-based rendering and lightfields 14) Virtual Reality: Stereo perception, stereo display, head-mounted displays, autostereoscoptic displays, holographic displays 15) Virtual Reality: haptic and tactile force feedback 16) Virtual Reality: virtual worlds, collision detection for VR 17) Augmented Reality: definitions and examples 18) Augmented Reality: calibration and tracking 19) Simulation training in medicine 1 20) Simulation training in medicine 2

C418 COMPUTER VISION
[Recommended: Graphics (C317)]

Professor G-Z. Yang ***Term 1***

Objectives
To introduce the concepts behind computer-based recognition and extraction of features from raster images. To illustrate some successful applications of vision systems and their limitations

Learning Outcomes
By the end of this course, students will be able to explain the concepts behind computer-based recognition and the extraction of features from raster images. Students will also be able to illustrate some successful applications of vision systems and will be able to identify the vision systems limitations.

Syllabus
Overview of early, intermediate and high level vision: first and second moments of illumination for recognition and classification of machine shop components in silhouette. Segmentation: region splitting and merging; quadtree structures for segmentation; mean and variance pyramids; computing the first and second derivatives of images using the

isotropic, Sobel and Laplacian operators; grouping edge points into straight lines by means of the Hough transform; limitations of the Hough transform; parameterisation of conic sections. Perceptual grouping: failure of the Hough transform; perceptual criteria; improved Hough transform with perceptual features; grouping line segments into curves.
Overview of mammalian vision: experimental results of Hubel and Weisel; analogy to edge point detection and Hough transform; neural networks based on the mammalian vision system.
Relaxation labelling of images: detection of image features; simulated annealing.
Grouping of contours and straight lines into higher order features such as vertices and facets. Depth measurement in images; triangulation; projected grid methods; shape from shading based on multi-source illumination.
Matching of images: the correspondence problem for stereo vision; two camera and multiple camera systems; shape from motion as a further stage of stereo vision; optical flow between adjacent video frames.
Expert system modelling in computer vision: model based vision using inference engines and rules.

C421 COMPUTATIONAL NEURODYNAMICS

Prof. M. Shanahan **Term 1**

Overview

Computational neurodynamics is the use of computer models to study the dynamics of large networks of interacting neurons. The rationale behind the field, which lies at the theoretical end of computational neuroscience, is that the language of dynamical systems is the right one to express the underlying principles of nervous system operation. The course has two parts. In Part One, the student will learn how to model single neurons mathematically, how to simulate them computationally, and how to construct models of large networks of such neurons with a variety of topologies. In Part Two, the student will learn how to characterise the resulting behaviour using various measures, and will acquire an understanding of the phenomena that are revealed as a result.

Syllabus

Part One: Modelling Methods

- Neuron models
    - Integrate-and-fire
    - Leaky integrate-and-fire
    - Izhikevich
    - FitzHugh-Nagumo
    - Hodgkin-Huxley

- Numerical simulation
    - Euler method
    - Runge-Kutta method

- Complex networks
    - Small-world networks
    - Efficiency
    - Modular networks
    - Hierarchical small-world networks
    - Hub nodes
    - Connective cores

Part Two: Measures and Properties

- Dynamical complexity
    - Neural complexity

        - Causal density
        - Transfer entropy
        - Information integration (phi)

- Synchronisation
    - Oscillator networks (Kuramoto model)
    - Synchronisation measures
    - Entrainment
    - Quenching

- Reverberation

- Attractor networks

- Metastability and chaos
    - Metastability in oscillator networks
        - Noise
        - Variable natural frequency
        - Phase lag (chimera states)
    - Metastability in neural networks
    - Chaotic itinerancy

- Criticality

- Residual Topics
    - Spike Timing-Dependent Plasticity (STDP)
    - Embodiment and cognition

C422 COMPUTATIONAL FINANCE

Dr. D. Kuhn ***Term 2***

[Prerequisite: All students are assumed to be familiar with basic analysis and linear algebra. The lecture notes for C233 are available from: http://tinyurl.com/c223-notes]

Recommended: Operations Research (C343). Students who have not taken this course are assumed to be familiar with linear programming duality. The lecture notes for C343 are available from http://tinyurl.com/kuhn-taught-courses

This course introduces the theory and application of modern quantitative finance from an engineering perspective.

Aims
After this course, the students will:
- understand the basic concepts of quantitative finance and financial engineering;
- be aware of the major decision, hedging, and pricing problems in finance, know how to formulate these problems as mathematical models, and understand the computational techniques to solve the arising models.

Learning Outcomes
This course will enable the students to:

- compare and appraise the basic theories that underlie current thinking in finance and investment;
- describe how these theories are applied in practical situations;
- describe the properties of the principle asset classes and securities;
- apply the basic analytical methods and computational tools used in finance;

- solve portfolio selection problems with off-the-shelf optimization software;
- solve option pricing problems based on binomial lattices;
- be able to read the technical literature in computational finance and undertake independent self-study (or research) in the future.

Course Outline
1. Introduction to Computational Finance
1.1 Course Organization
1.2 Cash Flow Streams
1.3 Investments and the Market
1.4 Toy Example for Financial Option Pricing
2. Mathematical Preliminaries
2.1 Functions
2.2 Differential Calculus
2.3 Optimization
2.4 Probability Theory
3. The Basic Theory of Interest
3.1 The Time Value of Money
3.2 Net Present Value
3.3 The Term Structure of Interest Rates
4. Fixed-Income Securities
4.1 Terminology and Examples
4.2 Pricing of Fixed-Income Securities
4.3 Risk-Management of Fixed-Income Securities
5. Mean-Variance Portfolio Theory
5.1 Asset Returns
5.2 Variance as a Risk Measure
5.3 Markowitz Problem
5.4 Parameter Estimation
6. The Capital Asset Pricing Model (CAPM)
6.1 The One- and Two-Fund Theorems
6.2 The Capital Market Line
6.3 Systematic and Unsystematic Risk
6.4 CAPM as a Pricing Formula
7. General Principles
7.1 Utility Theory and Risk Aversion
7.2 Portfolio Choice and Linear Pricing
7.3 Risk-Neutral Pricing
8. Asset Price Dynamics
8.1 Binomial Lattices
8.2 Ito Processes
9. Basic Options Theory
9.1 Definitions, Terminology, and Payoff Diagrams
9.2 Single-Period Binomial Options Theory
9.3 Multi-Period Binomial Options Theory
9.4 Real Options

Lecture Notes: *http://tinyurl.com/kuhn-taught-courses*

C424 MACHINE LEARNING AND NEURAL COMPUTATION

Dr. A. Faisal ***Term 1***

The course covers the foundations and methods of Machine Learning & Neural Computation and relates it to current thinking in neuroscience. We take a principled approach to the probabilistic foundations of the subject, following a pragmatic, Bayesian approach (you will learn what this means). Aim of the course is to look at those state-of-the-art methods that can be derived from first principles, allowing us an intuition as to how, why, and when these methods are best applied to real world problems - and in conclusion allow you to think about developing them further.

We will start with a single neuron as a model and see how we can think about learning, decision making and prediction. This forms a prelude to the three core topics covered: supervised learning - unsupervised learning - reinforcement learning. Gradually we will expand this view to networks, introducing any required mathematics along the way.

Supervised learning is the problem of finding an input-output mapping from empirical data - this problem is known as either regression for continuous outputs (e.g. learning the kinematics of a robot arm), or classification when outputs are discrete or categorical (e.g. recognising faces in photos). Here we start with the classic neural network approach, move on to the industry standard "Support Vector Machines" (and the underlying ideas of Kernel Methods) and look into the state-of-the-art Gaussian processes framework.

Unsupervised learning is the problem of finding structure in data. One can think of this as the process of self-organization of information by the brain. We will concentrate on 1. dimensionality reduction techniques such as probabilistic Principal Component Analysis (PCA) and learn about the more powerful Latent Variable Models and 2. clustering methods, both the classic k-Means Algorithm and Mixture Models. In this context we will learn about information theory and why learning is equivalent to data compression, and how neurones can do it.

Reinforcement learning is a growing sub-area of machine learning concerned with how an agent (computer or robot) should choose it's actions in an environment so as to maximise some notion of long-term reward. This is a crucial problem when we don't know what the correct output at any given moment should look like (that would be supervised learning), but have some goal we want to see achieved (which comes with a reward). Think of it as the method by which you learn to cycle: your instructor cannot tell you how to switch on and off your 600 muscles to ride the bike, only if you are doing "it" well or not. Your brain solves the reinforcement problem all by itself. We will cover Temporal Difference Learning and Q-Learning as two approaches that solve the task algorithmically and see the former is implemented in the brain.

The course thus shows how Machine Learning has both been inspired by and is inspiring neuroscience research over the past years - a prime example of Neurotechnology at work. There are no prerequisites.

C429 PARALLEL ALGORITHMS

Dr. W. Knottenbelt/Professor P. Harrison ***Term 2***

Syllabus
Introduction and motivation: key concepts, performance metrics, scalability and overheads. Classification of algorithms, architectures and applications: searching, divide and conquer, data parallel. Static and dynamic, message passing and shared memory, systolic.
Sorting and searching algorithms: mergesort, quicksort and bitonic sort, implementation on different architectures. Parallel depth-first and breadth-first search techniques.
Matrix algorithms: striping and partitioning, matrix multiplication, linear equations, eigenvalues, dense and sparse techniques, finite element and conjugate gradient methods. Optimisation: graph problems, shortest path and spanning trees. Dynamic programming, knapsack problems, scheduling, element methods.
Synthesis of parallel algorithms: algebraic methods, pipelines, homomorphisms.

C436 PERFORMANCE ANALYSIS
[Exclusive with M3S4 Applied Probability I]

Dr. G. Casale/Professor P. Harrison ***Term 2***

Objectives

To introduce analytical modelling techniques for predicting computer system performance.

Contents
Motivation and survey: the need for performance prediction in optimisation and system design.
Basic probability theory: renewal processes; Markov processes; birth and death processes; the single server queue; Little's law; embedded Markov chain; M/G/1 queue; queues with priorities; queueing networks - open, closed, multi-class; equilibrium state space probabilities, proof for single class; normalising constants; computation of performance measures; convolution algorithm; mean value analysis; application to multi-access systems with thrashing.
Decomposition and aggregation: Norton's theorem; M/M/n queue; multiple independent parallel servers.
The course also offers an introduction to performance modelling using a stochastic process algebra, e.g. PEPA. To include: expansion law, apparent rate, steady-state analysis, transient state analysis through uniformisation and reward vectors.

C437 DISTRIBUTED ALGORITHMS
[Recommended: Concurrency (C223), Distributed Systems (C335)]

Professor A. Wolf ***Term 2***

Learning Outcomes
By the end of this course students will be able to understand and explain the concepts behind distributed algorithms, including the assumptions made and the potential benefits and shortcomings. Students will also be able to assess the applicability of distributed algorithms to a particular circumstance.

Syllabus
Models of distributed computing
Synchrony, communication and failure concerns
Synchronous message-passing distributed systems
Algorithms in systems with no failures - Leader Election and Breadth-First Search algorithms
The atomic commit problem
Consensus problems - the Byzantine Generals Problem
Asynchronous message-passing distributed systems
Logical time and global system snapshots Impossibility of consensus
Fault-tolerant broadcasts
Partially synchronous message-passing distributed systems
Failure detectors
The Labelled Transition System Analyser (LTSA) tool http://tinyurl.com/magee-ltsa-book is used throughout the course for modelling and demonstrating the execution of various algorithms.

C438 COMPLEXITY
[Recommended: Models of Computation (C240) or (M20D)]

Dr. I. Phillips ***Term 2***

Aims
To describe the complexity classes associated with computational problems. To develop the ability to fit a particular problem into a class of related problems, and so to appreciate the efficiency attainable by algorithms to solve the particular problem. Turing machines, decidability, machine independence.

Syllabus
Time complexity: the classes P and NP, NP-completeness, example problems from logic and graphs

Space complexity classes
The parallel computation thesis, PRAMs, the class NC
Probabilistic algorithms
Connections with logic and/or process theory
Cryptography
Protocols, zero-knowledge proofs

C470 PROGRAM ANALYSIS

Dr. H. Wiklicky/Professor C.L. Hankin ***Term 2***

Aims
To gain an understanding of program analysis techniques that are used in optimising compilers for imperative and functional programming languages. To develop understanding of data flow analysis techniques, control flow analysis techniques and constraint solving.

Learning Outcomes
The student who completes this module will be able to apply classical analyses to programs; solve the resulting constraints; develop analyses from scratch; and informally reason about the complexity of constraint solving algorithms.

Outline Syllabus
Introduction to four main approaches: data flow analysis, control flow analysis, abstract interpretation, types and effect systems. Program transformations. Data Flow Analysis: basic blocks and control flow, classical analyses, monotone frameworks, worklist algorithms, interprocedural analysis and shape analysis. Flow logic: Control flow analysis, language based security and information flow. Algorithms: A generic worklist algorithm, the round robin algorithm and strong components.

C471 ADVANCED ISSUES IN OBJECT ORIENTED PROGRAMMING
[Recommended: Type Systems for Programming Languages (C382)]

Professor S. Drossopoulou ***Term 1***

Objectives
To discuss issues around the design and implementation of object oriented languages, the rationale and explore alternatives. To use formal calculi as an unambiguous notation, and as a way to establish soundness.

After the Course, students should

- be able to develop a formal description of a small extension of the languages described in the course
- understand the interplay between static and dynamic checks, and the various checks applied at the different phases of execution
- understand how efficiency of implementation issues, and understand how some features can be implemented efficiently e.g. object layout, virtual tables, (multiple/virtual) inheritance - understand the difference-similarities of the object based and class based paradigm

Contents
Motivation, type system of some language(s) unsafe. Static vs dynamic types. Sound type systems.
L1: a minimal, class based, imperative, object oriented language with methods and fields. Operational Semantics, Type system, Agreement, Soundness.
L2: L1+inheritance. Operational Semantics, Type system, Agreement, Soundness.
The expressive power of L2: Numbers and Boolean.
C++ features and implementation. Pointer vs Value Types. Single Inheritance, Multiple Inheritance. Virtual Inheritance. Object layout, virtual tables, implementation of assignment and of method call.

The Java virtual machine. The bytecode verifier. Formalization
Java dynamic linking.
Ownership types.
The Abadi & Cardelli Object Calculus.

C474 MULTI-AGENT SYSTEMS
[Recommended: AI (Year 2 or 3), Prolog and Java)]

Dr. F. Toni ***Term 2***

Learning Outcomes
At the end of the course students will understand the concept of a software agent as a process embedded in, sensing (reacting to changes in) an environment whilst pursuing implicit or explicit goals and holding beliefs.
They will know how to program a multi-agent application.
They will understand issues to do with co-operation between multiple agents and the need for standardised inter-agent communication languages. They will have some knowledge of various kinds of strategic behaviour and mechanisms for co-operation.

Content
General introduction to the concept of agent and multi-agent system.
Abstract agent architecture.
Several types of agents: BDI, AgentSpeak, KGP agents
Distributed problem solving and the contract net protocol.
Agent communication and agent communication languages - including KQML and FIPA ACL.
Communication policies and protocols, dialogues, various notions of conformance (of policies to protocols, of dialogues to policies and protocols).
Resource allocation, negotiation, various notions of social welfare (utilitiarian, egalitarian), various notions of negotiation deals.
Agents and game-theory.
Agents and argumentation.

Assessment
Assessed coursework will comprise of a programming assignment.

C475 SOFTWARE ENGINEERING FOR INDUSTRY

Dr. R. Chatley ***Term 1***

Aims
To introduce state-of-the-art advanced tools and techniques for large-scale software systems. To develop the critical skills to judge which technique would be most appropriate for solving large-scale software problems.

Learning Outcomes
1) Students will have an understanding of some current software process methodologies and be able to apply critical facilities to valuing methodologies as they become fashionable.
2) Students will understand the scope of the software maintenance problem and will be familiar with several techniques and tools for reverse engineering software.
3) Students will be better prepared to make the right technical decisions when working on changing, large codebases.

Syllabus
Maintenance and Evolution: Software maintenance/evolution is the modification of a software product after delivery to correct faults, to improve performance or other attributes, or to adapt the product to a changed environment.

We look at a collection of software evolution problems associated with modern code and at reverse engineering techniques for code improvement.
Test driven development (which is more than test first), mock objects, acceptance testing, continuous integration, coverage etc, acceptance testing and non-technical stakeholders.
Intentional programming: programs that are designed to be read, expressing why as well as what, embedded DSLs, domain driven design, building a bridge to the customer/tester/analyst
Software Components: Composition and Extension
'How to build a system from reusable components, and how to extend it'
Building software out of components, allowing us to take more of a construction-like approach. A hierarchical component model that allows software to be constructed from pieces. Examine component systems used commercially and show how they compare to the real thing. i.e. how do systems like Spring, EJB, COM match up? How do component systems effect day-to-day development in the real world.
Structuring (Large) OO Systems:
Architecting an OO system for size, clarity, reuse, extensibility and collaboration
What are the principles that prevent systems from descending into a tangled mess? How do some systems manage to stay well organised even with lots of people working on them?
Designing a System under Pressure:
How to design software well and apply the correct principles when faced with commercial pressures. In the commercial world, the pressure to deliver software quickly forces all sorts of shortcuts. How do you design and implement a system well and retain good development principles even when faced with commercial pressures and the reality of the workplace today. i.e. changing requirements (i.e. 'I've changed my mind. Here are my new requirements'), time pressures ('We want this yesterday, just do it!'), dangerous colleagues ('I didn't really understand the system, but I changed the code all over the place') and lack of understanding from management ('Just release it, we'll test it later').

C477 COMPUTING FOR OPTIMAL DECISIONS

[Recommended: Operations Research (C343), or (M3N3)]

Professor B. Rustem ***Term 1***

Aims
To develop a deeper understanding of optimal decision making models, algorithms and applications to finance. To provide an insight for algorithm design and formulation of decision models.

Learning Outcomes
The student will be able to analyse algorithms and judge the performance of algorithms and interpret their results. The course opens into issues related to current research.

Syllabus
To provide mathematical concepts and advanced computational methods for quantitative problems in management decision making. To introduce unconstrained and constrained optimal decision formulations and associated optimality conditions. To discuss quadratic and general nonlinear programming formulations and algorithms.
Introduction to optimisation and optimal decisions. Unconstrained optimisation. Constrained optimisation.
Management decision formulations. Optimality conditions for constrained problems. Necessary conditions, sufficient conditions. Quadratic programming: problem formalisation; portfolio selection. Optimality conditions.
Algorithms for quadratic programming. Nonlinear programming: example formulations; capacity expansion, inventory control. Problem formulation. Algorithms for nonlinear programming.

C480 AUTOMATED REASONING

Dr. K. Broda ***Term 1***

Aims
To develop understanding of the basic techniques used in automated reasoning.

Learning Outcomes
Students will gain knowledge of propositional and first order automated reasoning methods (Davis Putnam, Model Generation, resolution and refinements, connection graphs, Tableau methods, paramodulation and equational reasoning) and be able to carry out small examples by hand. They will know how to prove soundness and completeness of the techniques. Students will gain experience of using the Otter theorem prover to solve problems and to investigate search spaces. They will understand the Knuth Bendix completion procedure and its applications to equational reasoning and be able to apply it to small examples by hand.

Syllabus
To describe the main techniques for automated reasoning in classical logic.
Automated deduction: a brief history from 1960 to the present time.
Propositional theorem proving; model generation and Davis Putnam Procedure.
Resolution for the first order predicate calculus, completeness and soundness.
Refinements of resolution, hyper-resolution, locking, connection graphs, advantages, disadvantages, use of the Otter theorem prover
Tableau methods with unification; linear refinements, extensions.
Paramodulation, theory resolution.
Equational methods for equational theories.
Confluent rewriting systems and the Knuth-Bendix algorithm, unfailing Knuth-Bendix.

C481 MODELS OF CONCURRENT COMPUTATION
[Recommended: Concurrency (C223)]

Professor P. Gardner/Dr. N. Yoshida ***Term 1***

Aims
To describe process models for specifying and verifying concurrent systems.
In particular, to focus on CCS (the calculus for concurrent systems) and the pi-calculus, which provide communication via handshake and message passing.
CCS: handshake communication, operators and reduction congruence, case studies, Hennessey-Milner logic, Concurrency Workbench (software tool) and process equivalence.
Calculi for mobility and its applications: handshake communication and mobility via message passing, the Pi-Calculus, asynchrony, reduction congruence, distributed mobility and applications to Web protocol verifications, security and biological systems.

C484 QUANTUM COMPUTING

Dr. H. Wiklicky ***Term 2***

Objectives
To introduce the basic notions of quantum computing with particular emphasis on quantum algorithms.

Learning Outcomes
Students will, by the end of the course, be able to understand and explain the basic notions of Quantum Computing-including Quantum Bits and registers, Quantum Evolution, Quantum Circuits, Quantum Teleportation and the basic Quantum Algorithms known at the present time. They will also be able to identify the essential difference between the classical paradigm and the quantum paradigm of computation and

appreciate why quantum computers can solve currently intractable problems such as the prime factorisation of large numbers in polynominal time.

Contents
Introduction to Quantum Mechanics, Quantum Bits and Complex Vector Spaces, Quantum Evolution and Quantum Gates, Quantum Registers, Universal Gates, No-Cloning Theorem, Quantum Entanglement and Teleportation, Quantum Algorithms, Quantum Search, Quantum Fourier Transform, Phase Estimation, Quantum Counting, Order Finding for Periodic Functions, Quantum Factoring of Integers, Physical Realization of Quantum Gates and Quantum Error Correction.

C491 KNOWLEDGE REPRESENTATION

Professor M. Sergot ***Term 2***

Aims
The aim of the course is to present the theoretical foundations of the main formalisms for knowledge representation and reasoning, to show how they can be used in practice, and to provide an overview of current research trends. Particular attention will be given to logic-based formalisms, and to comparing and translating between different approaches.

Syllabus
Current general issues in knowledge representation.
Default and non-monotonic reasoning: default logic, autoepistemic logic, circumscription, negation as failure, abduction as a form of hypothetical and default reasoning, non-monotonic consequence relations and defeasibility.
Theories of argumentation.
Temporal reasoning: representation of action, non-monotonic features of persistence, the frame problem.
Dynamics of belief systems and databases: consistency and integrity, knowledge assimilation, theories of belief revision and update, counterfactuals.

C493 INTELLIGENT DATA AND PROBABILISTIC INFERENCE

Professor D. Gillies ***Term 2***

Syllabus
The course is concerned with probabilistic methods for modelling data and making inferences from it. The first part introduces Bayesian inference and networks and includes probability propagation and inference in singly connected networks, generating networks from data, and calculating the network accuracy. The course then goes on to consider highly dependent data and special techniques for exact and approximate inference in these cases. The next topics to be covered include data modelling using distributions and mixture models, sampling and re-sampling, data reduction by principal component analysis and special problems that occur with small sample sizes. The last part of the course is concerned with classification using linear discriminant analysis and support vectors.
The emphasis of the course is algorithmic rather than mathematical, and the coursework is a practical programming exercise in analysing data from a study into the prognosis of Hepatitis C.
Note that the course does not include non-probabilistic methods of data analysis such as neural networks, fuzzy logic or expert systems.

Course URL: http://tinyurl.com/gillies-c493

C499 MODAL AND TEMPORAL LOGIC
[Prerequisite: Familiarity with propositional logic and acquaintance with classical first-order predicate logic – e.g. the contents of course 140]

Professor I. Hodkinson and others ***Term 1***

Aims
To develop skills in the use of modal and temporal logics for specification, knowledge representation and practical reasoning in artificial intelligence and software engineering.

Learning Outcomes
Knowledge and understanding
Students should have assimilated the distinctive characteristics of modalities.
They should understand and recall the definitions of the logics and logical systems presented in the course, and the ideas of the proofs.
They should appreciate the differences in the logics, and their capabilities and limitations.

Skills and Other Attributes
a) intellectual skills - Students should be able to undertake simple proofs and examples, and modify coursework proofs appropriately to new situations. They should be able to follow the applications of modal logic in the computer science and AI literature.
b) practical skills - Students should gain familiarity with the issues involved in the use of modal and temporal logics in practical situations.
They should be able to recall relevant logics and reasoning techniques and modify and apply them accurately in appropriate situations.
c) transferable/key skills as above.

Syllabus
Elements of propositional modal logics: syntax, Kripke frames and models, validity, frame correspondents of modal formulas, Sahlqvist's correspondence theorem. p-morphisms and bisimulations.
Temporal logic: various connectives. Advanced topics such as separation, interval temporal logic, hybrid logic, as time permits.
Systems of modal logic: deducibility, soundness, completeness, consistency. Completeness via canonical models, for K and other logics.
Filtration, finite model property, decidability.
Advanced topics such as modal mu-calculus, as time permits.