Cambridge 2022–2023 [1st Class Honours]

- **Bioinformatics** Taught by Pietro Lio'
  A course studying the algorithms used in bioinformatics: Nussinov, UPGMA, K-means, Markov clustering, Burrows-Wheeler transform, Baum-Welch, Louvain method etc. Some surronding content on biology too, although this wasn't the focus.

- **Denotational Semantics** Taught by Marcelo Fiore
  A course that defined a denotational semantics for PCF, with Scott Induction, the least fixed point operator acting as a typed Y combinator etc. This course inspired my dissertation.

- **Information Theory** Taught by Robert Harle
  A course on Shannon's information theory: entropy, noisy coding, compression, huffman codes, continuous entropy, the theory's applications to machine learning and its applications in physics.

- **Types** Taught by Neel Krishnaswami
  A followup to the course Semantics of Programming Languages, studying more exotic semantics and types: Curry-Howard, polymorphic lambda calculus, monads, continuations and their correspondence to classical logic, dependent types.

- **Category Theory** Taught by Andrew Pitts
  A classic introduction to category theory, with a bend towards theoretical computer science : commutative diagrams, monoids as categories, modelling the semantics of the simply-typed lambda calculus in any cartesian-closed category. Got up to the the Yoneda lemma and monads. N.B. This course was quite difficult, and very rewarding.

- **Cryptography** Taught by Markus Kuhn
  A course in attacks and defence: ciphers and the methods to crack them, from a very mathematical perspective. Some time on elliptic curves and quadratic residues.

- **Machine Learning and Bayesian Inference** Taught by Sean Holden
  A follow-up course to the AI course before. Covered modern methods in AI (neural nets, bayesian networks, EM algorithm, SVMs, Gaussian processes) and more general reasoning probabilistically.

- **Quantum Computing** Taught by Steven Herbert
  An introduction to quantum computing, framed mostly as an linear algebra course. Covered Deuthsc-Jozsa, QFT, Grover search, quantum error correction and general trends in modern research and industry-work in the field. N.B. The professor worked as a researcher in industry, developing quantum processors.

- **Randomised Algorithms** Taught by Thomas Sauerwald
  A course on randomised algorithms and the mathematics required to analyse them. Covered tools from probability (inequalities: Cheeger, Chebyshev, Chernoff) and a whole host of beautiful algorithms. N.B. This course was very difficult, comfortably the most challenging one I took.

- **Deep Neural Networks** Taught by Ference Huszar and Nic Lane
  A course to get up to date with modern neural net research. Mostly practical in nature, with big projects (train a transformer, fine-tune a resnet) and lectures from guest researchers (e.g. Sara Hooker of "The Hardware Lottery" fame).

- **Hoare Logic and Model Checking** Taught by Christopher Pulte
  Two separate topics. Firstly, Hoare logic as a method for formal specification of programs, with proofs of soundness and completetness. Secondly, model checking as a field, and temporal logic for modelling system evolution.

Cambridge 2021–2022 [1st Class Honours]

- **Artificial Intelligence** Taught by Sean Holden
  A course mostly covering GOFAI, particularly search methods, CSPs, solving Planning problems, and game-playing (alpha-beta pruning). The last 3 lectures were devoted to neural networks, gradient descent, and backpropogation.

- **Compiler Construction** Taught by Timothy Griffin
  A first course on building a compiler, covering LL parsing, LR parsing, context free grammars etc. Also spent several lectures on more advanced topics, such as register allocation and spooling, bootstrapping, calling conventions, peephole optimisation, implementing inheritance.

- **Complexity Theory** Taught by Anuj Dawar
  A fairly standard course in Complexity Theory, characterising NP-completeness, the Cook-Levin Theorem, the Savitch Theorem, and Fagin's theorem.

- **Computation Theory** Taught by Andrew Pitts
  A course on the halting problems and several ways of defining computability. Covered register machines, Turing machines, partial recursive functions and the λ-calculus.

- **Computer Networking** Taught by Andrew Moore
  A comprehensive course that covered every layer of the internet stack in detail.

- **Concepts in Programming Languages** Taught by Alan Mycroft
  A historical and contemplative course about different programming languages, their evolutions and design decisions.

- **Concurrent and Distributed Systems** Taught by David Greaves and Martin Kleppmann
  A course split into two parts. The first was on basic concurrency primitives: locks, semaphores, monitors, condition variables, their implmentation and uses. The second was on distributed systems, covering Raft/Paxos and common problems and solutions in distributed systems. Included a case study of Google's Spanner database.

- **Data Science** Taught by Damon Wischik
  A course mostly centred on maximal likelihood estimation and finding estimators thus. Also included some basic theory of Markov chains.

- **Economics, Law and Ethics** Taught by Alice Hutchings
  A course covering basic theories of microeconomics, supply, demand, consumer surplus, monopoly rents etc. as well as common ethical paradigms and laws around software production and computer (mis)use.

- **Formal Models of Language** Taught by Paula Buttery
  An introduction to the intersection of linguistics and computer science, including the Chomsky hierarchy, Earley parsers, shift-reduce parsers, grammar induction, Gold's theorem, information theory, and word vectors.

- **Further Graphics** Taught by Cengiz Oztireli
  A second course in graphics, covering lighting equations, discrete differential geometry, dual quaternions and their use for fast approximations of skeletal transformations and inverse rendering.

- **Further Human-Computer Interaction** Taught by Alan Blackwell and Luke Church
  A second course in HCI, covering theory-driven approaches, including Bayesian methods for predictive interfaces, modern research directions, cognitive theories of planning, and optimising human performance via statistical experimental methods.

- **Further Java** Taught by Alastair Beresford and Andrew Rice
  A second course in Java, covering the use of concurrency, networking and reflection to culminate in building a distributed messaging service.

- **Introduction to Computer Architecture** Taught by Simon Moore
  A first course in Architecture, covering RISC-V, ISA design, ISA implementation, some assembly coding, processor pipelining, use of GPUs for computation and CUDA as well as memory models and the MSI protocol for cache coherency.

- **Logic and Proof** Taught by Larry Paulson
  A course on predicate calculus and the history of automated reasoning, including BDDs, SAT solvers, modal logic, Horn clauses (and Prolog), resolution and unification, Gentzen's sequent calculus, Skolemisation, Herbrand's theorem, SMT solvers and tableaux methods.

- **Programming in C and C++** Taught by David Greaves and Alan Mycroft
  The standard introductory course in C with a few lectures devoted to differences with C++ and new paradigms in C++ such as RAII and operator overloading.

- **Prolog** Taught by Andrew Rice
  A course on declarative programming in Prolog and what problems the language is suited for solving.

- **Security** Taught by Frank Stajano
  A course on security, taught from an adversarial perspective. Classic hacking techniques were covered, such as setuid attacks, buffer overflows, SQL injection, password cracking, cross-site request forgery, TCP attacks, virus designing.

- **Semantics of Programming Languages** Taught by Neel Krishnaswami
  A course covering the definition of a semantics for a simple programming language, including types, functions, product and sum-types, subtyping and concurrency.

- **Unix Tools** Taught by Marcus Kuhn
  A course covering a working programmer's everyday tools.

Cambridge 2020–2021 [Upper Second Class Honours]

- **Algorithms** Taught by Frank Stajano and Damon Wischik
  The standard first course on data structures and algorithms. Topics included sorting, dynamic programming, greedy algorithms, hash tables, B-trees (and red-black trees), graph algorithms (Ford-Fulkerson, Dijkstra, etc.) and amortized analysis.
  I was top of the year on this topic's exam question.

- **Databases** Taught by Timothy Griffin
  A course introducing the purpose and usage of databases, including the object-relationship model and the graph-oriented model and the relational calculus.

- **Digital Electronics** Taught by Ian Wassell
  A course on circuitry, combinational logic and sequential logic. Topics included Boolean algebra, adders, latches, flip flops and registers.

- **Discrete Mathematics** Taught by Marcelo Fiore and Frank Stajano
  A course in three parts: the first emphasises common proof techniques via basic number theory: solving congruence relations etc; the second is on set theory - Calculus of bijections, axiom of choice, Cantor-Schoeder-Bernstein Theorem etc; the third is on formal languages and automata, through to Kleene's theorem and the Pumping Lemma.

- **Foundations of Computer Science** Taught by Robert Harle and Jeremy Yallop
  An introductory computer science course taught via OCaml. Topics include Big O notation, basic algorithmic complexity assessment, abstract data structures, recursive functions, tail-recursion optimisation.

- **Introduction to Graphics** Taught by Rafal Mantiuk
  A course on basic graphics. Topics included implementing the standard ray tracing model; projection; rasterisation; homogeneous coordinates and colour theory.

- **Machine Learning and Real World Data** Taught by Simone Teufel
  A course in implementing basic machine learning methods. Taught in three parts: the first on sentiment classification via Naive Bayes, the second on identifying where protein sequences enter membranes via the Hidden Markov Models and the Viterbi algorithm, and the third on social networks and Brandes's algorithm for efficiently computing betweeness centrality of edges and nodes.

- **Mathematics for the Natural Sciences** Taught by Stuart Dalziel
  A course on mathematics required for university-level science problems. Topics included Cartesian geometry with vectors, complex numbers, probability, vector calculus and differential equations.

- **Object-Oriented Programming** Taught by Andrew Rice
  A course on OOP fundamentals, taught via Java. Topics include inheritance, classes and objects, polymorphism, garbage collection, error handling, design patterns and language evolution.

- **Operating Systems** Taught by Evangelia Kalyvianaki
  A course on function and design principles of operating systems. Topics included processes, scheduling algorithms, memory management (segmentation and paging), I/O management, file management and a case study of Unix.

- **Interaction Design** Taught by Hatice Gunes
  A course on designing things with human users. Topics included Gestalt psychology, usability, heuristics for design, data grathering, concept development and iterative design processes.

- **Introduction to Probability** Taught by Mateja Jamnik
  A standard first course in probability. Topics included Markov and Chebyshef inequalities, join and marginal distributions, covariance, the Weak Law of Large Numbers, the Central Limit Theorem and Maximum-Likelihood-Estimation.

- **Software and Security Engineering** Taught by Ross Anderson
  A course on building large systems; how to do it well, and how it can go wrong. Topics included security policies, security protocols (e.g. Needham-Schroeder), famous security attacks (Heartbleed), managing safety-critical systems and the modern software-development paradigms like Agile development.

Pomona College

Mathematics

- [A] **Math 60: Linear Algebra** Taught by Stefan Garcia
  Emphasised linear independence and bases, null spaces and ranks of linear transformations, representation of transformations

by matrices. Included diagonalisation, eigenvalues/eigenvectors and applications of linear algebra such as least squares problem, singular value decomposition.
No textbook used.

- [N/A]**Math 101: Introduction to Real Analysis** Taught by Edray Goins
  A course mostly aimed on structure and presentation of proofs. Construction of the real numbers using Dedekind cuts was also discussed, some elementary epsilon-delta analysis was performed.
  No textbook used.
  N.B. Due to course registration limits, I did not formally register for this class, though averaged over 95% across homeworks and tests for it.

- [P]**Math 103: Combinatorial Mathematics** Taught by Shahriar Shahriari
  An introduction to combinatorics and its techniques, including basic counting methods, Ramsey theory, generating functions, elementary graph theory and networking.
  Textbook used: *An Invitation to Combinatorics* (then in pre-print) by Shahriar Shahriari.
  N.B. Due to the coronavirus epidemic, I was not assigned a grade for this course, though averaged over 95% across homeworks and tests for it.

- [N/A]**Math 135: Functions of a complex variable** Taught by Stefan Garcia
  A course on introducing holomorphic functions and their basic properties, including the Cauchy Riemann equations, Cauchy's Integral formula, calculus of residues, winding numbers, conformal mappings and a proof of the Prime Number Theorem.
  No textbook used.
  N.B. Due to course registration limits I did not formally register for this class, though submitted the relevant work for it. I was probably on track for a B.

- [P]**Math 171: Abstract Algebra: Groups and rings** Taught by Ghassan Sarkis
  A course on basic group theory: from axioms to isomorphism theorems through Lagrange, Cauchy and Sylow theorems with emphasis on cosets and group actions. A few weeks devoted to basic ring theory: distinction between Euclidean domains, Principal ideal domains and unique factorisation domains.
  Textbook used: *Algebra in Action: A course in Groups, Rings, and Fields* by Shahriar Shahriari (Chapters 1-12 and 15-18)
  N.B. Due to the coronavirus epidemic, I was not assigned a grade for this course, though averaged over 90% across homeworks and tests for it.

- [N/A]**Math 173: Advanced Linear Algebra** Taught by Stefan Garcia
  A course designed to redo linear algebra from a more generalised infinite-dimensional perspective. Topics included general inner product spaces, similarity, the spectral theorem, Jordan canonical form, the Cayley Hamilton theorem, single value decomposition.
  Textbook used: *A second course in Linear Algebra* by Stefan Garcia and Roger Horn
  N.B. Due to course registration limits I did not formally register for this class, though averaged 90% across homeworks and tests for it.

Other courses

- [P]**Greek 104: Readings in Koine Greek** Taught by Ben Keim
  A half-credit course during which we translated the gospel according to John together.

- [P]**Greek 104: Readings in Koine Greek** Taught by Ben Keim
  A half-credit course during which we translated Revelation together.

- [A]**Greek 33: Intermediate Greek** Taught by Richard McKirahan
  A course in Ancient Greek, aimed at honing grammar and translation skills. Mostly focused on Xenophon.

- [P]**Greek 44: Advanced Greek** Taught by David Roselli
  A course in Ancient Greek, largely aimed at focusing on details of texts, rather than basic translation skills. Mostly focused on Homer, with some lyric poetry.
  N.B. Due to the coronavirus pandemic, I was not assigned a grade for this course, though believe I was on track for an A.

- [P]**History 101K: Politics of Honor in Ancient Greece** Taught by Ben Keim
  A course on the meaning of honor in ancient mediterranean society and how its importance was reflected in the materials we have from that time.
  N.B. Due to the coronavirus pandemic, I was not assigned a grade for this course, though believe I was on track for an A-/B+.

- [A]**Anthropology 145: Mesoamerican Archaeology** Taught by Arlen Chase
  An introductory course with a world-expert with his own dig site in Belize. Predominantly focused on the Maya, though the Aztec were covered too.

- [A]**Chemistry 51: Accelerated General Chemistry** Taught by Zhao Li
  A fast-paced course designed to cover first-year chemistry in one semester. Topics included chemical equilibria, atomic structure, themodynamics, basic quantum mechanics and experimental technique, including computer modeling.