This is a translation of my "Manifesto degli Studi".

Course name	Syllabus/Content
Calculus 1	• Definition of derivative as a limit and the rules of differentiation to differentiate functions. • The graph of a function using asymptotes, critical points, and the derivative test for increasing/decreasing and concavity properties. • Max/min problems and the use of differentiation to solve them • Integrals by using the Fundamental Theorem of Calculus. • Techniques of integration, such as substitution, inverse substitution, partial fractions and integration by parts.
Architettura dei Calcolatori (Computer Architecture and Organization)	• Main elements of a computer, performances and computer languages, high level language, assembler and assembly language (MIPS architecture) • A simple ALU model, boolean logic, CPU and control unit • Memory hierarchies, mass storage memories and circuits • Interrupts, traps and System Calls
Algoritmi e Strutture Dati (Algorithms and Data Structures)	• Methods for algorithm analysis: cost criteria, asymptotic notation, complexity analysis of recursive algorithms. Examples of development and analysis of algorithms. • Sorting algorithms: insertion sort, selection sort, bubble sort, mergesort, quicksort • Basic data structures: arrays and lists; stacks and queues; dictionaries implemented with lists. • Dictionaries: implementation with binary search trees and hash tables. • Trees: indexed and linked representations for binary trees and general trees; depth-first search and breadth-first search of trees. • Search Trees: Binary search trees, search trees as a data structure for implementing dictionaries, balanced trees. • Hash tables: collision lists and open addressing. • Priority queues: implementation with lists and heaps. • Graphs: definitions, data structures, primitives for querying and updating graphs; graph visits in depth and in width; examples of application of a graph visit algorithms. • Laboratory: C++ laboratories related to course topics
Introduzione alla Programmazione (Introduction to Computer Programming)	• Introduction to the course, computational thinking, and programming. • Programming environments and formal languages. • Imperative programming: variables and instructions; assignment; input / output; if-then-else; loops; types; arrays; strings; functions; struct; dynamically allocating memory (heap); dynamic vectors; linked lists ; recursion and iteration. • Simple algorithms on sequential data: ordering and search • Introduction to problem analysis and solution design, with examples.

Algebra e Logica per Informatica (Algebra and Logics for Computer Science)	● Sets, functions, complex numbers, cardinality, combinatorics ● Algebraic structures, monoids, groups, rings, vectorial spaces, k-algebras, strings ● Heterogeneous algebras ● Introduction to logic ● Syntax of propositional logic ● Semantics of propositional logic: truth tables ● Syntax of first order logic ● Semantics of first order logic: models theory ● Formalization of problems
Linguaggi e Programmazione orientata agli Oggetti (Programming Languages and Object-Oriented Programming)	● Principles of programming languages: overview, programming paradigms; syntax, static and dynamic semantics, development of a semantics-driven interpreter. ● Object-oriented paradigm (the Java programming language): objects, classes, inheritance and dynamic binding, casting, overloading, exceptions. Advanced features: generic classes and methods, outline on nested classes and lambda-expressions. Practical uses of the standard API, iterator and visitor design pattern. ● Functional paradigm (the OCaml programming language): higher-order and polymorphic functions, polymorphic type inference, currying, functions on lists, user-defined types.
Sistemi di Elaborazione e Trasmissione dell' Elaborazione (Computer and Communication Systems)	● Introduction to modern operating systems ● Processes and threads; scheduling ● Memory management and virtualization ● File systems ● Computer Security ● Introduction to synchronization, deadlock ● Network technologies and 'internetworking ● Protocol Layers ● Internet stack (from application level to local network) ● Web and HTTP protocol ● File transfer ● Electronic Mail ● DNS protocol ● UDP and TCP protocols..

Algebra Lineare e Analisi Numerica (Numerical Analysis and Linear Algebra)	● Error analysis
	  ○ Floating-point numbers and machine precision.
	  ○ Inherent error. Estimate for rational functions.
	  ○ Algorithmic error.
	  ○ Total error.
	● Solution of nonsingular linear systems
	  ○ Numerical solution of linear systems (direct method of Gaussian elimination).
	  ○ Conditioning of matrices.
	  ○ Complexity and algorithmic error for the solution of linear systems.

	● Other topics in linear algebra: geometric interpretation of vectors and matrices ○ Scalar product and orthonormal bases. ○ Matrices as geometric linear transformations. ○ Null space, range and rank. ○ Orthogonal matrices: rotations, reflections, QR factorization. ● Approximated solution of linear systems in the least-squares sense ○ Geometric formulation of the problem. ○ Normal equations. ○ Solution through orthogonalization. ● Interpolation by spline functions ○ Definition of interpolating spline. ○ Computational procedure. ○ Survey of mathematical and numerical properties. ● Other topics in linear algebra: eigenvalues ○ Eigenvalues, eigenvectors, eigenspaces. ○ Characteristic polynomial. ○ Similarity relations e diagonalization. ○ Applications. ● SVD and applications to least-squares ○ Singular values decomposition (SVD) and relations with eigenvalues. ○ Geometric properties of SVD and numerical rank. ○ Generalized inverse and conditioning. ○ Solution of the least-squares problem via SVD. ○ Application to discrete data approximation (smoothing). ● Numerical treatment of eigenvalues ○ Numerical properties: conditioning and localization. ○ Iterative power method and variants. ○ Other numerical methods: similarity reduction to a simplified form, QR method.
Basi di Dati (Database Management Systems)	● Introduction to DBMS concepts. ● Traditional data models; the relational model and languages. ● SQL language: querying and manipulation of relational datai. ● Entity-Relationship model. ● Logic database design. ● Quality control: normalization. ● Data management inside applications: stored procedures; SQL usage from a programming language, triggers. ● DBMS internals; indexes and query tuning, transactions and access control. ● Physical database design ● Lab: usage of a DBMS.
Teoria dell' Informazione e Inferenza (Information	● Elements of probability theory. Probability of an event, conditional probability, Bayes theorem, discrete random

Theory and Inference)	variables, probability distributions, expected values and variance, marginal probability, asymptotic laws.. ● Elements of Information Theory. Information and entropy (Shannon). Kraft Mc Millian inequality. Information encoding without noise, Huffman Coding, Arithmetic Coding, Convolutional Encoding ● Elements of inference. Maximum likelihood estimation . Bayesian learning. Monte Carlo and Markov chains.
Analisi e Progettazione di Algoritmi (Algorithm analysis and design)	● Design and analysis techniques, asymptotic notations, correctness and complexity of recursive and iterative algorithms, divide-et-impera, dynamic programming, greedy algorithms. ● Sorting: simple sorts, mergesort, quicksort, heapsort, lower bound for comparison-based sorting algorithms, linear sorts. ● Advanced data structures: heaps, union-find structures. ● Graphs: definitions, representations, visits, topological sorting, strongly-connected components, single-source shortest paths (Dijkstra algorithm), minimum spanning tree (Prim and Kruskal algorithms). ● Theory of NP-completeness: complexity classes, NP-complete problems, approximation algorithms
Teoria degli Automi e Calcolabilita' (Automata Theory and Computability)	● Alphabets, strings, languages ● Deterministic (DFA) and non deterministic (NFA) finite state automata ● DFA minimization, regular expressions ● Closure properties of regular languages, pumping lemma ● Context-free grammars, derivation trees, ambiguity ● Deterministic and non deterministic push down automata (PDA) ● Closure properties of context-free languages, pumping lemma ● Turing machines, accepted languages ● Undecidability, recursive and recursively enumerable languages, universal language and machine, undecidability of the universal language ● Halting problem ● Undecidable problems on Turing machines, Rice theorem, Church thesis
Calculus 2	● Improper integrals. ● Sequences and series of functions. ● Functions of several variables, continuity, differentiability, derivative. ● Multiple integrals, change of variables, ● Systems of differential equations

	● Fourier series
Computazione Quantistica (Quantum Computing)	● Introduction ○ Alternative systems and techniques of computation ○ Universal logic gates ○ Using 'snooker' as a computation system ○ DNA computation ● Mathematical foundations: complex number, tensor product, unary operators ● Introduction to quantum phenomena ○ 1900: experimental results: state superposition, measurements, entangled states ○ Qubits as quantum states with two levels. Qubit superposition, qubit state representations on the 'Bloch sphere' ○ Applications of unary operators to qubits. Classical logic gates, quantum logic gates, 2-qubits logic gates (“entangling gates”), usage of Pauli matrices to describe logic gates and unary operators ● Quantum information ○ Quantum parallelism ○ “No-cloning” Theorem ○ “Dense coding” ○ Quantum teleportation ● Simple quantum algorithms ○ Deutch, Deutch-Josa and Simon algorithms ● 5. Quantum cryptography: BB protocol, cryptographic protocols implementations ● Quantum algorithms: “Black Box” algorithm, Grover algorithm
Computer Security	● Introduction Computer Security [1h] ○ The concepts of resource, vulnerability, threat, countermeasure, and risk ○ Security goals: confidentiality, integrity, availability, ... ● Introduction to Cryptography [2h] ○ Fundamental concepts (cryptography, cryptanalysis, general cryptographic schema) ○ Monoalphabetic substitution ciphers (Caesar cipher) ○ Polyalphabetic substitution ciphers (Vigenère cipher) ○ One-time pads (Vernam cipher) ○ Transposition ciphers ○ Composite ciphers ● Symmetric Cryptography [3h] ○ Block and stream ciphers ○ Feistel cipher structure ○ DES and 3DES ○ Modes of operation (Electronic Code Book, Cipher-Block Chaining, Stream Ciphers) ○ Link vs end-to-end encryption ○ The key distribution problem

	● Public-Key Cryptography [6h] ○ Introduction to public-key cryptography ○ Introduction to Number Theory ○ The RSA algorithm ○ Diffie-Hellman key exchange ● Message Authentication and Digital Signatures [3h] ○ Message integrity and authentication functions (message encryption, message authentication code, cryptographic hash functions) ○ Digital signature ● Public Key Infrastructure (PKI) [3h] ○ PKI components ○ Digital Certificates ○ Trust models ● Security Protocols [6h] ○ Basic notions (protocol execution, assumptions and goals, attacker model) ○ Examples of protocols (NSPK, Otway-Rees, Andrew Secure RPC, Denning & Sacco) ○ Prudent engineering of security protocols ○ Kerberos (architecture, protocol, inter-realm communication, limitations) ● Secure mail [3h+3h hands on] ○ PGP ● Network Security [6h] ○ Link Layer: WiFi Security ○ Network Layer: IP-Sec ○ Transport Layer: SSL/TLS ○ Introduction to Firewalls ● Web Security [6h + 6h hands on] ○ Security on the client side (cookies and privacy, HTTP authentication mechanisms) ○ Security on the server side (unvalidated input, broken authentication and session management, cross-site scripting, injection flaws, denial of service, ...) ● Secure Programming [6h] ○ Buffer overflows ○ Format string vulnerabilities ● Access Control [6h] ○ Discretionary vs Mandatory Access Control ○ Access control matrix model ○ Role-Based Access Control (RBAC) ○ Administrative Role-Based Access Control (ARBAC) ○ I modelli di Bell-LaPadula, Harrison-Ruzzo-Ullman, Chinese Wall ● Cyber Exercise [12h hands on]
Fondamenti di Ingegneria del Software (Elements of Software Engineering)	● Introduction to the course and to software engineering ● Software development process models, agile methods (eg. Extreme programming) ● Requirements Engineering. Scenarios and use cases

	● Design principles and design methods, coupling and cohesion, software architecture, design patterns, refactoring ● The UML : Class diagram, Sequence diagram, State Machine diagram, Activity diagram ● Software Testing : Testing functional / structural and coverage. Unit tests, integration, system, acceptance tests and regression tests ● Software evolution and Maintenance
Fondamenti di Elaborazione dei Segnali e Immagini (Elements of signal and image processing)	● Discrete Fourier Transformation ● Representation and processing of signals through DFT (Discrete Fourier Transform) and FFT (Fast Fourier Transform) ● Representation and processing of images through DFT (Discrete Fourier Transform) and FFT (Fast Fourier Transform) ● Sampling Noise treatment Filters and applications
Introduzione alla data science (Elements of data science)	● Introduction to Python ● Data collection and management, including pre-processing for typical real-world datasets ● Data analysis: statistical analysis, (foundations of) predictive analysis ● Visualization: GeoPandas, Plotly, NetworkX ● Distributed systems foundations: HPC and Big Data ●
Programmazione Concorrente e Algoritmi Distribuiti (Concurrent Programming and distributed algorithms)	● OS-level concurrency & multi-threading ● Concurrent development techniques ○ Lock-free synchronization (focus on 'relaxed memory' architectures) ○ Standard synchronization techniques: mutexes and barriers ○ Memory models e Sequential consistencies ○ Pthread, semaphores, monitors and synchronization barriers ● Concurrency in Java ○ Threads as objects, fundamental synchronization primitives ○ Libraries for High Level Concurrency ○ Eclipse and debugging multithreaded code on Eclipse ○ Theads, monitors, semaphores ○ Producer-consumer pattern ● GUI, Events and Concurrency ○ Swing, Javafx and Concurrency ○ Async Tasks in Android (Java and Kotlin) ● Introduction to Distributed Algorithms ● Distributed Programming in Java ○ Connection Pool (Connection Caching) ○ Client-Server with Java Sockets

Sviluppo di Applicazioni Web (Web Application Developement)	● Static Web pages creation with HTML5 and CSS. ● Server side programming: Dynamic Web page creation using PHP, syntax, data and control structures, objects. Cookies management, access control, session control. Interacting with databases via PHP scripts. Accessing databases with PDO. ● Introduction to XML and JSON. ● Client side programming: Javascript and the Document Object Model specification. Introduction to AJAX and JavaScript libraries. ● The REST (REpresentational State Transfer) architectural paradigm. ● Security, usability and accessibility in Web applications. ● Practical and final project covering all the subjects seen during the semester.