ANNEX A - STRUCTURE AND CONTENTS OF THE "LAUREA MAGISTRALE IN INFORMATICA" AT THE UNIVERSITY OF PISA

The *Laurea Magistrale in Informatica* aims at educating specialists with solid foundations in computer science and high qualification in information technologies. Starting from the academic year 2017/18 the programme is organized into four curricula, with the goal to train professionals with high specialization in key areas of computer science:

- Artificial Intelligence (AI)
- Data and Knowledge: Science and Technologies (KD)
- ICT Solutions Architect (ICT)
- Software: Programming, Principles, and Technologies (SW)

Each curriculum has its own study plan, including a set of mandatory courses and a set of elective courses to be chosen from a specific group. The courses are entirely taught in English.
In the following for each curriculum the overall structure is described, together with the group of elective courses and the contents of the mandatory courses. Next the description of the elective courses is presented. Up-to-date information on the Laurea Magistrale can be found at the follwing URL:

https://www.di.unipi.it/en/education/mcs

Curriculum "Artificial Intelligence (AI)": OVERALL STRUCTURE

FIRST YEAR	Semester	ECTS: 63
Artificial intelligence fundamentals	1	6
Computational mathematics for learning and data analysis	1	9
Machine learning	1	9
Group: AI electives 6 cfu	1	6
Human language technologies	2	6
Parallel and distributed systems: paradigms and models	2	9
Intelligent systems for pattern recognition	2	6
Group: AI electives 9 cfu	2	9
SECOND YEAR		ECTS: 57
Smart applications	1	9
Group: AI electives 9 cfu	1	9
Group: Free choice	1	9
Group: AI electives 6 cfu	2	6
Master's thesis	2	24

Group: AI elective (9 CFU)	Semester	see
Algorithm engineering	1	KD
Data mining	1	KD
Mobile and cyber-physical systems	2	ICT
Group: AI elective (6 CFU)		
Information retrieval	1	KD
Computational neuroscience	2	
Robotics	2	
Semantic web	1	

Curriculum "Artificial Intelligence" (AI): Description of Mandatory Courses

Course Title	Aims and Contents	Cred.	Sem.
Artificial intelligence fundamentals	The course aims to offer a view of the classical/symbolic approach to Artificial Intelligence and serves as a basis for more in depth treatment of specific theories and technologies for building complete A.I. systems	6	1

	integrating different approaches and methods.		
Computational mathematics for learning and data analysis	Students are expected to acquire: some knowledge of the main techniques and methods for the solution of numerical and optimization problems; some understanding of the connections between typical techniques of numerical analysis and optimization algorithms; tools for modeling (through numerical analysis and optimization) specific problems from the following areas: regression and parameter estimation in statistics, approximation and data fitting, machine learning, data mining, image and signal reconstruction.	9	1
Machine learning	The course introduces the machine learning principles and models, including basic theory of learning. The course provides the Machine Learning basis for both the aims of building new adaptive Intelligent Systems and powerful predictive models for Intelligent Data Analysis. The focus is on the critical analysis of the characteristics for the design and use of the algorithms for learning functions from examples and for the rigorous experimental evaluation. The student who successfully completes the course will be able to demonstrate a solid knowledge of the main models and algorithms for learning functions from data, with a focus on Neural Networks and related methods. The student will be aware of the general conceptual framework of modern machine learning; of the basic principles of computational learning processes; of rigorous validation techniques; of the critical characteristics for the use of the learning models to design intelligent/adaptive systems and predictive models for data analysis.	9	1
Human language technologies	The course presents principles, models and the state of the art techniques for the analysis of natural language, focusing mainly on statistical machine learning approaches and Deep Learning in particular. Students will learn how to apply these techniques in a wide range of applications using modern programming libraries.	6	2
Parallel and distributed systems: paradigms and models	The course aims at providing a mix of foundations and advanced knowledge in the field of parallel computing specifically targeting data intensive applications. A first part of the course will provide the necessary background related to the parallel hardware, from multicore to accelerators up to distributed systems such as clusters and cloud. Then the principles of parallel computing will be addressed, including measures characterizing parallel computations, mechanisms and policies supporting parallel computing and typical data intensive patterns. Eventually a survey of existing programming frameworks will be included, aimed at preparing the students to use and exploit the more modern and advanced framework currently used in both research and production institutions. As a result, the student attending the course will be given a general perspective of the parallel computing area as well as a comprehensive survey of the currently available frameworks for data intensive computing.	9	2
Intelligent systems for pattern recognition	The course introduces students to the design of A.I. based solutions to complex pattern recognition problems and discusses how to realize applications exploiting computational intelligence techniques. The course also presents fundamentals of signal and image processing. Particular focus will be given to pattern recognition problems and models dealing with sequential and time-series data. A final project will introduce students to the implementation of a pattern recognition application or to the development of computational intelligence applications.	6	2

Smart applications	The course aim is to explore methods and technologies for the development of smart connected applications, i.e. applications which exhibit intelligent behaviour -- through the use of artificial intelligence techniques introduced in other courses -- and that are deployed in immersive environments, including smart objects (as embodied by Internet of Things devices), mobile devices (smartphones, tablets), wearables (smartwatches, fitness trackers), home automation devices, web technologies, and cloud services and infrastructure. As such, applications considered for the course will include elements of context-awareness, sensor intelligence, spoken-language interfaces, The course will be based around a single case study for a novel smart application.	9	1

Curriculum "Data and Knowledge: Science and Technologies (KD)": OVERALL STRUCTURE

FIRST YEAR	Semester	ECTS: 63
Algorithm engineering	1	6
Data mining	1	9
Information retrieval	1	6
Computational mathematics for learning and data analysis	1	9
Advanced databases	2	9
Bioinformatics	2	6
Parallel and distributed systems: paradigms and models	2	9
Group: KD electives 6 cfu	2	6
SECOND YEAR		ECTS: 57
Group: KD electives 6 cfu	1	6
Group: KD electives 9 cfu	1	9
Group: Free choice	1	9
Group: ICT electives 9 cfu	2	9
Master's thesis	2	24

Group: AI elective (9 CFU)	Semester	see
Human languages technologies	2	AI
ICT risk assessment	2	ICT
Mobile and cyber physical systems	2	ICT
Machine learning	1	AI
Group: AI elective (6 CFU)		
Big data analytics	1	
ICT infrastructures	1	ICT
Peer to peer systems and blockchains	2	ICT
Scientific and large data visualization	1	
Social and ethical issues in computer technology	2	

Curriculum "Data and Knowledge: Science and Technologies (KD): Description of Mandatory Courses

Course Title	Aims and Contents	Cred.	Sem.
Algorithm engineering	The student who successfully completes the course will have the ability to design and analyze (theoretically and experimentally) advanced algorithms and data structures for the efficient solution of combinatorial problems involving all basic data types, such as integers, strings, (geometric) points, trees and graphs. These algorithmic tools will be designed and analyzed in several models of computation— such	6	1

	as RAM, 2-level memory, cache-oblivious, streaming— in order to take into account the architectural features and the memory hierarchy of modern PCs.		
Data mining	The formidable advances in computing power, data acquisition, data storage and connectivity have created unprecedented amounts of data. Data mining, i.e., the science of extracting knowledge from these masses of data, has therefore been affirmed as an interdisciplinary branch of computer science. Data mining techniques have been applied to many industrial, scientific, and social problems, and are believed to have an ever deeper impact on society. The course objective is to provide an introduction to the basic concepts of data mining and the process of extracting knowledge, with insights into analytical models and the most common algorithms. Morover, the course will present advanced techniques which are variants of the basic techniques and will discuss the algorithmical aspects.	9	1
Information retrieval	The student who successfully completes the course will have the ability to design a simple search engine and/or one of the numerous IR tools which are at the core of modern Web applications. The syllabus includes study, design and analysis of IR systems which are efficient and effective to process, mine, search, cluster and classify documents, coming from textual as well as any unstructured domain.	6	1
Computational mathematics for learning and data analysis	See Curriculum AI	9	1
Advanced databases	Data management systems occupy a central position in our information-based society, and computer scientist and database application designers should have a good knowledge about both the theoretical and the engineering concepts that underline these systems to ensure the application performance desired. The student who completes the course successfully will be able to demonstrate advanced knowledge of the main issues related to the implementation of classical centralized relational database and of modern data managemente systems, in order to be a sophisticated user of data management technology and a high-performance applications developer.	9	2
Bioinformatics	This course has the goal to give the student an overview of algorithmic methods that have been conceived for the analysis of genomic sequences. We will focus both on theoretical and combinatorial aspects as well as on practical issues such as whole genomes sequencing, sequences alignments, the search of patterns in biological sequences, the inference of repeated patterns and of long approximated repetitions, the computation of genomic distances, and several biologically relevant problems for the management and investigation of genomic data.	6	2
Parallel and distributed systems: paradigms and models	See Curriculum AI	9	2

Curriculum "ICT Solutions Architect" (ICT): OVERALL STRUCTURE

FIRST YEAR	Semester	ECTS: 63
Advanced programming	1	9
Advanced software engineering	1	9
Algorithm engineering	1	9
ICT infrastructures	1	6
ICT risk assessment	2	9
Mobile and cyber-physical systems	2	9
Peer to peer systems and blockchains	2	6
Group: ICT electives 6 cfu	2	6
SECOND YEAR		ECTS: 57
Group: Free Choice	1	9
Group: ICT electives 9 cfu	1&2	18
Group: ICT electives 6 cfu	1	6
Master's thesis	2	24

Group: ICT elective (9 CFU)	Semester	see
Data mining	1	KD
Machine learning	1	AI
Parallel and distributed systems: paradigms and models	2	AI
Software validation and verification	1	SW
Group: ICT elective (6 CFU)		
Information retrieval	1	KD
Intelligent systems for pattern recognition	2	AI
Scientific and large data visualization	1	
Security methods and verification	2	

Curriculum "ICT Solutions Architect" (ICT): Description of Mandatory Courses

Course Title	Aims and Contents	Cred.	Sem.
Advanced programming	The course aims to provide the students with a deep understanding of how high level programming concepts and metaphors map into executable systems and which are their costs and limitations; to acquaint the students with modern principles, techniques, and best practices of sophisticated software construction; to introduce the students to techniques of programming at higher abstraction levels, in particular component programming and functional programming; to present state-of-the-art frameworks incorporating these techniques.	9	1
Advanced software engineering	The course aims to introduce some the main aspects in the design, analysis, development and deployment of modern software systems. Service-based and cloud-based systems are taken as references to present design, analysis and deployment techniques. DevOps practices are discussed, and in particular containerization is introduced. The course includes a "hands-on" lab where students will experiment weekly the design, analysis, development and deployment techniques introduced.	9	1
Algorithm engineering	See curriculum KD	9	1
ICT infrastructures	The course aims to introduce students to the computing infrastructures powering cloud services. At the end of the course a student should be able to understand the general organization of a datacenter and the	6	1

	logical infrastructure that power virtualization and containers. The course starts from physical infrastructures such as power and datacenter organization. The network fabric is introduced, with particular focus on SDN techniques used to balance East-West and North-South traffic. Storage and compute are then introduced with special attention to hyperconverged systems.		
ICT risk assessment	At the end of this course, the student should be able to discover and analyze the weaknesses and the vulnerabilities of a system to evaluate in a quantitative and formal way the risk it poses. The student should be able to select and deploy a cost-effective set of countermeasures at the various implementation levels to improve the overall ability of the system to withstand its attackers. Focus of the course is on a predictive approach where risk assessment and management is a step in the system design. The student should also be able to know the various tools that can support the assessment and simplify both the assessment and the selection of countermeasures. In this framework, the focus on cloud computing makes it possible to cover the most complex assessment.	9	2
Mobile and cyberphysical systems	The course covers mobile and cyber-physical systems by providing an overview of issues, solutions, architectures, technologies and standards. It offers to the students an overall, coherent view of the organization of IoT systems, from the networking and sensing levels to the applications. Specifically, it shows how mobile, heterogeneous elements (from low-end sensors to high-end devices) form pervasive networks integrated in the internet and how they interact among themselves and with the surrounding physical world.	9	2
Peer to peer systems and blockchains	Introduction of the basic technologies for the development of highly distributed systems and of some real scenarios exploiting them. Presentation of the disruptive technology of blockchains, and its numerous applications to different fields.	6	2

Curriculum "Software: Programming, Principles, and Technologies" (SW): OVERALL STRUCTURE

FIRST YEAR	Semester	ECTS: 63
Languages, compilers and interpreters	1	9
Competitive programming and contests	1	6
Algorithm design	2	9
Principles for software composition	2	9
Group: SW elective 9 cfu	1&2	18
Group: SW elective 6 cfu	1&2	12
SECOND YEAR		ECTS 57
Software validation and verification	1	9
Laboratory for innovative software	2	6
Group: ICT electives 9 cfu	1	9
Group: Free choice	1	9
Master's thesis	2	24

Group: SW elective (9 CFU)	Semester	see
Advanced programming	1	ICT
Advanced software engineering	1	ICT
Computational mathematics for learning and data analysis	1	KD
Machine learning	1	AI

Mobile and cyberphysical systems	2	ICT
Parallel and distributed systems: paradigms and models	2	AI
Smart applications	1	AI
Group: SW elective (6 CFU)		
Bioinformatics	2	KD
Foundations of computing	2	
ICT infrastructures	1	ICT
Information retrieval	1	KD
Security methods and verification	2	

Curriculum "Software: Programming, Principles, and Technologies" (SW): Description of Mandatory Courses

Course Title	Aims and Contents	Cred.	Sem.
Languages, compilers and interpreters	The student who successfully completes the course will be able to contribute to the design and implementation of a modern programming language. The syllabus includes Regular and context-free languages, Recognizers: scanners and parsers, Contextual analysis, Intermediate representations, Symbol table: representation and handling, Functional and procedural abstractions, Static analysis (data flow, control flow, ...), Optimizations and Code generation.	9	1
Competitive programming and contests	The goal of the course is to improve programming and problem solving skills of the students by facing them with difficult problems and by presenting the techniques that help their reasoning in the implementation of correct and efficient solutions. The importance of these skills has been recognized by the most important software companies worldwide, which evaluate candidates in their job interviews mostly by the ability in addressing such difficult problems. A natural goal is to involve the students in the intellectual pleasure of programming and problem solving, also preparing them for the most important international online contests, for internships in most important companies and their interviews.	6	1
Algorithm design	The course focuses on developing algorithmic design skills, exposing the students to complex problems that cannot be directly handled by standard libraries (being aware that several basic algorithms and data structures are already covered by the libraries of modern programming languages), thus requiring a significant effort in problem solving. These problems involve all basic data types, such as integers, strings, (geometric) points, trees and graphs, as a starting point and the syllabus is structured to highlight the applicative situations in which the corresponding algorithms can be successfully applied. Brainstorming activities will be central to help students learning from their mistakes.	9	2
Principles for software composition	This course introduces concepts and techniques in the study of advanced programming languages, as well as their formal logical underpinnings. The central theme is the view of individual programs and whole languages as mathematical entities about which precise claims may be made and proved. The course will cover the basic techniques for assigning meaning to programs with higher-order, concurrent and probabilistic features (e.g., domain theory, logical systems, well-founded induction, structural recursion, labelled transition systems, Markov chains, probabilistic reactive systems) and for proving their fundamental properties, such as termination,	9	2

	normalisation, determinacy, behavioural equivalence and logical equivalence. In particular, some emphasis will be posed on modularity and compositionality, in the sense of guaranteeing some property of the whole by proving simpler properties of its parts. Emphasis will be placed on the experimentation of the introduced concepts with state-of-the-art tools.		
Software validation and verification	The goal of the course is to introduce techniques for verifying and validating software properties, either by analysing a model extracted from a program with model checking, or by testing the software before (the next) deployment, or equipping the running software with tools that monitor its execution.	9	1
Laboratory for innovative software	Practical development of software requires an understanding of successful methods for bridging the gap between a problem to be solved and a working reliable software system. This course will train the student to develop large software systems working in real projects by exploiting the techniques and the skills acquired in the fundamental courses of the curriculum.	6	2

Description of Elective Courses

Course Title	Aims and Contents	Cred	Sem
Big data analytics	This course is meant to put at work the many data analytics technologies and competences: data mining, machine learning, social network analytics, visual analytics in realizing a whole big data analytics project: from acquiring and analyzing big data from multiple sources to the purpose of discovering the patterns and models that explain certain phenomena, till the validation and presentation of the discoveries. The students will be exposed to experience in different domains: mobility and transportation, urban planning, demographics, economics, social relationships, opinion and sentiment, etc.; and on the analytical and mining methods that can be used.	6	1
Computational neuroscience	The objectives of the course include bio-inspired neural modelling, spiking and reservoir computing neural networks, advanced computational neural models for learning, architectures and learning methods for dynamical/recurrent neural networks for temporal data and the analysis of their properties, the role of computational neuroscience in real-world applications (by case studies).	6	2
Foundation of Computing	Students are expected to learn the essential properties of some widely employed models of computation for higher order, concurrency, interaction, mobility. Algebraic semantics and elementary category theory are employed.	6	2
Robotics	The course introduces the fundamentals of robotics, viewed as an application domain for computer science, intelligent systems, and machine learning; provide students with the basic tools to integrate and program a robotic system, with special attention to the realization of perception-action schemes and behaviour control; improve students' experimental work capacity, through the analysis of case studies and laboratory work.	6	2
Scientific and large data visualization	Scientific Visualisation is an area concerned with the visualisation of large and complex data sets, where the data might come from experiments or computations. Visualisation is a way, in many cases the only possible way, to achieve insight and knowledge inside large structured amount of data.	6	1

	The course will discuss discrete models for data representation in low dimensional spaces, scalar and vectorial data in 2D, 3D and for temporal series and algorithms for processing and visualizing massive datasets.		
Security methods and verification	The student who successfully completes the course will be able to demonstrate a good knowledge of security issues that arise in Computer Science and to have an idea of how formal methods can help in addressing them. Furthermore the student will have the ability to read and understand a research paper on formal methods applied to security.	6	2
Semantic web	The course presents Semantic web technologies, making the student able to design and implement knowledge bases based on ontologies encoded with Semantic Web languages, and offered access as Linked Data.	6	1
Social and ethical issues in computer technology	The course aims to provide a thorough overview of the many ethical and social issues raised by computer technology, with particular attention to Artificial Intelligence and its multifarious impact on society and human existence. Students will learn about the most compelling social and ethical challenges posed by information technologies and how to approach them in a rigorous and critical fashion. Conceptual analysis will be supported by discussion of practical case studies.	6	2