University of Camerino

Faculty of Science and Technologies

Class no. 26

First Level Degree Course in Information Technology

COURSE INTRODUCTION

This course is intended to graduate students able to provide I.T. solutions to technical and management problems in industry, finance and public administration. To this aim, a theoretical background and practical experience, a knowledge of economics and legislation regarding the new technologies as well as the use of suitable mathematical modelling techniques are required.

This course in Information Technology is divided into two main branches:

1. INFORMATION TECHNOLOGY (I.T.)
2. I.T. AND MANAGEMENT (I.M.)

the purpose of which is to train two distinct types of I.T. professionals whose future employment will be either as a technical expert in I.T. or as an expert in I.T. applications and management. Students completing the course will obtain a First Level Degree in Information Technology.

COURSE ORGANISATION

Three years timetable. Every academic year is divided into two semesters (terms) in which periods of lectures and periods of examinations occur.

The first semester starts within the 15th October of each year and terminates at the end of February. It is organized as follows:

- 6 weeks of lectures
- 2 weeks for reading/examinations (approximatively end of November/beginning of December)
- 6 weeks of lectures (Christmas holidays excluded)
- 4 weeks for examinations (February)

The second semester starts at the beginning of March and terminates at the end of July.

It is organized as follows:

- 6 weeks of lectures
- 2 weeks for reading/examinations (in April)

- 6 weeks of lectures (until approximatively the middle of June)
- 6 weeks for examinations (2nd part of June, July)

August is period of vacations and there are 5/6 weeks for a supplementary session of examinations in September/1st part of October.

AVAILABLE COURSES

The student must acquire 180 ECTS (European Credits Transfer System) to obtain the Degree. The courses are distributed over the three years as shown in the following table:

Modules	Year	Semester	Lectures Hours		Individual Study Hours	ECTS
			Lessons	Excercise		
Programming	I	I	24	18	108	6
Programming – Lab	I	I	24	18	108	6
Mathematical Analysis	I	I	24	18	108	6
Discrete Mathematics	I	I	24	18	108	6
Computer Architecture	I	II	24	18	108	6
Computer Architecture – Lab	I	II	24	18	108	6
Mathematical Logics	I	II	24	18	108	6
Foundations of Computer Science	I	II	24	18	108	6
Physics	I	II	24	18	108	6
Private Law	I	II	24	18	108	6
English 0	I	I,II				2
English 1	I	I,II				3
English 2	I	I,II				4
Algorithms and Data Structures	II	I	24	18	108	6
Algorithms and Data Structures – Lab	II	I	24	18	108	6
Computer Networks	II	I	24	18	108	6
Economic Politics	II	I	24	18	108	6
Probability and Mathematical Statistics	II	I	24	18	108	6
Operative Research	II	I	24	18	108	6
Operating Systems	II	II	24	18	108	6
Operating Systems – Lab	II	II	24	18	108	6
Databases	II	II	24	18	108	6
Databases – Lab	II	II	24	18	108	6
Programming Languages and Compilers	II	II	24	18	108	6
Software Engineering	III	I	24	18	108	6
Software Engineering – Lab	III	I	24	18	108	6

Combinatorial Optimization	III	I	24	18	108	6
Numerical Analysis	III	I	24	18	108	6
Economics	III	I	24	18	108	6
Economics and Management	III	I	24	18	108	6
Economic Policy	III	I	24	18	108	6
Commercial Law	III	I	24	18	108	6
Economics and Management	III	I	24	18	108	6
Laboratory of Computer Networks	III	I	24	18	108	6
Web Technologies	III	I	24	18	108	6
Discrete Events Systems Simulation	III	II	24	18	108	6
Network Policies and Security	III	II	24	18	108	6
Enterprise Statistics	III		36		89	5
Enterprise Strategies	III		24	18	108	6
e-Commerce	III	I	24	18	108	6
Computational Graphics	III	II	36		89	5
Systems of Automation and Laboratory	III	I	45	30	225	12
Information Technology Law	III	I	40		100	6
Digital Economy Law	III	I	24	18	108	6
Mathematical Models for Enterprise Management	III	I	24	18	108	6
Codes and Cryptography	III	I	36		89	5
Enterprise Information Systems	III	II	45	30	225	12
C Programming	III	II	24	18	108	6

Language of Teaching

Currently all courses are in Italian. Courses in English will be activated in the near future. Tutoring and examinations can be currently done in English.

CURRICULA

The First Level Degree Course in Information Technologies has two curricula, corresponding to the two branches of the course:

1. Information Technology
2. Information Technology and Management

The following tables show the distribution of the ECTS in the two curricula.

Curriculum: Information Technology

Year 1	ECTS	Semester
Programming+Lab	6+6	1
Mathematical Analysis	6	1
Discrete Mathematics	6	1
Computer Architecture + Lab	6+6	2
Mathematical Logics	6	2
Foundations of Computer Science	6	2
Physics	6	2
English	9	(*)

(*) English courses are organized by the Faculty. Lectures (level 0, 1, 2) are distributed over the two semesters.

Year 2	ECTS	Semester
Algorithms and Data Structures +Lab	6+6	1
Computer Networks	6	1
Probability and Mathematical Statistics	6	1
Operational Research	6	1
Operating Systems + Lab	6+6	2
Databases + Lab	6+6	2
Programming Languages and Compilers	6	2

Year 3	ECTS	Semester
Software Engineering + Lab	6+6	1
Student choice (**)	6	1
Economics and Management	6	1
Student choice	12	2
Stage	12	2
Final examination	9	2

(**) Students may choose between Combinatorial Optimization and Numerical Analysis

Curriculum: Information Technology and Management

Year 1	ECTS	Semester
Programming+Lab	6+6	1
Mathematical Analysis	6	1

Discrete Mathematics	6	1
Computer Architecture + Lab	6+6	2
Private Law	6	2
Foundations of Computer Science	6	2
Physics	6	2
English	9	(*)

(*) English courses are organized by the Faculty. Lectures (level 0, 1, 2) are distributed over the two semesters.

Year 2	ECTS	Semester
Algorithms and Data Structures +Lab	6+6	1
Economic Politics	6	1
Probability and Mathematical Statistics	6	1
Operational Research	6	1
Operating Systems + Lab	6+6	2
Databases + Lab	6+6	2
Programming Languages and Compilers	6	2

Year 3	ECTS	Semester
Software Engineering + Lab	6+6	1
Student choice (**)	6	1
Economics and Management	6	1
Student choice	12	2
Stage	12	2
Final examination	9	2

(**) Students may choose between Combinatorial Optimization and Numerical Analysis

Courses of 12 CFU with Lab are considered unique courses composed by two modules. However, Erasmus students can take only one of the two modules of 6 CFU.

USEFUL INFORMATION

President of the Degree Course

Prof. Emanuela Merelli

Phone + 39 0737 402567

Mobile: +39 320 4381108
e-mail: emanuela.merelli@unicam.it

Tutoring Service

Dott. Pasini Leonardo
Phone: +39 0737 402562
e-mail: pasini.leonardo@unicam.it

Course Internationalization staff

Prof. Emanuela Merelli
Phone + 39 0737 402567
Mobile: +39 320 4381108
e-mail: emanuela.merelli@unicam.it

Dott. Luca Tesei
Phone: +39 0737 402572
Email: luca.tesei@unicam.it

Internationalization Office

ASSICOS
Via Le Mosse 22
62032 Camerino (MC)
Phone: +39 0737 404610 - +39 0377 404600
Email: relazioni.internazionali@unicam.it

How to enrol

Applications for enrolment should be addressed to the Rector and be written on official stamped paper and should be sent or delivered to the "Area Accademica e Didattica dell'Università" between the 15 th of July and the31st of October of each Academic Year.

Location

Department of Mathematics and Information Technology
Polo Informatico
Via Madonna delle Carceri 9
62032 Camerino (MC)

Phone: +39 0737 402565
Fax: +39 0737 402561
E-mail: emanuela.merelli@unicam.it
Web site: **http://www.cs.unicam.it**

Department

Department of Mathematics and Informatics
Via Madonna delle Carceri, 9
62032 Camerino (MC)
Phone: +39 0737 402549
Email: dipartimento.matematica@unicam.it
President: Prof. Flavio Corradini
Phone: +39 0737 402564
Mobile: +39 320 4381177
Email: flavio.corradini@unicam.it

Faculty Secretariat

Via Gentile III da Varano
62032 Camerino (MC)
Phone: +39 0737 402126
Mobile: +39 320 8604294
Email: segreteria.scienze@unicam.it

Students' Secretariat

Via Le Mosse 69
62032 Camerino (MC)
Phone: +30 0737 633517

COURSES DESCRIPTION

Algorithms and Data Structures

Course goals:

- Analysing the main techniques for designing algorithms
- Tackling in an integrated way classification, analysis, design, and development of

algorithms.

- Identifying the fundamental algorithmic choices and evaluating their costs in terms of computational efficiency.
- Choosing and developing suitable data structures for the problem to deal with.
- Reaching adequate compromises between conflicting requirements (cost, easiness, efficiency).

Needed knowledge for reaching the goals:

- Basic techniques for computational complexity analysis. Asymptotic analysis.
- Project techniques (divide-et-impera, greedy, etc.)
- Analysis of well-known basic algorithms and most utilized data structures.
- Fundamental algorithms (search, ordering, etc.)
- Graphs: representation, visit algorithms.
- Trees: visit, binary search tree.
- Algorithms on graphs (minimum path, minimum covering tree, etc.)

Algorithms and Data Structures – Lab

Course goals:

- Classifying fundamental data structures.
- Confronting theoretical and experimental results related to algorithms' behaviour.

Needed knowledge for reaching the goals:

- Fundamental data structures: vectors, matrices, lists. Algorithms for manipulating them: search, insert, delete.
- Fundamental data structures: heaps, queues. Algorithms for manipulating them: insert, delete.
- Programming data structures and comparison between theoretical and experimental results.

C Programming

Course goals:

- Knowing the C Programming Language
- Programming in C
- Gives the expertise for the autonomous study of advanced programming languages

Needed knowledge for reaching the goals:

- Introduction to programming languages: main principles, historical introduction. Functional, declarative and logic languages.

- The C Programming Language. Organization of a C program, types, operators and expressions, variables and constants, declarations, arithmetic and logic operators, type conversions, conditional expressions. Control statements, instructions and blocks, if-else, else-if, do, while and for cycles, switch. Functions, parameters passing, external variables, scoping rules, header files, static variables, recursion
- Pointers and vectors, pointers and addresses, pointers and structures, recursive structures. Input and output (mention): standard input and output, printf, scanf, getchar, getc, putchar, putc, gets, puts, access to a file (main concepts), reading and writing a file

Codes and Cryptography

Course goals:

- Introducing problems of classical cryptography and public key cryptography.
- Presenting the main public key cryptographic systems and their related theoretical fundaments, either in Number Theory and in Complexity.
- Listing the possible attacks and the corresponding security techniques.

Needed knowledge for reaching the goals:

- Classical cryptography: Julius Caesar, Vigenère, Enigma. Perfect crypto-systems. Outline of the Shannon theorem. Public key cryptography. One way functions. Possible one way function sources in Number Theory and in Theoretical Computer Science.
- Recall on P = NP problem and on NP-complete languages. The bag problem.
- Prime numbers and factorization in Number Theory.
- Prime numbers algorithms: Miller-Rabin and Solovay-Strassen probabilistic procedures, Agrawal-Kayal-Saxena deterministic polynomial procedure; outline of Shor quantum algorithm.
- Factorization algorithms: p-1 method, rho method, Fermat factorization.
- On public key cryptography: Diffie-Hellman crypto-system, Massey-Omura crypto-system and ElGamal crypto-system, crypto-systems based on the bag problem, RSA crypto-system, digital signature methods. Zero knowledge protocols, telephonic toss, telephonic poker. Possible attacks to the RSA crypto-system: random hardware/software errors, coding and decoding time acceleration, smart cards usage, errors caused by an hacker.

Combinatorial Optimization

Courses goals:

- Classifying and relating different modelling and algorithmic approaches to problems of combinatorial optimization, given their computational cost and the accuracy of solutions
- Showing the main characteristics of heuristics and meta-heuristics for combinatorial optimization problems
- Identifying valid inequalities

- Showing the main characteristics of network optimization algorithms

Needed knowledge for reaching the goals:

- Combinatorial optimization models
- Recalls of computational complexity
- Classical problems of combinatorial optimization
- Valid inequalities
- Classes of approximated algorithms for combinatorial optimization "hard" problems
- Examples based on classical problems
- Optimization problems on a network

Commercial Law

The course purpose is the acquisition of professionalism intended to support entrepreneurial choices from a juridical point of view. At the end of the course the student should be able to explain the ways that best permit to reach the entrepreneurial goal and to verify the juridical tools that best fit such a goal.

Needed knowledge for reaching the goals:

- Person and market.
- Relationship between law and market; relationship between private enterprise, market and person through the analysis of the function of the market.
- Contracts.
- Credit instruments.

Computational Graphics

Course goals:

- Managing graphical primitives usually available of graphical systems for raster devices and presenting I/O devices.
- Managing modelling and projection geometric transformations
- Managing basic techniques for representing curves and surfaces and for developing solid models.
- Presenting basic concepts of colorimetry and managing some simple techniques for scene lightning.
- Using the graphical library OpenGL

Needed knowledge for reaching the goals:

- Course introduction: what is computational graphics and when to use it.

- Output primitives, input primitives, hardware devices.
- Real coordinates and logic coordinates. "Window-to-viewport" transformation. The clipping problem, an example: Cohen-Sutherland algorithm.
- Short essay on vector analysis.
- Affine transformations.
- Projective transformations.
- Curves and surfaces description: implicit form and parametric form. Normal vector.
- Curves and surfaces description: interpolation and approximation. Bezier curves and surfaces. B-spline curves and surfaces and their properties. NURBS curves and surfaces and their properties.
- Curves and surfaces approximation using graphical primitives. Polygonal and mesh examples. Mesh properties. Polyhedron and Eulero formulas. Data structures for describing a mesh.
- Solid modelling. Dragging schemes and corresponding mesh definition. Spatial decomposition schemes: cell decomposition, spatial enumeration, efficient spatial enumeration (quadtree, octree). Structural solid geometry.
- Leveraging the realism of a scene: deletion of hidden surfaces, lightning calculation.
- Colours theory.
- OpenGL library: window view, window resize, mouse, keyboard. Recording event management functions and their usage. "View port" definition e and background cleaning. Graphical primitives and vertices description. Projection matrix management and view volume management, parallel projection example. Modelling matrix management and view matrix management. Central transformations and number of point at infinity management. Animations and double buffer usage: moving scenes generation and observer moment. Source of light management and object material management. Colour management.

Computer Architecture

Course goals:

- Explaining the structure of a computer, how it works, its hardware components.
- Analysing thoroughly the fundamental design choices for system architecture, with particular emphasis on parallelism at various levels.

Needed knowledge for reaching the goals:

- Tier systems
- Principles of parallelism and cooperation, performance.
- Processes, address spaces, minimum nucleus, processors, memory hierarchy, communication, devices.
- Firmware structure and communication, CPU architecture, I/O architecture.

- Multi-tier memory and parallelism of instructions.
- Multiprocessors and multi-calculators.

Computer Architecture – Lab

Course goals:

- Knowing how computers represent and manipulate information.
- Presenting system programming principles and techniques, with a particular emphasis on firmware machine and assembler.

Needed knowledge for reaching the goals:

- Microprogramming languages features, with respect to a specific case study.
- Assembler programming languages features, with respect to a specific case study.
- Development of microprograms and assembler programs.

Computer Networks

Course goals:

- Showing the foundations of modern networks of computers and giving the needed knowleges for programming and analysing distributed networks of computers

Needed knowledge for reaching the goals:

- Transmissions media, low level protocols
- Communication protocols, routing and transport
- Services and supports for network applications
- Programming inter-operative applications
- Remote programming paradigms

Databases

- Database management systems.
- Data models.
- Database query languages.
- Logical and conceptual database design.
- Database management systems evolution.

Databases – Lab

Course goals:

- Describing DBMS architecture
- Trying main techniques for designing and managing database systems, with reference to the SQL language.

Needed knowledge for reaching the goals:

- DBMS Architecture.
- Databases development.

Digital Economy Law

Course goals:

- Understanding the main contractual typology used by software houses.
- Understanding the main national, EU, and international rules concerning on line service providing.
- Acquiring the fundamental categories and juridical notions concerning e-commerce.

Needed knowledge for reaching the goals:

- Analysis of EU normative regarding software.
- Analysis of EU normative regarding copyright protection in Internet.
- Analysis of EU and international normative regarding e-commerce.
- Main contractual models used by Internet Providers.
- Netiquette and computerized transactions.

Discrete Events Systems Simulation

Course goals:

- Knowing how to use a specification based on queues for describing traffic systems
- Knowing how to construct discrete events simulators of traffic in communication systems and software systems. Application of this technique to construct a simulator of vehicular traffic systems

Needed knowledge for reaching the goals:

- Qnap2 language
- Models of software systems
- Models of communication protocols at connection level
- Models of protocol of level 2, response elements
- Models of protocol of network level

Discrete Mathematics

Courses goals:

- Acquiring the concepts and basic tools of arithmetic and combinatorial calculus used in Computer Science
- Showing the basic concepts of linear algebra and affine geometry

Needed knowledge for reaching the goals:

- Equivalence relations and quotients
- Combinatorial calculus
- Arithmetic: division, greatest common divisor, lowest common multiple, prime numbers, remainder classes, polynomials.
- Induction techniques
- Algebra notions: groups, rings, morphisms
- Linear equations systems, matrices, determinants, vectorial spaces

e-Commerce

Course goals:

- Understanding the various types of e-Commerce and acquiring the ability to design an e-Commerce solution, either in B2B area and in B2C area.
- Forming the student on the analysis phase regarding the organizational evolution that companies need in order to enter the e-business world.

Needed knowledge for reaching the goals:

- The various e-commerce business models (B2B, B2C, C2C) and how they are implemented (e-stores, e-brokerage, auctions, e-trading centres, and virtual communities).
- The main characteristics of e-commerce (availability, ubiquity, globalism, multimediality, etc.); analysis of some of the most significant e-commerce solutions presently on the net (Marshall Industries, Fedex, Industri.net, Amazon, Tradezone, Infomar, etc.)
- Marketing programs and strategies applied to e-commerce.
- Structural modifications in B2B companies.
- Web site usability.
- E-procurement: the possibility, given by e-commerce to public bodies, to supply themselves through the net in order to have a more efficient purchase process.
- Project techniques.

Economics

Course goals:

- Providing a picture of the fundamental principles regarding the theory of consumer and enterprise, the operation of the market, as well as a basic knowledge of the main macroeconomics aggregates by means of analysis of consumption, savings, effective demand, and currency.

Needed knowledge for reaching the goals:

- Microeconomics and macroeconomics: subjects, needs, goods.
- Main schools of economic thought.
- Consumer theory: utility, indifference curves, budget constraint and consumer equilibrium
- Supply factors and production function: isocost, isoproduct, and producer equilibrium, cost analysis.
- Kinds of market: concurrence, monopoly, oligopoly.
- The distributive problem: wage, profit, income, interest.
- Consumption and saving functions; investment analysis according to Keynes and to neoclassical theory; effective demand and multiplier.
- Money: monetary base, monetary tools, quantitative theory, Keynes' preference for liquidity.
- AD-AS model. Neoclassical synthesis and IS-LM model: implication on economic policy.
- Inflation, relationship between unemployment and inflation.

Economics and management

Enterprise is an institution absolving two fundamental tasks for the organization and the development of an economic system: goods and services production, and the introduction of new products and productive processes.

This course analyses the various dimensions of an enterprise and takes into account the relationship between enterprises and the economic-institutional surrounding environment, which is constituted by technology, competitors, buyers, goods and services providers, and public institutions. The educational goals are the following:

- Providing the microeconomics foundations for enterprise's behaviour;
- Offering the basic tools for managing production, marketing, and innovation;
- Showing how the various organizing structures work and the relationship between strategies and structure.

Needed knowledge for reaching the goals:

- Rational behaviour: microeconomics and game theory.
- Enterprise as connection of contracts.
- Enterprise strategies: vertical integration and differentiation.
- Innovation.

- Internalization.

Economics Politics

Course goals:

- Giving a landscape of the fundamental principles aiming public choices and analysing the the role of these choices in the enterprises strategies

Needed knowledge for reaching the goals:

- Function of social welfare, market failures and public intervention
- Theory of the Economics Politics
- The economic policy in a closed system: goals, microeconomic and macroeconomic policies
- The economic policy in an open system: policies for the balance of payments and the commercial policies
- Private and public international Institutions

Enterprise Statistics

Course goals:

- Giving the basic descriptive and inferential statistical tools for studying data internal or external to the enterprise. In particular: quality control, analysis of the efficiency and of the productivity of the enterprise, market analysis

Needed knowledge for reaching the goals:

- Methodological bases: descriptive and inferential statistics
- Analysis of groups as market study
- Quality control inside enterprises
- Analysis of the efficiency and of the productivity using statistical indexes
- An application: an analysis of the productivity of Italian artisan enterprises

Enterprise Strategy

Course goals:

- Introducing the strategy problem: how carrying out a chosen strategy derives from complex processes of strategies formation
- Analysing markets and competitors in the point of view of the economy and the management of the enterprise
- Showing the picture of the elements of the analysis of competitive benefit: analysis of the components, external sources, internal sources

- Showing the Resource Based approach underlying the importance of the role of the resources and competences as a base for formulating the strategy
- Showing the models needed to understand the competition between products that are technologically characterised by direct externalities; technological wallet choices and relations between lead users and producers in developing new products
- Analysing the impact of technological innovation on the the sources of competitive benefit. Analysis of the development processes of new products: how the marketing activity influences the development of new products

Needed knowledge for reaching the goals:

- The strategy concept: the different approaches to the strategy, the difference between group strategy and business strategy, the studies on the enterprise strategies
- The nature and the sources of the competitive benefit: its nature, defence, types
- The cost benefit: experience economies, the origin of the cost benefit
- The benefit of differentiation: the nature and the benefits of differentiation, the analysis of the differentiation, supply and demand
- Business strategies in different competitive contexts: the evolution of the area, structure, competition and success factors
- Product innovation: the concept of product, the innovation of the product, the innovation of the product and the competitive benefit; marketing and the development process of a new product: objectives and phases, strategies in the commercialization of new products, the diffusion of the innovation; technological innovation and competitive context: innovation, competition analysis and technological placing; network externalities, installed base and competition, technological discontinuity: nature, sources and competitive impact, prediction techniques and technological monitoring techniques

Foundations of Computer Science

Course goals:

- Discussing the concept of computability: what is really computable?
- Presenting the different theoretical approaches to computability and confronting them.
- Showing the existence of non-solvable problems.
- Discussing the concept of computational complexity: what is computable at a reasonable cost?

Needed knowledge for reaching the goals:

- The problem of computability
- Various theoretical models of computability: Turing machine, recursive function. Equivalence between models. Church-Turing thesis and its discussion. Examples of non-computable function and of non-solvable problems.

- Context-dependent grammars, context-free grammars, regular grammars. Relationships with Automata Theory.
- Computability and programming languages.
- Outline of the problem P = NP.

Information Technology Law

Course goals:

- Describing the general picture concerning the present UE structure and the role, the composition, and the functions of UE institutions.
- Explaining the basic notions regarding: procedures of adoption of EU normative acts, different kinds of act and differences regarding their range and effectiveness.
- Explaining the most relevant aspects of the main EU normative acts concerning the juridical regulation of Information Society, with particular emphasis on intellectual property, privacy, and e-signature.
- Describing the main consequences on Italian law of EU normative regarding information technology.

Needed knowledge for reaching the goals:

- Basic notions of EU law: history, institutions, normative sources.
- EU directives regarding software protection, databases protection, attuning of intellectual property in the information society, outlines on the main effects on Italian law.
- EU directives regarding privacy protection, outlines on the main effects on Italian law.
- EU directives regarding electronic signature, outlines on the main effects on Italian law.

Laboratory of Computer Networks

Course goals:

- Using the main network tools
- Learn how to build and maintain a computer network
- Train students to CCNA certification examination of the CISCO ACADEMY programme, required by the major enterprises in the area.

Needed knowledge for reaching the goals:

- Base principles of computer networks
- Transmission means and network appliances
- Routing, routers configuration, RIP routing protocol, Access Control Lists
- Computer networks analysis and configuration tools

Mathematical Analysis

Course goals:

- Providing fundamental concepts e techniques for infinitesimal calculus and integral calculus.
- Using and applying the concepts of limit, derivative, and integral, also to problems which are not mathematically formalized.

Needed knowledge for reaching the goals:

- Coordinates, plane loci, functions.
- Sequences and remarkable limits.
- Continuity and differential calculus.
- Integration.

Mathematical Logics

Courses goals:

- Acquiring the basics of propositional and first order logic
- Applying the methods of Logics in Computer Science (in writing and verifying programs, etc.)
- Introduce problems without solutions or without an efficient solution, evaluate the complexity of a problem

Needed knowledge for reaching the goals:

- Predicative logics, propositional logics, logical calculi, natural deduction, models and theories

Mathematical Models for Enterprise Management

Courses goals:

- Defining linear programming and discrete linear programming models; using specific software to solve them
- Interpreting the results obtained by the solution of a linear programming model
- Identifying structural components, variables and constraints of a model; using tools to verify the correctness of the model
- Defining the main linear programming models for problems of production planning and stock management

Needed knowledge for reaching the goals:

- Linear programming: optimality conditions, duality and analysis of sensibility.
- Software for specification and solution of (discrete) linear programming models. Structures

of the models, definition of indexes and sets, definition of the data of the model.

- Declarations of variables and constraints
- Models for problems of resources allocations, transports, assembly, disassembly, blending.
- Problems of production planning and stock management. Multi-period problems
- Project Management techniques: PERT e CPM

Network Policies and Security

Course goals:

- Monitoring the parameters, analysing the traffic schema, managing and configuring a complex inter-networking system
- Optimizing the network performance and defining the services levels (SLA) founding on Quality of Services objectives.
- Constructing a security plan in an enterprise for protecting the informative systems knowing the mechanisms, services and attacks of security and applying up to date methodologies and provisions

Needed knowledge for reaching the goals:

- The format of the protocols of the TCP/IP suite: IP, TCP, UDP, ICMP, DNS, FTP, Telnet, SMTP, routing protocols RIP and OSPF. Unicast, multicast and broadcast traffic characterization
- Magnitude of network traffic, congestion phenomena, traffic measurement methodologies.
- Definition of QoS, tools and protocols for its realization and management
- Studying the functional areas of management: configuration, fault, performance, accounting, security. Management on IP: SNMP, RMON, MIB
- Security issues: weakness and typical attacks, terminology and risk analysis. Definition and development of an enterprise security policy
- Basic techniques for data and communication security; cryptography, authentication, managing cryptographic key: digital sign, CA, PKI, Directory Services, X509 systems, smart cards
- Security in the network: weaknesses of the TCP/IP protocols; Firewalls and proxies, NAT, VPN and IPSec
- Security of application services: email (PGP, S/MIME), FTP, Telnet, HTTP, HTTPs (SSL)
- Security of systems: types of attacks, malicious programs (virus and worms)

Network Programming

Course goals:

- Understanding the basic concepts of network programming

- Illustrating the client/server model
- Understanding the basic concepts of system programming

Needed knowledge for reaching the goals:

- Interfaces and protocols; socket interfaces, client/server model, addressing
- Connection: accept/connect mechanism, client programs examples, server programs examples
- Transmission: send/receive mechanism, buffering, examples
- Multi-process operating systems: file system, process management

Numerical Analysis

Course goals:

- Using main numerical methods for evaluating systems of linear equations, non-linear equations, problems connected with interpolation and numerical derivation.
- Interpreting numerical results relating them to problems of stability and error propagation.
- Being able to organize in an algorithm the various methods and to graphically verify their potential.

Needed knowledge for reaching the goals:

- Fundamental notions of error theory.
- Direct and iterative methods for the solution of systems of linear equations.
- Numerical methods for the solution of non-linear and transcendental equations.
- Global and local polynomial interpolation techniques.
- Mathematica software as a tool for verifying numerical methods studied during the course.

Operating Systems

Course goals:

- Showing the structure of an operating system and the policies adopted in the modern systems

Needed knowledge for reaching the goals:

- Structure, kernel, systems calls
- Memory and devices management
- Files management in centralized or distributed environment
- Hints of distributed operating systems, client/server

Operating Systems – Lab

Courses goals:

- Show the main concepts founding the management and administration of modern operating systems.

Needed knowledge for reaching the goals:

- Using at user level of specific operating systems (Unix and Windows 2000)
- Programming over Unix and Windows 2000

Operative Research

Course goals:

- Modelling simple linear programming problems (PL) and discrete linear programming (PLI)
- Evaluating the complexity of the models and of the tools required for their solution
- Interpreting the information obtained by resolving a PL or a PLI problem
- Using simple software for solving PL or PLI problems
- Using efficiently a simple specification algebraic language for PL or PLI problems

Needed knowledge for reaching the goals:

- Geometric and algebraic basic aspects of (Discrete) Linear Programming. Extreme point, vertex and base solution
- Graphical solving of 2-dimension problems
- Simplex algorithm: structure and properties. The two phases of the simplex method. Convergence of the algorithm
- Duality theory: constructing the dual problem, economic interpretation of duality. Theorems of duality. Complementarity
- Classical problems of PLI and main modelling techniques
- Valid inequalities. Solving methods for PLI problems
- Optimization problems on networks: definition, algebraic formulation and algorithms

Physics

Course goals:

- Determining hourly law and motion trajectory for systems which can be schematized as dot-like particles.
- Applying dynamics laws to the study of motion for systems with multiple particles.
- Analysing potential energy curves and characterizing, at least qualitatively, system

dynamics.

- Understanding basic principles of thermodynamics and the connections with mechanical energy.

Needed knowledge for reaching the goals:

- Kinematics, motion (one-dimensional, bi-dimensional, three-dimensional), kinematics quantities (scalar and vectorial), hourly equations and trajectories, Cartesian coordinates and polar coordinates. Reference systems, relative motion, Galilean invariance.
- The concept of force and Newton's laws.
- Kinetics energy and work, theorem of kinetics energy.
- Conservation of energy and potential energy.
- Definition of centre of mass and second law of Newton for a system with multiple particles.
- Quantity of motion and its conservation. Rudiments of rotational dynamics: rotational variables, moment of inertia, moment of a force. Angular moment and its conservation. Newton's law of gravitation, gravitational potential energy, Keplero's laws.
- Introduction to thermodynamics: temperature, heat quantity, specific heat, latent heat, mechanical equivalent of calorie, thermodynamic equilibrium, reversible and irreversible transformations.

Private Law

Course goals:

- Understanding principles and values expressed by the present Italian and European juridical system.
- Acquiring concepts and fundamental categories for private law science.
- Acquiring knowledge and interpretative skills concerning the most recurring subjects in the entrepreneurial context.

Needed knowledge for reaching the goals:

- Notion of juridical situation, subject, property, private autonomy.
- Rights on someone else's property, contract, tort liability.

Probability and Mathematical Statistics

Course goals:

- Having basic knowledge of statistical analysis and probability theory.
- Being able to summarize and graphically represent data, also using statistic software.
- Being able to recognize which kind of regression could better approximate two related variables, also using statistic software.

- Being able to model and solve simple problems using probability theory and without using measure theory.
- Being able to apply the concepts of probability theory to statistical survey.
- Being able to apply the main statistical methodologies, also using statistic software.

Needed knowledge for reaching the goals:

- Descriptive statistics and computer applications.
- Linear and non-linear regression, correlation among variables. Usage of statistic software.
- Rudiments of probability calculus. Classical probability. Conditioned probability. Random variables. Main discrete and continuous probability laws.
- Limit theorems. Approximation of finite models using Gauss distribution.
- Introduction to inferential statistics: population and random sample, fiduciary intervals, hypothesis tests.

Programming

Course goals:

- Understanding and defining the syntactic structure of programming languages
- Understanding and defining, informally and formally, the semantics of fundamental constructs of imperative programming languages
- Understanding and defining, informally and formally, the semantics of fundamental constructs of object oriented programming languages
- Writing basic algorithms correctly using a programming language: search on sequences of data, filtering of sequences of data, checking properties of sets of data

Needed knowledge for reaching the goals:

- Algorithms and problem solving
- Recursion and inductive definitions
- Formal tools for defining the syntax of programming languages: finite states automata and context free grammars
- Formal tools for defining the operational semantics of programming languages: transition systems
- Variables, types, modelling the state
- Basic constructs of object oriented programming languages: classes, objects, methods call
- Definition and use of sequences of data (array and list) or sets of data. Programming schemes on these data structures: linear searches, reading, filtering, computing aggregates

Programming – Lab

Course goals:

- Deeply understanding an high-level language programming.
- Being able to use language for simple algorithm programming.
- Being able to design a medium complexity application using the paradigm of object programming.
- Being aware of the various phases of the development of a program (compilation, loading, running, debugging, etc.).

Needed knowledge for reaching the goals:

- Programming languages and their implementation.
- Java language: structure, foundations.
- Object programming in Java: classes, objects, methods, inheritance.
- Application Programming Interface (API): Java library, API and corresponding classes development, using javadoc tool.
- Separation between application front-end and application back-end: guidelines for designing and implementation.
- Imperative core of Java programming: declarations, data types, iterative statements, conditional statements.
- Interfaces and polymorphism, strategic interfaces, generic code writing.
- Inheritance and code reuse.
- Classes and interfaces for managing object collections in Java.
- Exception handling in Java.
- Development of a software project in Java.

Programming Languages and Compilers

Courses goals:

- Understanding of the foundations of programming languages, of the definition of the syntax and semantics, of the main translation techniques

Needed knowledge for reaching the goals:

- Introduction: programming languages, paradigms, history
- Translation process: compilation and interpretation
- General structure of a compiler: phases
- Lexical analysis: tokens recognizing, regular expressions, finite states automata, algorithms
- Semantic analysis: context-free grammars, top-down analysis, bottom-up analysis,

construction of LR(1), LL(1), LALR(1), SLR(1) tables

- Static analyses using attributes and grammar productions. Es: type checking

Software Engineering

Course goals:

- Showing main software development methodologies, system models, software architectures, verification and validation techniques, as well as development support tools.

Needed knowledge for reaching the goals:

- Requirement analysis and specification.
- Design and development.
- Verification and validation.
- Support processes (management and control, documentation, etc.)

Software Engineering – Lab

Course goals:

- Trying the different phases of software systems design, with reference to a particular modelling formalism.

Needed knowledge for reaching the goals:

- Main UML diagrams.
- Design of a software system using UML.

Systems of Automation and Laboratory

Course goals:

- Introducing the problems studied by Automation
- Acquiring basic methods and techniques of modelling, analysis and control of systems
- Acquiring the basic skills for the representation and the analysis of systems, for the definition of the problem of their control and for the solution of this problem
- Acquiring the needed operative skills for implementing the studied methods and techniques

Needed knowledge for reaching the goals:

- Introduction: general structure of a dynamic system
- General properties of input-output maps, their representation in the time domain and in the domain of Laplace transform
- Bode diagrams and harmonic response
- Graphical representation of complex variables functions: polar diagrams

- Input-output stability of linear and stationary systems
- Analysis of LTI systems and hints on synthesis of continue time control systems

Web Technologies

Course goals:

- Giving a guide to means, methodologies and tools that are available for designing and realizing interactive and dynamic web sites

Needed knowledge for reaching the goals:

- Server side: IP network, HTTP and HTTPS protocols, active pages, unix world, microsoft world, security
- Client side: browsers, HTML, Styles/CSS, Javascript, Applets, Active X