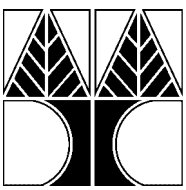
UNIVERSITY OF CYPRUS

ECTS GUIDE

DEPARTMENT OF COMPUTER SCIENCE

2003-2004



CONTENTS

SOCRATES PROGRAMME

MAIN OBJECTIVES

The Republic of Cyprus participates in the SOCRATES Programme through a 1998 bilateral agreement with the European Union. Under the terms of this agreement the Government of Cyprus pays an annual subscription of approximately 500,000 EURO to ensure the right of participation, up to this amount, for all eligible educational institutions of Cyprus.

The University of Cyprus, as one of the eligible institutions nominated by the SOCRATES National Agency which, in the case of Cyprus, is the Ministry of Education and Culture, submitted its first Institutional Contract in November 1997.

The objective of the University is to fully participate in European programmes, with the broad aim of extending its international cooperation, and with specific emphasis on the promotion of European ideals in education. The goals set by SOCRATES - those of promoting a European dimension, transparency, mutual trust and recognition of programmes - are ideals which the University of Cyprus fully supports. Through its existing international contacts, the University has already made a first but significant step in contributing to the realisation of these common goals. SOCRATES provides us with a welcome opportunity to make our contribution towards this pan-european effort, a challenge that we take up with eagerness as well as full awareness of the responsibilities that participation involves.

ECTS CREDITS

The European Union's goal of promoting the European dimension in education is facilitated to a large extent through the mobility of students. The ERASMUS Action contributes to this effort by enabling students to learn about, and experience European countries other than their own, their languages, ideals and cultures. Moreover, student exchanges are increasingly becoming a major factor in the development of academic and professional careers.

The future aim of the Union is the recognition, on a pan-european level, of study programmes and qualifications, thus making a reality the vision of an open European educational and vocational training area. The European Credit Transfer System (ECTS) was created and is being promoted to contribute to this goal. It is a tool for establishing and securing transparency, as well as a means of building communication and cooperation among institutions, while simultaneously broadening the educational choice of students.

The system is based on three criteria: **Information** (concerning the programme of study and student performance), **Mutual Learning Agreements** (which are signed by the participating institutions and students), and the use of the **ECTS** (which guarantees transparency with respect to student workload, and which is calculated based on lecture hours, laboratories, workshops, assignments and self-study components of the programme). The system is implemented, and its transparency secured, through the following: the information package, the student application/learning agreement and the transcript of records. The use of ECTS is voluntary and is based on mutual trust and recognition of the academic performance of the collaborating institutions.

The main facilitators of ECTS are the students and academic faculty who believe in its goals and who wish to extend student studies abroad. Full academic recognition is a prerequisite for the mobility of students. This means that the study-abroad period (which includes examinations and other forms of assessment) substitutes for an equal period of home study (which also includes examinations and other forms of assessment), irrespective of the fact that the content of study may be different. In other words, ECTS is based on the full-time workload of the students and is not restricted to the hours of attendance in class.

There are 60 credit units in ECTS, which represent the workload for a full academic year; 30 credits for a semester and 20 credits for a trimester/term. At the University of Cyprus the correspondence between home credits and ECTS is exactly 2:1 (2 European credits: 1 University of Cyprus credit).

The University of Cyprus has implemented the credit system in all departments since its inception, and has established this system on criteria almost identical to that of ECTS.

ELIGIBILITY FOR PARTICIPATION

Students who wish to participate in ECTS must secure in advance the mutual recognition of their credits. Thus, submission of the **Learning Agreement** signed by the host, sending departments and the student is required. Participation in the exchange programme is conditional as follows:

- Students must be citizens of any EU country, the EFTA countries, associated countries in Eastern Europe (Czech Republic, Hungary, Poland, Romania and Slovak Republic), or Cyprus.
- Study at the host University will be free; however, any fees payable to the sending University will still be required.
- Any grant/loan given to students will not be affected by their participation in the ERASMUS exchanges. The period of study abroad should be between 3-12 months.
- First-year students are not entitled to participate in any exchanges.

DEADLINES

Fall Semester 2003-2004

Submission of Application: Beginning of July

Arrival of Students/Registration: End of August / Beginning of September

Spring Semester 2003-2004

Submission of Application: Beginning of November

Arrival of Students/Registration: Third Week of January

SCHOOL OF GREEK LANGUAGE

SOCRATES/ERASMUS students can attend free of charge the intensive Greek language programme that is offered before the beginning of the Fall Semester (August). The intensive programme is of four (4) weeks duration and is intended primarily for exchange students who will be attending courses (of graduate or postgraduate level) at the University of Cyprus.

The intensive programme is offered at both beginner and intermediate levels. Placement of students is based on written examinations.

The courses offered are the following:

(a) Greek language course, which includes familiarisation with the Cypriot dialect, with which the students will have daily contact.

(b) Greek civilisation course, which refers to cultural and social issues facing contemporary Greek society.

(c) Practical exercises in Greek (eg., phonetics and phonology).

(d) Courses of special interest.

Upon course completion, which is followed by both written and oral examinations, a certificate of the School of Greek Language is issued, recording the programme attended, the level, the duration of attendance, the teaching hours and the distinction achieved.

The School of Greek Language is housed at the renovated Axiothea Street Building situated within the city walls.

FURTHER INFORMATION

More details concerning the University of Cyprus, Cyprus itself, the administrative services supporting the SOCRATES Programme and other useful information can be found in the *SOCRATES/ERASMUS Information Guide*, which is published separately. The Guide can be obtained from the Service for Research, International and Public Relations of the University of Cyprus (tel.: +357 22 375115, ext. 16, 18, fax: +357 22 375866, e-mail: adavgous@ucy.ac.cy).

DEPARTMENT OF COMPUTER SCIENCE

Computer Science is considered to be the unquestionable science of the future. Although it is a newly born science, it has already been established as an important science and our society. In conjuction with the development of the worldwide information networks (e.g. Internet) that help to materialize this Information Society, new developments in Computer Science include:

- Wareable Computing, e.g., portable computers for health and information purposes
- Telematics, for education, trade and distance services
- Data Mining, for the extraction of knowledge from large data collected daily from organizations and businesses
- understanding and automation of phenomena of human intelligence, like that of natural language understanding and learning, for more advanced help from computers to humans in their mental activities.

The Department offers programs of studies that lead to the following titles: B.Sc. in Computer Science (since 1992), Ph.D. in different specializations of Computer Science (since 1997) and M.Sc. in Advanced Information Technologies (since 2000). The number of new undergraduate students has increased from 30 to 45 since September 2000, while the annual inflow of postgraduate students is now about 40.

Teaching is carried out in the internationally established way through lectures, tutorials, laboratories and seminars. Students are expected to consistently participate in all activities of classes they are attending. The Department reserves the right to disallow students who systematically skip the activities of attended classes from taking the corresponding final examinations.

In most classes, homework is assigned regularly to help students master the course material and develop practical skills. Homework is carried out either individually or collaboratively in small student groups. Evaluation is usually based on assigned homework, written and oral examinations, etc. Special efforts are made to employ continuous evaluation as much as possible. Students are informed about the specific procedure of teaching and evaluation for each class from the class instructor. The relevant information is included in the "General Information" document for the class, which is handed out to the students during the first week of classes each semester. The instructor is available to the students during predetermined weekly office hours.

The human resource of the Department of Computer Science consists of 17 academic members: 2 Professors, 7 Associate Professors, 3 Assistant Professors and 5 Lecturers. Furthermore, the Department employs around 50 Special Scientists and Postgraduate Associateç, 2 IT Officers και 2 Assistants of IT Officers.

This year is especially important for our Department, since it moved this year to the new campus of the University where, for the first time, the Department has its own fully equipped laboratories. The Department will therefore be able to develop further its teaching by offering more advanced specialized courses at both the undergraduate and postgraduate level. The Department's move to the new facilities coincides with the rapid development of its research projects and teams, funded mainly by the European Union. This fortunate coincidence will help significantly in consolidating and further developing the research activities of the Department.

The space currently employed by the Department at the new Campus includes the following four Teaching Labs, which are fully equipped with the latest generation of hardware/software:

1. Laboratory of Undergraduate Studies
2. Laboratory of Postgraduate Studies
3. Digital Systems Design and Microprocessor Laboratory
4. UNIX Laboratory.

Furthermore, the Department has a Walk-in Laboratory, in which its personnel and students may bring their portable computers and connect to the Internet and the departmental computer systems.

The Department continues since its foundation 11 years ago, to be very active in research and to contribute significantly in this international arena. Of utmost importance is its recent participation in research projects of the 5th Framework for Research and Technological Development of the European Union, with the largest of its four thematic units devoted to the Information Society. Our Department secured recently twelve such participations, ranking first, with respect to the number of approved projects, among all universities and research centers of the European Union that submitted proposals for research funding within Global Computing. In addition, the Department collaborates, in the context of its research and developmental activities, with various public and private organizations in Cyprus.

The research programs of the Department are supported by relevant Research Laboratories. Within the context of its strategic development, the Department has founded until now and set into operation the following eleven Research Laboratories:

1. Computer Architecture Laboratory
2. E-Commerce Laboratory
3. Graphics and Hypermedia Laboratory
4. High-Performance Computing Laboratory
5. Laboratory for Artificial Intelligence and Intelligent Systems
6. Laboratory for Foundations of Computing Systems and Theoretical Computer Science
7. Medical Informatics Laboratory
8. Mobile and Wireless Computing Laboratory
9. Multimedia Research and Development Laboratory
10. Networks Laboratory
11. Software Engineering & Internet Technologies Laboratory

Each Research Laboratory can additionally support a number of Diploma Projects and Postgraduate Projects (at the M.Sc. level).

UNDERGRADUATE DEGREE PROGRAMME

The undergraduate curriculum leads to a Batchelor's degree in Computer Science. The Department's course material is conceptually divided into the following areas or components: Theory, Computer Systems, Problem Solving and Applications.

(a) The Theory component is concerned with the foundations of Computer Science: theory and models of computation, and the design and analysis of algorithms. Generally this component aims at cultivating a formal approach to thinking, and organizing and processing information. Logic, and its role as the calculus of Computer Science, is an important topic. Essential concepts of Discrete Mathematics are taught as an integral part of related courses offered by the Department. Students are also required to follow a number of courses offered by the Department of Mathematics and Statistics in order to develop their abilities for abstraction and formal thinking, and to acquire other useful mathematical skills.

(b) The Computer Systems component is concerned with hardware and software systems and develops the concept of virtual or abstract machine. It includes basic principles of computer architecture and organization, operating systems, programming language design and implementation, microprocessor systems, data communications, networks, distributed systems, and parallel and novel architectures.

(c) The Problem Solving component aims at developing algorithmic thinking, with emphasis on principles of programming and program design. Through this component, students will acquire competence in a number of programming languages using a variety of programming paradigms (imperative, object-oriented, logic-based). Concurrent, parallel, distributed and heuristic techniques of problem solving are addressed in restricted choices. In this component, students learn how to design, implement and evaluate solutions to significant, albeit relatively small, problems. (The wider integration of these techniques in building methodological frameworks to solve real-world problems is studied in courses on systems analysis and design, and software engineering in the "applications" component.)

(d) The Applications component aims at bringing together the knowledge and skills acquired in the other three components for the development of useful applications to solve "real-world problems". Important technologies such as databases, knowledge bases, graphics and user interface managers are introduced as examples of applications in themselves and as vital tools for the construction of higher level applications. Modern Software Engineering methodologies that address every stage in the planning, design, development, and maintenance of high-quality applications are studied, and subsequently put into practice in the context of a group project undertaken by all students during their sixth semester. Finally, important social and ethical issues concerned with the spread of computers are raised and discussed.

Each Computer Science course is equivalent to 8 ECTS units, apart from CS102-Explorations into Computer Science, which is equivalent to 2 ECTS units. The lessons offered to students of other departments (restricted or unrestricted choices) may carry 6 or 8 ECTS units.

The courses of the Program of Undergraduate Studies are divided into 3 categories:

(1) Compulsory courses, that constitute the core material

(2) Restricted Choices, which are offered by the Department and allow the student to specialize in a specific direction of Computer Science or to gain knowledge that covers a wide spectrum of areas of Computer Science, and

(3) Unrestricted Choices, which are offered by other Departments.

Some courses have other courses as prerequisites. The course dependencies between the compulsory courses are depicted in Table 1. Almost all courses offered during the first six semesters are compulsory, whereas the last two semesters comprise mostly restricted choices, in several topics in Computer Science.

Some courses have other courses as prerequisites. The course dependencies between the compulsory courses are depicted in Table 1. Almost all courses offered during the first six semesters are compulsory, whereas the last two semesters comprise mostly restricted choices, in several topics in Computer Science.

In addition, students must fulfill the University's foreign language requirement by attending three courses in a foreign language. The Department has allocated 18 ECTS units to these courses, and specifies English as the foreign language.

ANALYTICAL PROGRAM OF UNDERGRADUATE STUDIES

1st Semester (36 units)		ECTS Units
Code	**Course Title**	
CS 111	Discrete Structures in Computer Science and Computation	8
CS 131	Programming Principles I	8
MAS 004	Introductory Mathematics I	8
PBA 131	Principles of Management I	6
FLL 100	General Advanced English	6
2nd Semester (32 units)		
CS 102	Explorations into Computer Science	2
CS 121	Digital Systems	8
CS 132	Programming Principles II	8
MAS 121	Linear Algebra I	8
FLL 101	Academic English	6
3rd Semester (36 units)		
CS 221	Computer Organization and Assembly Language Programming	8
CS 231	Data Structures and Algorithms	8
MAS 055	Introduction to Probability and Statistics	8
FLL 200	Topics in Academic English	6
	Unrestricted Choice	6
4th Semester (32 units)		
CS 211	Theory of Computation and Complexity	8
CS 222	Operating Systems	8
CS 233	Object-Oriented Programming	8
CS 241	Systems Analysis and Design	8
5th Semester (32 units)		
CS 224	Communications and Networks	8
CS 242	Databases	8
CS 244	Software Engineering	8
	Restricted Choice	8
6th Semester (32 units)		
CS 223	Theory and Practice of Compilers	8
CS 440	Software Engineering Group Project	8
	Restricted Choice	8
	Restricted Choice	8
7th Semester (30 units)		
CS 400	Diploma Project I	8
	Restricted Choice	8
	Restricted Choice	8
	Unrestricted Choice	6
8th Semester (30 units)		
CS 401	Diploma Project II	8
	Restricted Choice	8
	Restricted Choice	8
	Unrestricted Choice	6
	Total	**260**

Restricted Choices

Some of the Restricted Choices may be linked to the research interests of members of the Academic Staff. There is, however, provision so that adequate cover is offered in important areas such as Artificial Intelligence, Computational Complexity, Databases, Distributed Algorithms and Systems, Distributed Databases, Distributed Transaction Processing, Information Management Systems, Modern and Parallel Architectures, Networks and Communications, Multimedia, Neural Networks, Parallel Algorithms, Programming Models, Semantics of Programming Languages, Signal and Image Processing, and Software Engineering Methodologies.

Software Engineering Group Project

During the sixth semester of her/his studies, every student is required to participate in a Software Engineering Group Project, where the knowledge, tools and techniques which they have been taught in the Software Engineering and other related courses, are put into practice. Besides developing technical skills, students obtain a first experience of group work, and have the opportunity to develop important communication and co-ordination skills.

Diploma Project

During the last two semesters of her/his studies, each student undertakes an individual Diploma Project in accordance with regulations, which are included in the Undergraduate Studies Prospectus of the Department.

TABLE 1: Dependencies between courses

Code	Course	Prerequisites / Conditions
CS 102	Explorations into Computer Science	_____
CS 111	Discrete Structures in Computer Science and Computation	_____
CS 121	Digital Systems	_____
CS 131	Programming Principles I	_____
CS 132	Programming Principles II	CS 131 Programming Principles I
CS 211	Theory of Computation and Complexity	CS 111 Discrete Structures in Computer Science and Computation MAS 004 Introductory Mathematics I
CS 221	Computer Organization and Assembly Language Programming	CS 121 Digital Systems CS 131 Programming Principles I
CS 222	Operating Systems	CS 221 Computer Organization and Assembly Language Programming
CS 223	Theory and Practice of Compilers	CS 132 Programming Principles II CS 211 Theory of Computation and Complexity CS 231 Data Structures and Algorithms
CS 224	Communications and Networks	CS 121 Digital Systems CS 131 Programming Principles I
CS 231	Data Structures and Algorithms	CS 111 Discrete Structures in Computer Science and Computation CS 132 Programming Principles II
CS 232	Algorithms and Complexity	CS 231 Data Structures and Algorithms

CS 233	Object-Oriented Programming	CS 132 Programming Principles II CS 231 Data Structures and Algorithms
CS 241	Systems Analysis and Design	CS 231 Data Structures and Algorithms CS 241 Systems Analysis and Design
CS 242	Databases	CS 221 Computer Organization and Assembly Language Programming Approval by Academic Advisor
CS 244	Software Engineering	CS 224 Communications and Networks
CS 321	Computer Architecture	CS 221 Computer Organization and Assembly Language Programming CS 222 Operating Systems
CS 400	Diploma Project I	CS 221 Computer Organization and Assembly Language Programming CS 111 Discrete Structures in Computer Science and Computation MAS 004 Introductory Mathematics I MAS 121 Linear Algebra I
CS 420	High Speed Networks with Multiservice Connections and Multimedia	CS 224 Communications and Networks CS 233 Object-Oriented Programming
CS 421	Parallel Processing: Architectures and Languages	CS 132 Programming Principles II CS 231 Data Structures and Algorithms CS 211 Theory of Computation and Complexity
CS 422	Multimedia, Hypermedia and Cyberspace	CS 231 Data Structures and Algorithms
CS 423	Microprocessor Systems	CS 231 Data Structures and Algorithms CS 434 Logic Programming and Artificial Intelligence
CS 424	Digital Signal Processing	CS 111 Discrete Structures in Computer Science and Computation CS 132 Programming Principles II
CS 425	Internet Technologies	CS 244 Software Engineering CS 132 Programming Principles II CS 244 Software Engineering CS 231 Data Structures and Algorithms
CS 426	Computer Graphics	CS 442 Learning Systems CS 231 Data Structures and Algorithms MAS 121 Linear Algebra I
CS 431	Synthesis of Parallel Algorithms	CS 242 Databases CS 231 Data Structures and Algorithms
CS 432	Distributed Algorithms	CS 233 Object-Oriented Programming CS 233 Object-Oriented Programming
CS 433	Constraint Programming and Satisfaction	
CS 434	Logic Programming and Artificial Intelligence	
CS 435	Human-Computer Interaction	
CS 440	Software Engineering Group Project	
CS 441	Parallel and Distributed Software Engineering	
CS 442	Learning Systems	
CS 443	Artificial Intelligence and Expert Systems	
CS 444	Computational Intelligence Systems	
CS 445	Digital Image Processing	
CS 446	Advanced Databases	
CS 447	Artificial Intelligence	
CS 448	Intelligent Systems and Multi-Agent Systems	
CS 449	E-Business/E-Commerce	

DESCRIPTION OF UNDERGRADUATE COURSES

Compulsory Courses

CS 102 - Explorations into Computer Science

A. Kakas

Course Units: 1 / ECTS Credits:2

Weekly lectures/seminars that cover a broad spectrum of Computer Science and introduces first-year students to subjects that compose a spherical picture of Computer Science and its basic areas, starting from its birth and reaching its modern evolutions.

Bibliography

No bibliography is required for this course.

CS 111 - Discrete Structures in Computer Science and Computation

Y. Dimopoulos

Course Units: 4 / ECTS Credits: 8

Foundations: sets and functions. Mathematical reasoning: methods of proof, induction. Counting: basics of counting, pigeonhole principle, permutations and combinations. Program specification and correctness. Relations: properties and applications, equivalence relations, partial orders. Logic: Propositional Logic, basics of Predicate Logic. Graphs: basic concepts.

Bibliography

K. H. Ronen, *"Dicsrete Mathematics and its Applications"*, Mc Graw Hill, 1999.

CS 121 - Digital Systems

C. Pattichis

Course Units: 4 / ECTS Credits: 8

Principles of design and construction of digital electronic systems and computers. Representation of data with binary sequences. Data storage and processing by electronic digital circuits. Consolidation of theoretical knowledge through practical exercises in the design and construction of digital circuits in the Laboratory for Digital Systems Design and Microprocessors.

Bibliography

M. M. Mano and C. R. Kime, *"Logic and Computer Design Fundamentals and Xilinx 4.2", 2nd edition*, Prentice Hall, 2002.

CS 131 - Programming Principles I

Y. Sazeides

Course Units: 4 / ECTS Credits: 8

Presentation of the software development process and introduction to the basic principles of programming and program design using the C language. Global overview of the C language with emphasis on data types, control structures, data structures, functions and modular programming.

Bibliography

J. R. Hanly and E. B. Koffman, *"Problem Solving & Program Design in C"* Addison-Wesley, 3rd edition, 1999.

B. W. Kernigham and D. Ritchie, *"The C programming Language-ANSI edition"*, Prentice Hall.

CS 132 - Programming Principles II

Y. Sazeides/P. Trancoso

Course Units: 4 / ECTS Credits: 8

Advanced Programming techniques and methodologies based on the C language. Topics include static and dynamic memory manipulations, file management, pointers, recursion, dynamic data structures and the C preprocessor. Introduction to the object-oriented programming model through the C++ language with emphasis on objects, abstraction and encapsulation, methods, classes and class inheritance.

Bibliography

J. R. Hanly and E. B. Koffman, *"Problem Solving & Program Design in C"*, Addison-Wesley, 3rd edition, 1999.

B. W. Kernigham and D. Ritchie, *"The C programming Language-ANSI edition"*, Prentice Hall.

M. Smith, *"Object Oriented Software in ANSI C++"*, McGraw Hill, 2nd edition, 1999.

CS 211 - Theory of Computation and Complexity

M. Mavronicolas

Course Units: 4 / ECTS Credits: 8

Formal methods of computation based on machines, grammars and languages: finite automata vs. regular languages; pushdown automata vs. context-free grammars; Turing machines vs. unrestricted grammars. Models of computation equivalent to Turing machines and Church_s Thesis. Computability and Uncomputability. Introduction to Theory of Computational Complexity with emphasis on the Theory of NP-completeness.

Bibliography

H.R. Lewis and C.H. Papadimitriou, *"Elements of the Theory of Computation"*, 2nd edition, Prentice-Hall, 1998.

M. Sipser, *"Introduction to the Theory of Computation"*, PWS Publishing Company, 1997.

J.E. Hopcroft, R. Motwani and J.D. Ullman, *"Introduction to Automata Theory, Languages and Computation", 2nd edition*, Addison-Wesley, 2001.

CS 221 - Computer Organization and Symbolic Programming

P. Evripidou

Course Units: 4 / ECTS Credits: 8

Introduction to computer organization and technology. Instruction types and coding. Arithmetic and logic unit. Principles of organization of the basic units of a computer: central processing unit (CPU), memory, input, output. Interface of CPU and peripheral devices. Assembly language programming for the MIPS R2000/R3000 and Intel Pentium, and practical experience in the Digital Systems and Microprocessor Laboratory.

Bibliography

D. A. Patterson and J. L. Hennessy, *"Computer Organization & Design the Hardware/Software Interface", 2nd edition*, Morgan-Kaufmann, 1998.

CS 222 - Operating Systems

G. Papadopoulos

Course Units: 4 / ECTS Credits: 8

Basic principles of design and operation of modern operating systems. Study of various operation levels and mechanisms. Case studies involving typical operating systems such as Unix, VMS, Windows and Macintosh. The dual role of an operating system, as manager of the various parts of the computer hardware and supplier of offered services to the user.

Bibliography

W. Stallings, *"Operating Systems: Internals and Design Principles"*, Prentice Hall, 4th edition, 2001.

CS 223 - Theory and Practice of Compilers

P. Trancoso

Course Units: 4 / ECTS Credits: 8

Fundamental principles of compiler design. Relation of translators to formal languages and automata theory. Lexical, syntactic and semantic analysis, code generation and optimization, etc. Practical exercises using lex and yacc.

Bibliography

A.V. Aho, R. Sethi and J.D. Ullman, *"Compilers Principles, Techniques, and Tools"* Addison-Wesley, 1986.

T. Mason and D. Brown, *"lex & yacc"*, O_Reilly & Associates Inc., 1990.

A. W. Appel, *"Modern Compiler Implementation in C"*, Cambridge University Press, 1998.

R. Hunter, *"The Essence of Compilers"*, Prentice Hall, 1999.

CS 224 - Communications and Networks

A. Pitsillides

Course Units: 4 / ECTS Credits: 8

Data and computer communication fundamentals. Protocols, local and wide area networks, open systems, such as OSI and the Internet, interconnection of networks. Introduction to network and protocol performance evaluation.

Bibliography

F. Halsall, *"Multimedia Communication"*, 2001 (webpage: http://www.booksites.net/halsall).

Lecture notes, research studies and articles (webgpage: http://www.cs.ucy.ac.cy/courses/ EPL224/lectures.htm).

CS 231 - Data Structures and Algorithms

A. Philippou

Course Units: 4 / ECTS Credits: 8

Study of data structures for the organization and efficient processing of data. Linear and non-linear data structures, bit-vectors. Hashing techniques. Issues of memory management. Introduction to algorithm design techniques. Sorting Algorithms. Graph Algorithms. Analysis of the average and worst-case complexity of algorithms.

Bibliography

M. A. Weiss, *"Data Structures and Algorithms Inc"*, Addison-Wesley, 1996.

Cormen, Leiserson and Rivest, *"Introduction to Algorithms"*, McGraw Hill, 2001.

CS 233 - Object-Oriented Programming

M. Dikaiakos

Course Units: 3 / ECTS Credits:6

Principles of object-oriented programming, and implementation using the language JAVA. Problem-solving and programming using object-oriented methodologies.

Bibliography

B. Eckel, *"Thinking in Java," 2nd edition*, Prentice Hall, 2000.

B. Kernighan and R. Pike, *"The Practice of Programming"* Addisson Wesley, 1999.

CS 241 - Systems Analysis and Design

Visiting Professor

Course Units: 4 / ECTS Credits: 8

Study of the theory and the methodologies which have been developed over the years in the area of systems, with the objective to introduce techniques and methodologies for systems analysis and design of Information Systems. Special attention to the study of the "Information Society" and its effect in system development and maintenance.

Bibliography

J. Whitten, L. Bentley and K. Dittman, *"Systems Analysis & Design Methods"*, 5th edition, McGraw-Hill, 2001.

Lecture transparencies (webpage: http://www.mhhe.com/business/mis/whitten/).

Lecture notes (webpage: http://www.cs.ucy.ac.cy/courses/EPL241/).

CS 242 - Databases

G. Samaras

Course Units: 4 / ECTS Credits: 8

Introduction to Databases. Organization and proper management of large quantities of data for use in applications. Database models such as the entity-relation model, the relational model, the network model and the hierarchical model.

Bibliography

Elmasri and Navathe, *"Fundamentals of Database Systems"*, Benjamin Cummings, 1989.

Korth and Silberschatz, *"Database System Concepts"*, McGraw-Hill, 1986.

CS 244 - Software Engineering

A. Andreou

Course Units: 4 / ECTS Credits: 8

Methods, tools, and procedures for the development and maintenance of large-scale software systems within specified quality and cost constraints. Life-cycle models, specification techniques, software development methodologies, verification and validation, CASE, and other tools. Project planning and management. Practical experience with CASE tools for modeling data and procedures.

Bibliography

I. Sommerville, *"Software Engineering", 6th edition*, Addison-Wesley, 2001.

S. Scach, *"Object Oriented and Classical Software Engineering"*, McGraw-Hill, 2002.

G. Booch, J. Rumbaugh and J. Jacobson, *"The UML user guide"*, Addison-Wesley, 2001.

CS 440 - Software Engineering Group Project

A. Andreou

Course Units: 4 / ECTS Credits: 8

Undertake and carrying out to completion of a significant software project by small student groups (of about 4-5 students each). All phases in the development of software. Some of the specific projects come from the industrial sector.

Bibliography

I. Sommerville, *"Software Engineering", 6th edition*, Addison-Wesley, 2001.

S. Scach, *"Object Oriented and Classical Software Engineering"*, McGraw-Hill, 2002.

G. Booch, J. Rumbaugh and J. Jacobson, *"The UML user guide"*, Addison-Wesley, 2001.

Books and manuals for several programming languages and tools.

Restricted Choices

CS 232 - Algorithms and Complexity

A. Philippou

Course Units: 4 / ECTS Credits: 8

Topics in the design and analysis of efficient algorithms and their complexity. Significant algorithms in Graph Theory, Algebra, Geometry, Number Theory and Combinatorics. General algorithmic techniques (e.g., divide-and-conquer, backtracking, dynamic programming). Randomized algorithms. Advanced topics: Fast Fourier Transform, sorting networks, Parallel Algorithms, Inherent (lower) bounds on problem complexity.

Bibliography

Cormen, Leiserson and Rivest, *"Introduction to Algorithms"*, McGraw Hill, 2001.

Brassard and Bradley, *"Fundamentals of Algorithmics"*, Addison-Wesley, 1995.

Sedgewick and Flajolet, *"An Introduction to the Analysis of Algorithms"*, Addison-Wesley, 1995.

CS 321 - Computer Architecture

Y. Sazeides

Course Units: 4 / ECTS Credits: 8

Introduction to the state-of-the-art of uniprocessor, high performance computer architecture. Emphasis on quantitative analysis and cost/performance trade-offs in the design of the basic units of a RISC processor: instruction set, pipelining, memory system, and input/output systems. Qualitative analysis of real machines and their performance data complementing the quantitative analysis.

Bibliography

J. L. Hennessy and D. A. Patterson, *"Computer Architecture: A Quantitative Approach"*, Morgan-Kaufmann, 3rd edition, 2003.

CS 411 - Semantics of Programming Languages

A. Kakas

Course Units: 4 / ECTS Credits: 8

Basic types of semantics of programming languages: procedural, declarative and axiomatic. Relations among these basic types. Formal concepts of semantics and their applications to common programming languages. Introduction to Field Theory and Information Systems within the framework of semantics of programming languages.

Bibliography

G. Winskel, *"The Formal Semantics of Programming Languages: An Introduction"*, MIT Press, 1993.

C. Gunter, *"Semantics of Programming Languages: structures and techniques"*, MIT Press, 1993.

CS 420 - High Speed Networks with Multiservice Connections and Multimedia

A. Pitsillides

Course Units: 4 / ECTS Credits: 8

Design, and management and control of communication networks. Special emphasis on integrated service networks (fixed and mobile) (e.g., 100Mbit/sec Ethernet, Gigabit Ethernet, ATM) new Internet architectures (e.g. Diff, IPUG, Internet 2).

Bibliography

J. F. Kurose and K. W. Ross, *"Computer Networking - A Top-Down Approach Featuring the Internet", 2nd edition*, Addison-Wesley, 2003.

D. E. Comer, *"Computer Networks and Internets"*, Prentice Hall, 1997.

W. R. Stevens, *"TCP/IP Illustrated, Volume I"*, Addison-Wesley, 1994.

CS 421 - Parallel Processing: Architectures and Languages

P. Evripidou

Course Units: 4 / ECTS Credits: 8

The entire spectrum of parallel machines as appearing in Flynn's classification: SISD, SIMD, MISD, MIMD. The main approaches for design and operation of multiprocessor systems. Conventional and non-conventional machines (Data-flow and reduction). Parallel programming approaches: (1) Automatic-parallelizing compilers, (2) Extending serial languages with parallelizing constructs, (3) parallel languages for Functional Programming. Special emphasis on parallel architectures and parallel programming.

Bibliography

D. E. Culler, J. Pal Singh and A. Gupta, *"Parallel Computer Architecture: A Hardware/Software Approach"*, Morgan-Kaufmann.

CS 422 - Multimedia, Hypermedia and Cyberspace

S. Retalis

Course Units: 4 / ECTS Credits: 8

Introduction to multimedia, hypermedia and the development of applications for tele-cooperation systems via computers. User interfaces, methods of communication and cooperation among the users, design of shared workspace, combining information. Emphasis on applications built on top of Internet and the World Wide Web.

Bibliography

R. Steinmetz and K. Nahrstedt, *"Multimedia: Computing, Communications & Applications"*, Innovative Technology Series, Prentice Hall, 1995.

T. Vaughan, *"Multimedia: Making It Work"*, 4th edition, Osborne McGraw-Hill, 1998.

CS 423 - Microprocessor Systems

C. Pattichis

Course Units: 4 / ECTS Credits: 8

Design, implementation, programming and control of microprocessor systems. Emphasis is given in laboratory work, and the overall target is to design and implement plug-in PC cards for the development of stand-alone microprocessor systems.

Bibliography

B. Brey, *"The Intel Microprocessors, Architecture, Programming, and Interfacing, 6th edition*, Prentice Hall, 2003.

CS 424 - Digital Signal Processing

A. Pitsillides

Course Units: 4 / ECTS Credits: 8

Introduction to the basic principles, techniques, and applications of Digital Signal Processing (DSP). Analysis of discrete-time signals and systems in the time and frequency domains. Z and Fourier transforms. Discrete Fourier Transform (DFT), Fast Fourier Transform (FFT) and their applications. Digital filters. Examples of applications.

Bibliography

J. G. Proakis and D. G. Manolakis, *"Digital Signal Processing: Principles, Algorithms and Applications", 3rd edition* or later, Brooks Cole, 1999.

CS 425 - Internet Technologies

M. Dikaiakos

Course Units: 4 / ECTS Credits: 8

This course addresses topics of Internet and World-Wide Web technologies, with an emphasis on WWW applications and Internet programming. In the first part of the course we examine the foundations of WWW applications including hypertext, navigation in hyperspace, hypertext usability, information overload, markup languages and methodologies of WWW application design. In the second part of the course we examine systems issues related to Internet programming and performance: protocols, servers, WWW interactivity, Internet-based distributed systems.

Bibliography

E. Wilde, *"WWW: Technical Foundations of the World Wide Web"*, Springer.

B. Krishnamurthy and J. Rexford, *"Web Protocols and Practice"* Addison-Wesley.

CS 426 - Computer Graphics

Y. Chrysanthou

Course Units: 4 / ECTS Credits: 8

Basic principles of Computer Graphics. Modeling of 3D virtual worlds. Scene construction, camera specification and projections of primitives, clipping, visibel surface determination, polygon rasterization (z-buffer). Texture mapping, local and global illumination and shadows. Practical experience with the use of OpenGl and VRML.

Bibliography

M. Slater, A. Steed and Y. Chrysanthou, *"Computer Graphics and Virtual Environments: From Realism to Real-Time"*, Addison-Wesley, 2001.

Th. Theocharis, A. Bem, *"Graphcis – Principles and Algorithms"*, (in Greek), Symmetria Publishers.

J. D. Foley, A. van Dam, S. K. Feiner, J. F. Hughes, R. L. Phillips, *"Introduction to Computer Graphics"*, Addison-Wesley, 1994.

CS 431 - Synthesis of Parallel Algorithms

M. Mavronicolas

ECTS units: 8

Introduction to parallelism. Models of parallel computation, complexity of parallel algorithms (time and processor complexity), performance measures for parallel algorithms (cost, speed-up, work, etc.). Efficient parallel algorithms in Combinatorics (sorting, merging, selection, searching, combinatorial optimization), Graph Theory, Geometry, Algebra and Matrix Theory. Complexity analysis of the algorithms on a Parallel Random Access Machine (PRAM). Comparison with other parallel models of computation such as linear array, mesh, cube and hypercube, tree, perfect shuffle, etc. Computational limitations of PRAMs.

Bibliography

J. Jaja, *"An Introduction to Parallel Algorithms"*, Addison-Wesley, 1992.

F. T. Leighton, *"Introduction to Parallel Algorithms and Architectures"*, Morgan-Kaufmann, 1992.

CS 432 - Distributed Algorithms

M. Mavronicolas

Course Units: 4 / ECTS Credits: 8

Formal models of distributed computing: shared memory versus message passing, determinism versus randomisation, concepts of synchronism, asynchrony and real-time. Design and analysis of distributed algorithms and impossibility/improbability results for fundamental problems such as mutual exclusion, consensus, synchronization, leader election, construction of minimum spanning trees. Fault tolerance: Byzantine generals, wait-free algorithms, fault degrees. Formal methods for proving correctness of distributed algorithms. Advanced topics. Special emphasis throughout the course on lower and upper bounds on time and memory.

Bibliography

H. Attiya and J. L. Welch, *"Distributed Computing: Fundamentals, Simulations and Advanced Topics"*, McGraw Hill, 1998.

N. Lynch, *"Distributed Algorithms"*, Morgan-Kaufmann, 1996.

CS 433 - Constraint Programming and Satisfaction

Y. Dimopoulos

Course Units: 4 / ECTS Credits: 8

Definition of constraint satisfaction problems. Constraint representation and complexity. Various forms of consistency. Backtracking and look-ahead techniques. Intelligent backtracking and condition for solution finding without backtracking. Heuristic and local methods for solution searching. Available commercial products. Study of problems from different application domains, their modeling and the complexity of various algorithms solving them.

Bibliography

K. Marriott and P. J. Stuckey, *"Programming with Constraints: an Introduction"*, MIT Press, 1998.

Research articles.

CS 434 - Logic Programming and Artificial Intelligence

A. Kakas

Course Units: 4 / ECTS Credits: 8

Basic principles of Logic Programming and implementation using the language Prolog. Relation of Logic Programming to modern considerations regarding Artificial Intelligence. Solving application problems drawn from the fields of Artificial Intelligence and Databases, making use of Logic Programming and Constraint Logic Programming.

Bibliography

L. Sterling and E. Shapiro, *"The Art of Prolog", 2nd edition*, MIT Press.

I. Bratko, *"PROLOG: Programming for Artificial Intelligence", 3rd edition*, Addison-Wesley.

CS 435 - Human-Computer Interaction

S. Retalis

Course Units: 4 / ECTS Credits: 8

Analysis of the human as a computer system user (knowledge models, graphical animation, cognitive models). Interactive technologies (input-output devices, window environments, systems for collaborative support, virtual reality). Methodologies for the design of interactive systems.

Bibliography

N. Avouris, *"Introduction to Human Computer Interaction"*, (in Greek), Diavlos Publishers, Athens 2000.

A. Dix, J. Finlay, G. Abowd and R. Beale, *"Human Computer Interaction"*, Prentice Hall, 2000.

CS 441 - Parallel and Distributed Software Engineering

G. Papadopoulos / A. Andreou

Course Units: 4 / ECTS Credits: 8

Introduction to advanced techniques for the development of component-based software systems, such as design and implementation of components, software architectures and middleware platforms. Emphasis on developing large-scale software systems in parallel and distributed environments.

Bibliography

C. Szyperski, *"Component Software: Beyond Object-Oriented Programming"*, Addison-Wesley, 2001.

I. Sommerville, *"Software Engineering", 6th edition*, Addison-Wesley, 2001.

R. Pressman, *"Software Engineering: A Practitioner's approach"*, McGraw-Hill, 2001.

CS 442 - Learning Systems

C. Schizas

Course Units: 4 / ECTS Credits: 8

Theoretical and practical training in the analysis, development and implementation of information learning systems. Studies of the basic methods of machine learning as they have been formulated in the last decade, including classical statistical methods such as: Bayes rule, discriminant and cluster analysis; artificial neural network methods such as supervised and unsupervised networks; other knowledge representation techniques like genetic algorithms.

Bibliography

G. Karayannis and G. Steinhauer, *"Pattern Recognition and Learning Machines"*, (in Greek), 3rd edition, Symeon Publishers, Athens 1991.

Eberhart, Simpson and Dobbins, *"Computational Intelligence PC Tools"*, AP-Professional.

S. Haykin, *"Neural Networks-A Comprehensive Foundation"*, McMillan College Publishing.

S. Kartalopoulos, *"Understanding Neural Networks and Fuzzy Logic"*, IEEE Press.

D. Goldberg, *"Genetic Algorithms in Search of Optimization & Machine Learning"*, Addison-Wesley.

Lecture notes, research articles and webpage: http://www.cs.ucy.ac.cy/courses/EPL442/

CS 443 - Artificial Intelligence and Expert Systems

E. Keravnou-Papailiou

Course Units: 4 / ECTS Credits: 8

Artificial Intelligence from the perspective of problem solving through search and the use of heuristics, knowledge representation and reasoning C basic concepts, predicate logic, associative networks, frams, production rules. Information Systems that represent expert knowledge as a main application area of techniques of Artificial Intelligence. "Embodiment" of specialized knowledge in Expert Systems with high level of competence. Applications of the Expert Systems Technology in various disciplines, such as Engineering, Medicine, Finance, etc. Appreciation of the nature of human expertise and the technology of Expert Systems.

Bibliography

E. Keravnou, *"Artificial Intelligence and Expert Systems"* (in Greek), Greek Open University, 2000.

P. Jackson, *"Introduction to Expert Systems", 3rd edition*, Addison-Wesley, 1999.

G. F. Luger και W. A. Stubblefield, *"Artificial Intelligence: Structures and Strategies for Complex Problem Solving", 3rd edition*, Addison-Wesley, 1998.

E. Rich and K Knight, *"Artificial Intelligence", 2nd edition*, McGraw-Hill, 1991.

CS 444 - Computational Intelligence Systems

C. Schizas

Course Units: 4 / ECTS Credits: 8

Neural networks, genetic algorithms, fuzzy systems, evolutionary computing and artificial life as constituent components of Computational Intelligence. Global overview of Computational Intelligence and its applications in solving "real" problems in various disciplines such as decision making support, classification, prognosis and prediction, systems optimization and recreational design.

Bibliography

L.S. Haykin, *"Neural Networks", 2nd edition*, Prentice Hall, 1999.

G. Karayannis and G. Steinhauer, *"Pattern Recognition"*, (in Greek), Symeon Publishers, 1991.

CS 445 - Digital Image Processing

C. Pattichis

Course Units: 4 / ECTS Credits: 8

Basic concepts of digital image processing: image representation, binary image processing, sampling, quentization, two-dimensional transforms, enhancement, restoration, coding and compression. Introduction to digital image analysis. Development and implementation of image processing algorithms with applications to industrial and biomedical systems.

Bibliography

R. C. Gonzalez and R. E. Woods, *"Digital Image Processing", 2nd edition*, Addison-Wesley, 2002.

CS 446 - Advanced Databases

G. Samaras

Course Units: 4 / ECTS Credits: 8

Theoretical approach to logical and physical design of databases. Algorithms for logical and physical design of databases. Primary and secondary indexing techniques. Advanced query processing and query optimisation. Query parallelism. Concurrency control and recovery, integrity and security of data. Distributed databases and introductory concepts distributed transaction processing involving multiple and heterogeneous databases. Problems of interfacing a database with software.

Bibliography

Elmasri and Navathe, *"Fundamentals of Database Systems"*, Benjamin Cummings, 1989.

Korth and Silberschatz, *"Database System Concepts"*, McGraw-Hill, 1986.

CS 447 - Artificial Intelligence

A. Kakas

Course Units: 4 / ECTS Credits: 8

Problem solving techniques in Artificial Intelligence. Formal methods of knowledge representation. Specific application areas such as multi-agent systems, robotics, expert systems, machine learning and natural language processing. Problems of current focus and investigation in Artificial Intelligence. Links between Artificial Intelligence and other fields of Computer Science such as Distributed Systems, Databases and Multimedia.

Bibliography

S. Russell and P. Norvig, *"Artificial Intelligence: A Modern Approach"*, Prentice Hall, 1995.

CS 448 - Intelligent Agents and Multi-Agent Systems

P. Moraitis

Course Units: 4 / ECTS Credits: 8

Introduction to Distributed Artificial Intelligence (DAI). Intelligent Agents (basic concepts, applications). Software Agents (main types, applications). Agent Societies. Distributed Programming of Actions. Formalisms in DAI: representation and reasoning based on Logic. Industrial and practical applications of DAI.

Bibliography

M. Wooldridge, *"Introduction to Multi-Agent Systems"*, J. Wiley, 2002.

G. Weiss, (Ed.), *"Multi-Agent Systems: A Modern Approach to Distributed Artificial Intelligence"*, 1999.

Lecture transparencies and research papers.

CS 449 - E-Business/E-Commerce

P. Evripidou

Course Units: 4 / ECTS Credits: 8

In-depth study of the technologies used for e-business/e-commerce. Fundamentals of the Internet and WWW technologies protocols such as TCP/IP, HTTP, HTML. Study of the different operational models and strategies of e-commerce and practical experience with systems such as Javascript, Java, DHTML, ASP, PHP, Websphere, and .NET. Study of e-Government that covers electronic transactions between Public authorities and Citizen/business. System security such as access security and cryptographic security, electronic signature and electronic payments.

Bibliography

Dietel and Nieto, *"E-business & E-commerce: How to program Dietel"*, Prentice Hall.

COURSES FOR STUDENTS OF OTHER DEPARTMENTS

CS 001 - Introduction to Computer Science

Visiting Professor

Course Units: 4 / ECTS Credits: 8

Fundamentals of Computer Science, the main historical events which have contributed in its development, and the possibilities if offers. Basic constituent elements of Computer Science and methods for making it valuable to other sciences and applications. Practical experience with application packages, basics of programming, and programming in a fourth generation language such as Logo.

Bibliography

"Computer Science and Information Systems", Lecture and Laboratory notes, (in Greek), Nicosia, 2002.

Ch. S. Parker, *"Computer Science – Present and Future"*, (in Greek), I. Floros Publishers, 1991.

L. & N. Long, *"Computers"*, 2001.

P. Makris, *"Introduction to Computer Science"*, (in Greek), Personal Publishers, 1992.

Ch. Koilia and Str. Kalafoutis, *"The first book in Computer Science"*, (in Greek), New Technology Publishers, 1992.

CS 002 - Introduction to Computer Science

Visiting Professor

Course Units: 4 / ECTS Credits: 8

Fundamentals of Computer Science, the main historical events which have contributed in its development, and the possibilities if offers. Basic constituent elements of Computer Science and methods for making it valuable to other sciences and applications. Practical experience with application packages, and the UNIX environment. Basic principles of programming in a fourth generation language.

Bibliography

"Computer Science and Information Systems", Lecture and Laboratory notes, (in Greek), Nicosia, 2002.

Ch. S. Parker, *"Computer Science – Present and Future",* (in Greek), I. Floros Publishers, 1991.

L. & N. Long, "Computers", 2001.

P. Makris, *"Introduction to Computer Science"*, (in Greek), Personal Publishers, 1992.

Ch. Koilia and Str. Kalafoutis, *"The first book in Computer Science"*, (in Greek), New Technology Publishers, 1992.

CS 003 - Introduction to Computer Science

Visiting Professor

Course Units: 4 / ECTS Credits: 8

Fundamentals of Computer Science, the main historical events which have contributed in its development, and the possibilities if offers. Basic constituent elements of Computer Science and methods for making it valuable to other sciences and applications. Practical experience with application packages, basics of programming, and programming in a fourth generation language such as Logo.

Bibliography

"Computer Science and Information Systems", Lecture and Laboratory notes, (in Greek), Nicosia, 2002.

Ch. S. Parker, *"Computer Science – Present and Future"*, (in Greek), I. Floros Publishers, 1991.

L. & N. Long, *"Computers"*, 2001.

P. Makris, *"Introduction to Computer Science"*, (in Greek), Personal Publishers, 1992.

Ch. Koilia and Str. Kalafoutis, *"The first book in Computer Science"*, (in Greek), New Technology Publishers, 1992.

CS 011 - Introduction to Information Society

Visiting Professor

Course Units: 4 / ECTS Credits: 8

Presentation of the formed framework for Information Society (IST). Basic concepts and constituent elements of IST, and the wider context for its application. Issues such as electronic government, telematics, digital business, electronic commerce, telemedicine, etc. Effects of IST on society and economy.

Bibliography

P. S. Anastasiades, *"In the Century of Information: approaching the New Digital Era"*, (in Greek), Livani Publishers, October 2000.

A. Duff, *"Information Society Studies"*, London: Routledge, 2000.

CS 031 - Introduction to Programming

Y. Chrysanthou

Course Units: 4 / ECTS Credits: 8

Basic principles of programming with emphasis on structured programming, abstraction, and the design, implementation, checking and debugging of modular programs. Mastering the material through laboratory exercises in a traditional programming language such as FORTRAN.

Bibliography

D. Mataras and F. Koutelieris, *"Fortran 90/95 for Scientists and Engineers"*, (in Greek), Tziola Publications, 1st edition, 2001.

L. Nyhoff and S. Leestma, *"Introduction to Fortran 90"*, Prentice Hall, 1999.

CS 032 - Programming of Problem-Solving Techniques

Visiting Professor

Course Units: 4 / ECTS Credits: 8

Introduction to the principles of programming with emphasis on structured programming, abstraction, and the design, implementation, checking and debugging of modular programs. Mastering the material through laboratory exercises in the C programming language. (Note: This course has as prerequisite either of the courses Cs001 or CS003.)

Bibliography

A. Tomaras, *"C – Theory and Practice"*, (in Greek), New Technology Publishers, 1994.

B. Kernighan & D. Ritchie, *"The Programming Language C"*, (in Greek), Kledarithmos Publishers, 1990.

D. Photiades, *"Programming Courses in C/C++"*, (in Greek), University of Ioannina, Greece, 1997.

P. Aitken και B. L. Jones, *"Manual of C"*, (in Greek), Giourda Publishers.

"C, 1990, Revision and Redesignation of ANSI X3.159-1989", American National Standards Institute, New York, NY, ANSI/ISO 9899-1990, American National Standard for Programming Languages.

"Data Structures and Algorithm Analysis in C++", Mark Allen Weiss, Addison-Wesley, 1994.

CS 033 - Introduction to Programming

Visiting Professor

Course Units: 4 / ECTS Credits: 8

Basic principles of programming with emphasis on structured programming, abstraction, and the design, implementation, checking and debugging of modular programs. Mastering of the material through laboratory exercises in a traditional programming language such as FORTRAN.

Bibliography

1. D. Mataras and F. Koutelieris, *"Fortran 90/95 for Scientists and Engineers"*, Tziola Publishers, 1st edition, 2001.

2. L. Nyhoff and S. Leestma, *"Introduction to Fortran 90"*, Prentice Hall, 1999.

POSTGRADUATE PROGRAMMES OF STUDIES

The Department offers postgraduate programs at the Ph.D. level in different specializations of Computer Science (since 1997), and a Master degree in Advanced Information Technologies (since 2000).

PH.D. PROGRAMME

A Ph.D. program comprises (a) the completion of postgraduate courses amounting to at least 60 ECTS units , (b) success in a comprehensive examination taken no later than the 5th semester of studies (c) approval of a research proposal, and (d) the submission and proposal of an original thesis which represents a substantial contribution to the relevant field of knowledge. The maximum period of time to earn a Ph.D. degree is eight (8) academic years.

M.SC. PROGRAMME

GENERAL INFORMATION

The M.Sc. program in Advanced Information Technologies is addressed primarily to graduate students who want to specialize in advanced areas of Computer Science (and possibly also want to prepare for research leading to a Ph.D. degree), but also to professionals who wish to earn a postgraduate degree.

A goal of the M.Sc. program is to equip researchers with the knowledge necessary to assist the Cypriot industry develop into new areas, thus increasing its competitiveness in relation to the industries of neighboring countries. The M.Sc. program has already attracted a number of Bachelor degree holders from other universities.

In order to be admitted, a candidate must possess a first degree in Computer Science or in a related subject from an accredited university with an overall grade of "Very Good". Any relevant industrial experience may be considered an additional advantage.

To earn an M.Sc. degree, a student must successfully complete 48 ECTS units from the Postgraduate Program and must prepare a Master's Thesis under the guidance of his/her academic supervisor. Students who may have successfully completed courses from another postgraduate program may be credited with up to 16 ECTS units. Each course carries 8 ECTS units, while the Thesis amounts to 12 ECTS units.

The Thesis may focus either on research or on development in technical areas, and must be original and/or prove a good knowledge and understanding of the specific area covered. The Thesis is usually completed within six months (full-time). Students are allowed, after the successful completion of courses equivalent to 32 ECTS units, to submit a topic for a Thesis on a Special Form, co-signed by his/her Research Advisor. The Thesis is submitted to the Department and defended in front of an Examination Committee according to rules set by the Departmental Council.

The duration of studies for acquiring an M.Sc. degree is normally three semesters, while it cannot exceed eight semesters. The program is structured as follows:

First Semester

3 core courses, chosen from a global list of core courses.

Second Semester

3 courses of special interest, chosen from a specific list of courses according to the student's immediate interests.

Third Semester

Master's Thesis.

DESCRIPTION OF COURSES FOR MASTER DEGREE

Core Courses

CS 601 - Distributed Systems

M. Mavronicolas

Course Units: 4 / ECTS Credits: 8

Basic concepts and principles of Distributed Systems. Communication, processes and synchronization. Naming. Distributed file systems and distributed operating systems. Security and cryptography. Distributed shared memory and its consistency. Distributed algorithms and distributed programming. Design and development of applications in a distributed environment. Faults in distributed systems: detection and recovery. Case studies and practical experience with programming project or programming exercises.

Bibliography

G. Coulouris, J. Dollimore and T. Kindberg, *"Distributed Systems: Concepts and Design", 3rd edition*, Addison-Wesley, 2001.

A. S. Tanenbaum and M. van Stenn, *"Distributed Systems: Principles and Paradigms"*, Prentice Hall, 2002.

CS 602 - Programming of Internet Systems and Services

M. Dikaiakos

Course Units: 4 / ECTS Credits: 8

Scalable Internet systems and services. Several aspects of software infrastructures used to develop, deploy and support Internet and WWW services. Software components of the WWW, communication protocols, data organization approaches and advanced issues from the recent research literature, such as Search Engine Design, Web Characterization and WWW Middleware. Laboratory component focusing on Internet programming with Java.

Bibliography

B. Krishnamurthy and J. Rexford, *"Web: Protocols and Practice"* Addison-Wesley.

CS 603 - Advanced Software Engineering

G. Papadopoulos / A. Andreou

Course Units: 4 / ECTS Credits: 8

Advanced principles of designing and implementing component-based systems. The meaning of component-based systems and component software. Patterns and software architectures. Basic principles of component software. Coordination programming. Middleware platforms. Different approaches to the development of middleware. Development of components in distributed systems. Related programming languages. Configuration management. Advanced issues in Software Engineering: Requirements Engineering. Real-time Software Design. Design with Reuse. User Interface Design. Legacy Systems. Software Change.

Bibliography

C. Szyperski, *"Component Software: Beyond Object-Oriented Programming"*, Addison-Wesley, 2001.

I. Sommerville, *"Software Engineering", 6th edition*, Addison-Wesley, 2001.

R. Pressman, *"Software Engineering: A Practitioner's approach"*, McGraw-Hill, 2001.

CS 604 - Artificial Intelligence

Y. Dimopoulos/C. Pattichis

Course Units: 4 / ECTS Credits: 8

Introduction to Artificial Intelligence. Intelligent Agents. Search. Constraint Satisfaction. Knowledge Representation and Reasoning. Planning. Representation of time and change. Introduction to Machine Learning. Introduction to Artificial Neural Networks. Single-layer and multilayer Neural Networks. Radial-basis Function Networks. Self-organizing Maps. Modular Neural Networks.

Bibliography

S. Russell and P. Norvig, *"Artificial Intelligence: A Modern Approach"*, Prentice Hall, 1995.

Research articles.

CS 605 - Advanced Computer Architecture I

Y. Sazeides/P. Trancoso

Course Units: 4 / ECTS Credits: 8

A spectrum of issues relating the technology, performance and organization of computer systems. Special emphasis on contemporary research trends.

Bibliography

J. L. Hennessy and D. A. Patterson *"Computer Architecture: A Quantitative Approach"*, Morgan-Kaufmann Publishers, 3rd edition, 2003.

Courses of Specific Interest

CS 651 - Data Management for Mobile Computing

G. Samaras

Course Units: 4 / ECTS Credits: 8

Introduction (Wireless technologies, Architectures, Applications, Limitations). Mobile computing models. Theoretical models for mobile computation. Support for information retrieval in the wireless environment. Data management and recovery issues in the mobile and wireless environment. Dynamic move of computation. Indicative application. Open problems.

Bibliography

E. Pitoura and G. Samaras, *"Data Management for Mobile Computing", 1st edition*, Kluwer Academic Publishers, 1998.

CS 652 - Electronic Commerce

P. Evripidou

Course Units: 4 / ECTS Credits: 8

In-depth analysis of the techniques involved in E-Commerce. Introduction to Software Engineering techniques for the client sever model and business models for E-Commerce. Infrastructure, XML and data. Architecture of E-Commerce systems. Electronic Government. Cryptographic systems and Electronic Signatures, Electronic payments, Mobile and wireless commerce. Data Mining. Software engineering techniques for Internet programming. Business models.

Bibliography

G. P. Schneider and J. T. Perry, *"Electronic Commerce"*, Thomson Learning.

CS 653 - Computer Networks and the Internet

A. Pitsillides

Course Units: 4 / ECTS Credits: 8

Networking Issues featuring the Internet. Internet and Networking Technologies. Applications and Services (e.g. the Web). TCP/IP suite of protocols. Interactivity and Real Time applications over the Internet (e.g Video conferencing, VoIP, RTP/RTCP, SIP). Quality-of-Service (QoS). New Networking Architectures, Protocols, and Standards (e.g. DiffServ, IPv6, MPLS), Networking Performance Evaluation, Traffic Modeling; Congestion Control and Resource Allocation. Network Design and Optimization. Mobile and Wireless Networks. Internet Security and System Issues.

Bibliography

J. F. Kurose and K. W. Ross, "Computer Networking - A Top-Down Approach Featuring the Internet", Addison-Wesley, 2nd edition, 2003.

Research articles (webpage: http: www.cs.ucy.ac.cy/courses/EPL653/ lectures.htm).

CS 654 - E-learning Technologies

S. Retalis

Course Units: 4 / ECTS Credits: 8

Theories of learning. Different types of educational software (e-books, simulations, drill & practice, Logo-like environments, intelligent tutoring systems, Learning management systems, etc). Courseware engineering methodologies. Didactics of informatics.

Bibliography

S. Retalis, *"Advanced Hypermedia Systems for Learning"*, (notes in Greek).

M. Driscoll, *"Web-Based Training: Designing e-Learning Experiences"*.

S. M. Alessi, S. R. Trollip *"Multimedia for Learning: Methods and Development"*, *3rd edition*.

CS 655 - Advanced Computer Architecture II

Y. Sazeides/P. Trancoso

Course Units: 4 / ECTS Credits: 8

Advanced Computer Architecture with emphasis on three basic topics: Support for the execution of parallel programs, parallel architectures, and interaction between software and architecture.

Bibliography

D. A. Patterson and J. L. Hennessy, *"Computer Architecture: A Quantitative Approach", 3rd edition*, Morgan-Kaufmann, 2002.

D. Culler, J. P. Singh and A. Gupta, *"Parallel Computer Architecture: A Hardware/Software Approach"*, Morgan-Kaufmann, 1998.

CS 656 - Computer Graphics: Modeling and Realism

Y. Chrysanthou

Course Units: 4 / ECTS Credits: 8

Principles of Computer Graphics. Modeling of 3D virtual worlds. Scene construction, camera specification and projections of primitives, clipping, visible surface determination, polygon rasterisation (z-buffer), texture mapping, local and global illumination and shadows. Photo-realism and light transport. Practical excercises with the use of OpenGl and VRML.

Bibliography

M. Slater, A. Steed and Y. Chrysanthou, *"Computer Graphics and Virtual Environments: From Realism to Real-Time"*, Addison-Wesley, 2001.

Th. Theocharis, A. Bem, *"Graphics – Principles and Algorithms"*, (in Greek), Symmetria Publishers.

A. Watt, *"3D Computer Graphics", 3rd edition*, Addison-Wesley, 2001.

Egerton and Hall, *"Computer Graphics, Mathematical First Steps"*, Prentice Hall, 1998.

CS 661 - Multi-Agent Systems

Visiting Professor

Course Units: 4 / ECTS Credits: 8

Agent categories. Agent architectures. Multi-agent systems principles and characteristics. Coordination through distributed planning. Negotiation Agent communication languages and dialogues. Agent cooperation. Argumentation Agent oriental Software Engineering. Applications on E-Commerce, Enterprise Modeling, etc.

Bibliography

M. Wooldridge, *"Introduction to Multi-Agent Systems"*, J. Wiley, 2002.

G. Weiss, (Ed.), *"Multi-Agent Systems: A Modern Approach to Distributed Artificial Intelligence"*, 1999.

Lecture transparencies and research papers.

CS 662 - Machine Learning and Data Mining

C. Pattichis

Course Units: 4 / ECTS Credits: 8

Data warehousing and OLAP technology for data mining. Data preprocessing. Data mining primitives, languages and system architectures. Algorithms, applications, and comparison of: mining association rules, classification, prediction, and clustering analysis. Mining complex types of data. Data mining applications and trends in data mining.

Bibliography

J. Han and M. Kamber, "Data Mining – Concepts and Techniques", Morgan-Kaufmann, 2000.

CS 663 - Computational Logic

A. Kakas

Course Units: 4 / ECTS Credits: 8

Computational Logic as a cut-edge research area of Artificial Intelligence. Classical Logic and aspects of Computational Logic. Applications of Computational Logic to areas such as Cognitive Robotics, Multi-Agent Systems and the Semantic Web.

Bibliography

A Kakas and F. Sadri (Eds), *"Computational Logic: Logic Programming and Beyond"* Parts I and II, Springer-Verlag, 2002.

CS 664 - Systems Analysis and Verification

A. Philippou

Course Units: 4 / ECTS Credits: 8

Formal methods for the modeling, analysis and development of asynchronous systems, such as parallel and distributed systems, communication and security protocols, and embedded software. Hoare logic for sequential and parallel programs. Temporal logics and model-checking: state, past and future formulae, safety, liveness, guarantee and reactivity properties. Evaluation of real-time and probabilistic behavior. Process algebras: syntax and semantics, observational equivalences, axiom systems. Automated verification tools.

Bibliography

D. Peled, *"Software Reliability Methods"*, Springer-Verlag, 2001.

Berard et al, *"Systems and Software Verification, Model-Checking Techniques and Tools"*, Springer-Verlag, 2001.

GENERAL INFORMATION

Library Use

The University's Library is equipped with a large number of books and scientific journals in Computer Science. These include books that will assist the students in mastering the taught course materials, and, in particular, the books recommended or suggested by Instructors. Regulations for using the Library are separately handed out to the students at the beginning of each academic year.

Electronic Mail

The use of Electronic Mail for communication between academic staff and students, and also between students themselves, is considered a must, and students are strongly encouraged to learn as soon as possible how to effectively use Electronic Mail. However, it is emphasized that the use of Electronic Mail is not a right of the students, but rather a service provided by the Department. In cases where bad use of Electronic Mail is noticed, the Department reserves the right to disallow its use.

Laboratory Equipment Use Regulations

Students are kindly requested to respect fundamental principles of professional behaviour regarding health and safety in common rooms and responsible use of laboratory equipment. In particular, the following are indicative examples of disallowed tasks:

- Access to computer systems using somebody else's account.
- Inappropriate use of Electronic Mail.
- Use of computer systems for purposes other than the "normal" (e.g., development of commercial products, disturbance of users, etc.).
- Use of software products other than those provided by the Computer Center, without the consent of the Computer Center or the Department, respectively.
- Attempt to access confidential information.
- Copying software products that belong to others, in violation of international laws of liability.

More complete information regarding Regulations and Time Table for using the Computer Laboratories is handed out to the students at the beginning of each academic year.

ACADEMIC STAFF

Andreas Andreou
Lecturer
E-mail: aandreou@cs.ucy.ac.cy
Tel.: 22-892692

Yiorgos Chrysanthou
Assistant Professor
E-mail: yiorgos@cs.ucy.ac.cy
Tel.: 22-892719

Marios Dikaiakos
Assistant Professor
E-mail:: mdd@cs.ucy.ac.cy
Tel.: 22-892720

Yannis Dimopoulos
Assistant Professor
E-mail: yannis@cs.ucy.ac.cy
Tel.: 22-892718

Paraskevas Evripidou
Associate Professor
E-mail: evripidu@cs.ucy.ac.cy
Tel.: 22-892696

Antonis Kakas
Associate Professor
E-mail: antonis@cs.ucy.ac.cy
Tel.: 22-892706

Elpida Keravnou-Papaeliou
Professor
E-mail: elpida@cs.ucy.ac.cy
Tel.: 22-892098

Marios Mavronicolas
Associate Professor
E-mail: mavronic@cs.ucy.ac.cy
Tel.: 22-892702

George Papadopoulos
Associate Professor
E-mail: george@cs.ucy.ac.cy
Tel.: 22-892693

Constantinos Pattichis
Associate Professor
E-mail: pattichi@cs.ucy.ac.cy
Tel.: 22-892697

Anna Philippou
Lecturer
E-mail: annap@cs.ucy.ac.cy
Tel.: 22-892699

Andreas Pitsillides
Associate Professor
E-mail: cspitsil@cs.ucy.ac.cy
Tel.: 22-892695

Symeon Retalis
Lecturer
E-mail: retal@cs.ucy.ac.cy
Tel.: 22-892716

George Samaras
Associate Professor
E-mail: cssamara@cs.ucy.ac.cy
Tel.: 22-892698

Yannos Sazeides
Lecturer
E-mail: yanos@cs.ucy.ac.cy
Tel.: 22-892704

Christos N. Schizas
Professor
E-mail: schizas@cs.ucy.ac.cy
Tel.: 22-892045

Pedro Trancoso
Lecturer
E-mail: pedro@cs.ucy.ac.cy
Tel.: 22-892703

Department of Computer Science
University of Cyprus
75 Kallipoleos Street
P.O. Box 20537
CY-1678 Nicosia
CYPRUS

Tel.: +357-22892700
Fax: +357-22892701
E-mail: cs@ucy.ac.cy
http://www.cs.ucy.ac.cy