TOWARDS AN EUROPEAN CERTIFICATION OF COMPUTER SCIENCE UNIVERSITY CURRICULA

Agostino Cortesi
Università Ca' Foscari di Venezia
cortesi@unive.it

Enrico Nardelli
Università di Roma "Tor Vergata"
nardelli@mat.uniroma2.it

ABSTRACT

The paper describes the certification mark promoted by the Italian Association of Computer Science University Professors (GRIN) for undergraduate and graduate degree programs in Computer Science. The certification process yields a system of comparable and transparent curricula that enables the comparison of different sites and supports student mobility across different Universities. It can be seen as a first step towards a European Certification of Computer Science University Curricula.

Keywords

Quality assurance, Transparency of curricula.

1. INTRODUCTION

Computer Science university degree programs are required both to provide students with a solid scientific and technological background, and to cover as much as possible the wide spectrum of disciplines exploiting the computer science methodology.

The autonomy that has been granted to universities in the specification of their curricula forces the faculties to qualify their products with respect to the plethora of courses that are offered in the educational market. Students and families often get confused by radio and TV commercials. Also when focusing just on curricula offered by the university system, it is very difficult to evaluate the values of curricula that look very similar. The same holds for enterprises and recruitment agencies, having difficulties to classify the different universities with respect to the quality of each degree program as the content and the structure of the whole university system has deeply changed in the last decade.

In this context, each university delivering a degree program in Computer Science (CS) should be fair enough to guarantee that their curricula cover the basic spectrum of scientific knowledge in CS, and that courses are given by qualified professors.

Unfortunately, the conditions above are not always satisfied. For instance, there are university degree programs in CS where not even one faculty member has a PhD neither in Computer Science nor in Computer Engineering.

GRIN, the Italian Association of Computer Science University Professors (http://www.di.unipi.it/grin) promoted a common effort involving almost all of the Italian universities towards the elicitation of the "product qualities" of undergraduate degree programs in CS, and characterized the constraints to be fulfilled in order to obtain the GRIN quality certification. Rules and results of this certification process are made public, in order to provide families and enterprises with information to make the right choice. There is no other similar initiative in the Italian academic context, and as far as we know, in Europe only the Chemistry Eurobachelor has similar purposes. In order to manage this certification process, a web site is maintained by Università di Roma "Tor Vergata". This project is supported by the Italian Council of University Deans (CRUI).

The result of the 2004, '05 and '06 certification process are available at the URL http://grin.informatica.uniroma2.it/ by connecting as generic user. In 2006, more than 40 undergraduate degree programs in CS have been granted the GRIN certification mark (out of 54).

2. CERTIFICATION REQUIREMENTS

The GRIN Association decided to adopt the following criteria in order to avoid the need for heavy organizational duties: the quantity of data to be treated should be quite limited; data should be easy to get and to check; data should be already available at each site, as part of the usual public information provided to potential students.

The GRIN quality certification is based on the verification of the fulfilment of a set of constraints on the programs.

The general guidelines to define the certification rules are as follows:

• The percentage of courses in CS in the degree program has to be relevant.

• The main areas of CS should be properly covered

• The degree program should not too much focussed on a single area.

Two certification levels were designed, the first one aimed at undergraduate degree programs (certificazione base), the second one for graduate degree programs or for very focussed undergraduate programs (certificazione avanzata).

Observe that also graduate programs having an interdisciplinary character might be granted a first level certification.

Taking inspiration from the ACM-IEEE classification, a list of 11 main CS areas was identified, as depicted in Fig.1, and a detailed list of subtopics were associated to each of them.

A. Foundations
B. Algorithms
C. Programming
D. Computer Languages
E. Computer Architectures
F. Operating Systems
G. Data Bases
H. Network Computing
I. Software Engineering
L. Human Computer Interaction – Graphics - Multimedia
M. Knowledge Representation

Fig.1: Main CS Areas

The certification rules are defined in terms of credits (cfu, where one cfu corresponds to 25 hours of learning activities, including lectures, training, and individual study for the average student).

In order to be eligible for the GRIN quality certification mark, each University degree program in CS must satisfy the following constraints:

1. At least 78 cfu must be assigned to learning activities in CS or in Computer Engineering.
2. At least 60 cfu (out of the 78 above) must be assigned to learning activities in the 11 areas listed in Fig.1
3. At least 7 areas (out of the 11 listed in Fig.1) must be covered by at least 6 cfu.

It is easy to verify that these rules strictly correspond to the three criteria presented above. The first rule guarantees that more than 1/3 of the program is specifically dedicated to CS topics; the rest of the credits should cover mathematical and physical foundations, more specialized topics, or complementary aspects (e.g., legal, economical, and ethical issues). The second rule guarantees a good coverage of the main areas of CS; observe that credits assigned to the same area can be spread among different courses. Finally, the third rule prevents from degree programs whose scope is too narrow: at least half of the 11 areas must be properly covered, say by at least 48 hours of lectures.

We are aware that this approach only provides benchmarks for the scope of curriculum and that other issues need to be considered in the future to get to a complete accreditation process.

3. THE SUPPORTING WEB APPLICATION

Each year, the chair of each CS degree program may apply for the GRIN quality certification by inserting in the certification web site the data concerning the activated curricula, and the syllabus of each course taught. For each course, the following information has to be provided: total number of credits, number of credits labelled as "computer science", and corresponding area (see Fig.1). A brief description of the contents of each credit (ects) in the 11 areas has to be inserted as well, according to the sub-areas depicted in Fig.2. Observe that this data results in a system of comparable and transparent curricula that allows both the comparison of different curricula and student mobility. Three screenshots of the systems are depicted in Fig. 3 to Fig. 6.

4. THE CERTIFICATION AUTHORITY

The quality certification is issued by AICA, a certification authority that grants other CS certification marks like ECDL and EUCIP. It is important to notice that all the data about the certified courses are made public in the web: they offer the reader a complete and synoptic picture of the main academic degree programs in Computer Science (also of the universities that do not fulfil the certification requirements).

5. CONCLUSIONS

The GRIN quality certification mark of university degree programs in Computer Science might be the basis for discussion about a European certification of Computer Science curricula, as it yields a system of comparable and transparent curricula that allows the comparison of what is offered by the different institutions, and most of all it supports student mobility in an European context. As a future work, we also aim at comparing our approach with the UK benchmarks (HTTP://WWW.QAA.AC.UK).

6. REFERENCES

[1] ACM-IEEE Computing Curricula 2001 Joint Task Force on Computing Curricula. http://www.computer.org/education/cc2001/

[2] Cortesi A., and Nardelli E., The Quality Certification Mark for Italian University Degree Programs in Computer Science. *Proc. 8th IFIP World Conference on Computers in Education, Cape Town, South Africa*, July 2005, ISBN 1-920-01711-9

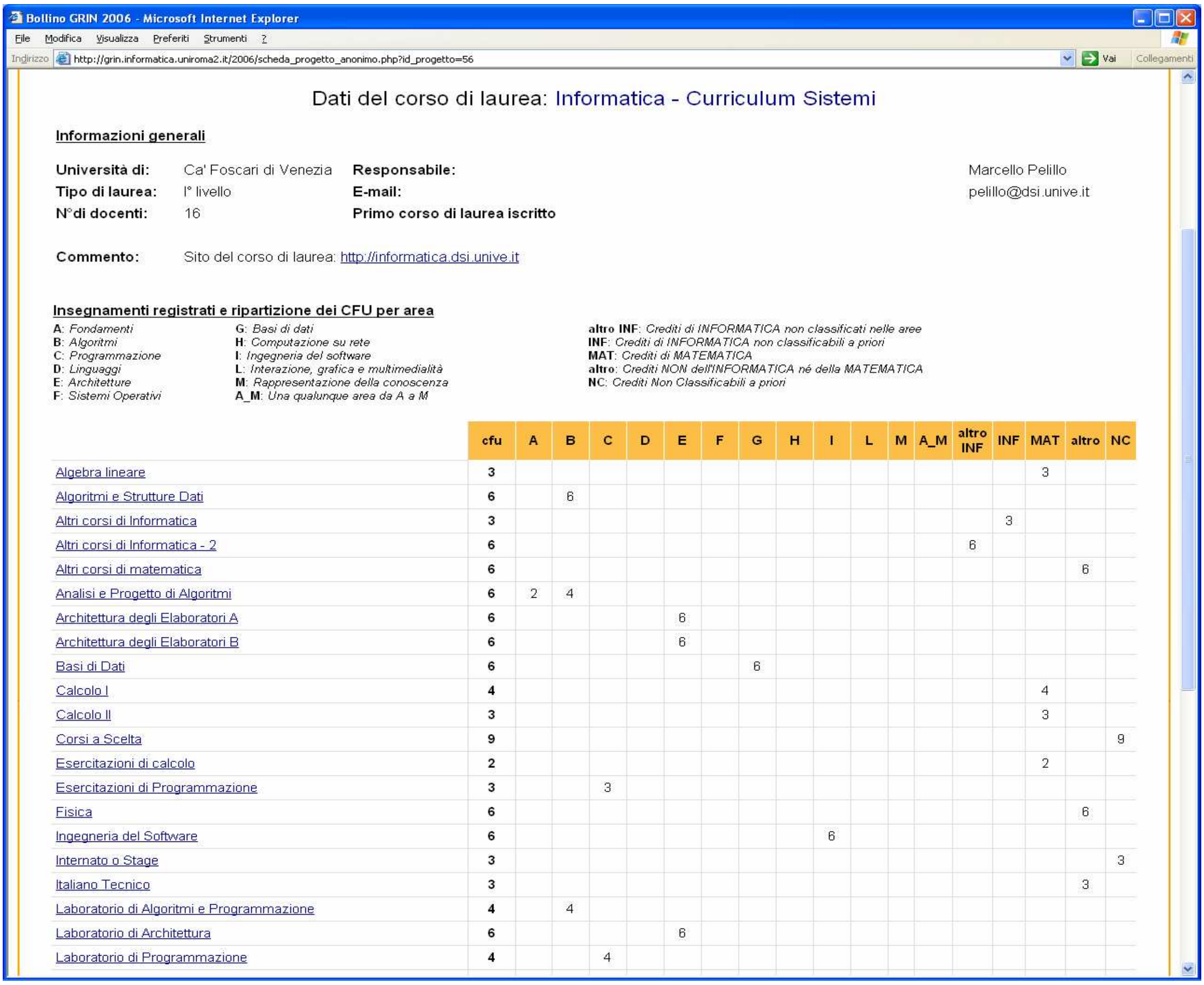
A. **FOUNDATIONS** * ALF - Automata and Formal Languages * CAL -Computability * COM - Complexity * SLP - Semantics of Programming Languages TIC - Information Theory L - Logics SD - Dynamical Systems V – Other Topics	B. **ALGORITHMS** * SDF - Fundamentals of Data Structures * TAPA - Basic Techniques for Algorithm Analysis and Design * A - Fundamentals of Algorithms * ASC - Algorithms for Combinatorial Structures TAA - Advanced Algorithmic Techniques SDA - Advanced Data Structures AD -Distributed Algorithms AP - Parallel Algorithms AN - Numerical Algorithms V - Other Topics
C. **PROGRAMMING** * PSA - Problem Solving and Algorithms * SS - Syntax and Semantics * CB - Fundamentals of Programming Structures * P – Procedures and Functions * R - Recursion * SDTD- Basic Data Structures and Abstract Data Types * SCP - Program Development and Correctness * POO - Object Oriented Programming PP - Programming Paradigms PPCC - Concurrent Programming V – Other Topics	D. **LANGUAGES** * LF - Formal Languages * S - Semantics * MATR - Abstract Machines and Techniques for Implementing Programming Languages * TTCI - Compilers and Intepreters PLN - Programming Languages Paradigms ALC - Linguistic Abstraction and Compositionality MP - Programming Methodologies TAV - Analysis and Verification Techniques V- Other Topics
E. **ARCHITCTURES** * CCS - Combinatorial and Sequential Circuits * AC - Computer Arithmetics * LIS - Instruction Set Level * LMP - Microprogramming Level * ASS - Assembly Language GM - Memory Management GIO - Input/Output Management VMP - Performance Evaluation and Optimization AA - Advanced Architecture V – Other Topics	F. **OPERATING SYSTEMS** * SC - Structure and Component of an Operating System * GSP - Process Management and Synchronization * GM - Memory Management * FS - File System * AMM -System Administration GP - Peripheral Management GCA - Access Control and Management PS - System Programming MA - Models and Architectures of Operating Systems SAA - Operating Systems for Advanced Computing Architectures V – Other Topics
G. **DATA BASES** * ML - Logical Models * PC - Conceptual Design * PL - Logical Design * LI - Query Languages * DBMS - Data Base Management Systems LP - Data Base Programming Languages NBD - Data Base Normalization OFGI - Physical Organization and Query Processing TCR - Transactions, Concurrency, and Recovery BDA - Advanced Data Bases V – Other Topics	H. **NETWORK COMPUTING** * FCD - Fundamentals of Distributed Computation * ARTC - Computer Network Architectures * PT - Protocols * SR - Network Security * MIR - Models for Interaction in Networks SRM - Network Operating Systems and Network Programming Middleware PASR - Application and Service Programming GRC - Network Management DR - Network Devices V – Other Topics
I. **SOFTWARE ENGINEERING** * PSS - Software Development Processes * LMS - Software Modeling Languages * AR - Requirement Analysis * ASW - Software Architecture * PSC - Software Design and Coding * TVV - Testing, Verification, and Validation AS - Development Environment MES - Software Maintenance and Evolution EPGS - Economics of Software Production and Management MSQ - Software Measurement and Quality EPG - Ethical, Professional and Legal Issues V Other Topics	L. **HC INTERACTION, GRAPHICS AND MULTIMEDIA** * MMPI - Models and Methods for Interaction Design * PMTV - Principles, Methods and Techniques for Interface Evalution * IMW - Hypertexts, Multimedia and WWW TMI – Human-Computer Interaction: Theories and Models PIRV - Interaction Paradigms and Virtual Reality SIAS - Interaction Support System and Development Environment MG - Geometrical Modelling RV - Rendering and Visualization ESM - Multimedia Signal Processing V – Other Topics
M. **KNOWLEDGE REPRESENTATION** * RP - Problem Representation * SBC - Knowledge-Based Systems * LPD - Logics and Declarative Programming ARC - Knowledge Acquisition and Representation AI - Intelligent Agents RA - Automated Reasoning AASC - Automated Learning and Knowledge Discovery BC - Knowledge Bases AIA - Artificial Intelligence Application V – Other Topics	

Fig. 2: The GRIN classification of Computer Science Education Areas

Fig. 3: The home page of the Certification Site

Corsi di laurea di primo livello	Università di	Responsabile	Certificazione
Informatica - Percorso A: Sistemi basati su conoscenza	Bari	Maria Francesca Costabile	AVANZATA
Informatica - Percorso B: Progettazione del software	Bari	Maria Francesca Costabile	AVANZATA
Informatica - Percorso C: Sistemi di elaborazione intelligenti	Bari	Maria Francesca Costabile	AVANZATA
Informatica e Comunicazione Digitale - Indirizzo: Sistemi di e-topics	Bari	Vito Leonardo Plantamura	AVANZATA
Informatica e Comunicazione Digitale - Indirizzo: Sistemi Software Avanzati	Bari	Vito Leonardo Plantamura	AVANZATA
Informatica e Tecnologie per la Produzione del Software	Bari	Giuseppe Visaggio	AVANZATA
Informatica	Bologna	Maurizio Gabbrielli	BASE
Scienze dell'Informazione	Bologna - sede di Cesena	Marilena Barnabei	BASE
Informatica - Curriculum Sistemi	Ca' Foscari di Venezia	Marcello Pelillo	BASE
Informatica - Curriculum Applicazioni	Ca' Foscari di Venezia	Marcello Pelillo	BASE
Informatica - Curriculum Gestionale	Ca' Foscari di Venezia	Marcello Pelillo	BASE
Informatica	Cagliari	G. Michele Pinna	BASE
Informatica - Indirizzo Tecnologie Informatiche	Camerino	Flavio Corradini	BASE
Informatica - Indirizzo Informatica e Management	Camerino	Flavio Corradini	BASE
Informatica	Catania	Giovanni Gallo	AVANZATA
Laurea in Informatica	Federico II di Napoli	Adriano Peron	BASE
Informatica	Firenze	Elena Barcucci	BASE
Informatica - Curriculum Progettazione Software	Genova	Enrico Puppo	BASE

Fig. 4: The (partial) list of 2006 certified curricula



Bollino GRIN 2006 - Microsoft Internet Explorer

File Modifica Visualizza Preferiti Strumenti ?

Indirizzo http://grin.informatica.uniroma2.it/2006/scheda_progetto_anonimo.php?id_progetto=56

Dati del corso di laurea: Informatica - Curriculum Sistemi

Informazioni generali

Università di: Ca' Foscari di Venezia
Tipo di laurea: I° livello
N°di docenti: 16

Responsabile: Marcello Pelillo
E-mail: pelillo@dsi.unive.it
Primo corso di laurea iscritto

Commento: Sito del corso di laurea: http://informatica.dsi.unive.it

Insegnamenti registrati e ripartizione dei CFU per area

A: *Fondamenti*
B: *Algoritmi*
C: *Programmazione*
D: *Linguaggi*
E: *Architetture*
F: *Sistemi Operativi*
G: *Basi di dati*
H: *Computazione su rete*
I: *Ingegneria del software*
L: *Interazione, grafica e multimedialità*
M: *Rappresentazione della conoscenza*
A_M: *Una qualunque area da A a M*
altro INF: *Crediti di INFORMATICA non classificati nelle aree*
INF: *Crediti di INFORMATICA non classificabili a priori*
MAT: *Crediti di MATEMATICA*
altro: *Crediti NON dell'INFORMATICA né della MATEMATICA*
NC: *Crediti Non Classificabili a priori*

	cfu	A	B	C	D	E	F	G	H	I	L	M	A_M	altro INF	INF	MAT	altro	NC
Algebra lineare	3															3		
Algoritmi e Strutture Dati	6		6															
Altri corsi di Informatica	3														3			
Altri corsi di Informatica - 2	6													6				
Altri corsi di matematica	6																6	
Analisi e Progetto di Algoritmi	6	2	4															
Architettura degli Elaboratori A	6					6												
Architettura degli Elaboratori B	6					6												
Basi di Dati	6							6										
Calcolo I	4															4		
Calcolo II	3															3		
Corsi a Scelta	9																	9
Esercitazioni di calcolo	2															2		
Esercitazioni di Programmazione	3			3														
Fisica	6																6	
Ingegneria del Software	6									6								
Internato o Stage	3																	3
Italiano Tecnico	3																3	
Laboratorio di Algoritmi e Programmazione	4		4															
Laboratorio di Architettura	6					6												
Laboratorio di Programmazione	4			4														

Fig. 5: The description of a CS curriculum. Each course is classified within the CS areas (see Fig. 1), and it may cover more than one of them (for instance, the 6 cfu of "Analisi e Progetto di Algoritmi" are classified in the "Foundation" and in the "Algorithms" areas).

Fig. 6: The syllabus of a course. In the example, course on Algorithm Design is described. The content of each credit unit (corresponding to 8 hours in class) is described in terms of sub-area labels (see Fig.2), with additional specifications.