Roberto Bruni, Stefano Chessa, Antonio Cisternino,
Gianna M. Del Corso, Nadia Pisanti (Eds.)

WiGoWiN

1st Internal Conference on

What is Going on and What is Next?

Pisa, Italy, May 26–27, 2010

Proceedings

Dipartimento di Informatica, Università di Pisa

Preface

This volume contains the proceedings of WiGoWiN, the “1st Internal Conference on What is Going on and What is Next?”, held in Pisa, Italy on May 26–27, 2010.

The event aimed at establishing a forum for spreading knowledge on current research activity and main achievements carried out at the Computer Science Department, promoting the exchange of ideas about future trends/perspectives for research in computer science, and strengthening collaborations between different research groups.

Presentations on the above issues have been given by faculty members of the Computer Science Department to a wide, heterogeneous audience that included MSc and PhD students, postdocs, and Computer Science related researchers from Academy, Industry and other research Institutions in Pisa.

The event was initially planned to last one day, but since the call for abstracts was very successful it needed to be expanded on one day and a half. The Organizing Committee received 29 abstracts. Due to time constraints, it was possible to allocate 25 presentation slots only, and therefore some members of the Organizing Committee volunteered not to present some activities they co-authored to leave room for those of other colleagues. This volume collects all 29 abstracts, listed according to the alphabetical order of authors. Although some abstracts reported work carried out in collaborations with researchers from other Universities and Institutions, for conciseness in the Table of Contents we list only authors from the Computer Science Department of the University of Pisa. Full credits can be found in each abstract.

The topics of the event included talks on: motivations, descriptions and applications of recent results; methodologies and tools for addressing novel challenges; key achievements and main open problems of specific research fields; experimental teaching experiences; interdisciplinary approaches; visionary perspectives on where computer science should head to in a particular area, in the near future.

Our best thanks go to Prof. Giorgio Levi and the direction of the Dipartimento for being the main sponsor of this event. We also thank our Preside Prof. Umberto Mura for making Aula Magna available for presentations. Finally, we want to thank our colleagues Prof. Chiara Bodei, Prof. Gianluigi Ferrari, Prof. Ornella Menchi, and Prof. Susanna Pelagatti who, jointly with ourselves, made it possible to cover event fees.

Pisa, May 2010

Antonio *Gianna* *Nadia* *Roberto* *Stefano*

Table of Contents

References

1. M. Bartoletti, P. Degano, G. Ferrari, and R. Zunino. Semantics-based design for secure web services. *IEEE Trans. Software Eng.*, 34(1):33–49, 2008.
2. M. Bartoletti, P. Degano, and G. Ferrari. Planning and verifying service composition. *Journal of Computer Security*, 17 (5), 2009.
3. , M. Bartoletti, P. Degano, G. Ferrari, and R. Zunino. Model Checking Usage Policies Trustworthy Global Computing, Lecture Notes in Computer Science, 5474, 2009.
4. M. Bartoletti, P. Degano, G. Ferrari, and R. Zunino. Local policies for resource usage analysis, *ACM Trans. Program. Lang. Syst.*,31 (6), 2009.
5. M. Bartoletti, G. Costa, P. Degano, F. Martinelli, and R. Zunino. Securing Java with Local Policies, Journal of Object Technology, 8(4) 2009.
6. C. Bodei, L. Brodo, R. Bruni. Static Detection of Logic Flaws in Service-Oriented Applications. Proceedings of the International Workshop on Issues in the Theory of Security (WITS'09), LNCS 5511, pp. 70-87, 2009.
7. C. Bodei, L. Brodo, P. Degano, H. Gao. Detecting and preventing type flaws at static time. To appear in Journal of Computer Security, 2010

Teaching Computer Science at School: some ideas

Chiara Bodei, Roberto Grossi, and Maria Rita Laganà

joint work with
Marco Righi

Abstract. As a young discipline, Computer Science does not rely on longly tested didactic procedures. This allows the experimentation of innovative teaching methods at schools, especially in early childhood education. Our approach is based on the idea that abstracts notions should be gained as the final result of a learning path made of concrete and touchable steps. To illustrate our methodology, we present some of the teaching projects we proposed.

1 Our Didactic Projects

Robot [6, 4] Using and building robots is one of the most effective and consolidated activities to naturally initiate children to computer programming. It is sufficient to think how easily they understand that a robot without a program is a useless object and the program is the only thing able to "give life" to it. Programming concepts could be very difficult to teach without any reference to concrete things. Pupils are not aware that they are learning and implicitly using not trivial concepts like the one of algorithm. In particular, we played with the Lego programmable robots and we helped children to design them in order to simulate behaviour that recall "biological" ones, like: be frightened of the light, follow a clue, according to an imprinting phenomenon, find the way out of a labirinth, follow a light, as moths do, produce malicious look that invites to cuddle. In addition, robot construction offered a chance to cooperate, and realise how important is to work together as a team, like in a real team of industrial design. Since there are several required skills, pupils can find their own roles and feel involved and useful.

Danza dei bit [1, 3, 2] We have experimented a multidisciplinary approach to teach the binary representation of numbers and their arithmetic and also the basic notions on the computer architecture. We pretended that children had been turned in bits and lived in Computer City. As insiders, by acting in the imaginary town and by playing their roles, they could understand how numbers are formed, which are the parts a Computer is composed with and how it works. An operetta, called "Danza dei bit" (Bits' dance), concluded the experience. Children as actors have been taken to actively understand the target notions, with the help of dialogues, music and also the used sets.

AI-Game [5] Another didactic experimentation proposed to 9-10 year old pupils of the Primary School is based on a programmable LOGO-like environment, called AI-Game, that we developed to smoothly introduce children to the art of programming. Without any programming experience, children can easily create drawings and animations. Four graphic characters can indeed move and act in a grid on the screen, following instructions, like: move forward, move backward, pick a ball, Children can compose programs, i.e. sequences of these instructions, in order to create simple games, consequently being for once behind and not in front of the video game console. Focussing on the construction of new games, children have mastered the basic programming notions quite naturally, because they represent just a mean to play. We only gave pupils the basic instructions and let them to ask us for new ones. It comes as a nice surprise that they also required something very similar to the standard conditional and iterative instructions. From their point of view, these composed instructions represented just a way to obtain more succinct programs.

Kara [7] Programming with finite state automata and data structures in the form of matrices is another way to develop and coordinate the abstraction capabilities of pupils: every automaton state represents a class of possible computations for the same configuration. Kara is a software tool, developed by W. Hartmann, J. Nievergelt and R. Reichert, that can be used to program the actions of a ladybird (Kara) on a bi-dimensional grid, by means of a finite state automaton. Movements can be towards the four directions; it is moreover possible to put tree logs as obstacles in crossing-points. The ladybird can put or get four-leaf clovers (bits), move mushrooms (like marks) and be aware of the objects it could be surrounded. This tool is simple, nevertheless it has the same expressive power of other computational models (it is Turing-complete). Its visual programming is easy to learn; as a consequence children can address not trivial computational problems, without mastering a particular programming language (that represents the next natural step, though). Once translated the documentation and the tool in Italian (from German), Kara has been presented to secondary school teachers, in order to propose it to their pupils and extend with them the set of case studies.

References

1. Chiara Bodei, Angela Giannetti, Maria Rita Laganà. La danza dei bit. In *Difficoltà di apprendimento, 14/1*, 2008 Erickson.
2. Chiara Bodei, Angela Giannetti, Roberto Grossi, and Maria Rita Laganà. Danzando coi bit. Da presentare allo Workshop GRIN@Kangourou: Informatica e Scuola, maggio 2010.
3. Chiara Bodei, Roberto Grossi, Maria Rita Laganà. La danza dei bit: un approccio multidisciplinare per l'apprendimento dell'informatica nella scuola primaria. *Atti del convegno DIDAMATICA 2009 (Informatica per la Didattica)*.
4. Jasmine Del Vecchio, Michele Albano, Marco Righi, Maria Rita Laganà, Marco Maria Massai. Piccoli robot per riflettere insieme giocando. *Atti del convegno DIDAMATICA 2009 (Informatica per la Didattica)*.
5. Michele Freschi. AI-Game: un ambiente didattico per la programmazione. Tesi di Laurea. Dipartimento di Informatica. Università di Pisa. [http://etd.adm.unipi.it/theses/available/etd-06302009-112756/]
6. Maria Rita Laganà, Marco Righi: Romeo and Juliet. An infrared search system. ROBOCOMM 2007: 38.
7. Kara. [http://www.swisseduc.ch/compscience/karatojava/kara/]

Ongoing Research in Wireless Sensor Networks

M. Bonuccelli, S. Chessa, S. Pelagatti

Dipartimento di Informatica, Università di Pisa,
Largo Pontecorvo 3, 56127 Pisa, Italy

1 Introduction

Wireless Sensor Networks (WSN) [1] are a recent technology suitable for unattended monitoring of a wide range of environments, spanning infrastructures such as factories or public buildings, as well as houses or even humans. In a WSN a set of low-power, inexpensive embedded devices (called sensors) spontaneously cooperate to construct a wireless network to support their monitoring activities. In such cases the sensors are active and typically comprise a processor, one or more sensing units (transducers), a radio transceiver and an embedded battery. A special sensor, called sink, acts as a gateway with the external networks, and it makes the sensed data available to external users. Prominent metrics for the evaluation of solutions for WSN are the energy consumption, the sensors' memory occupation, the code footprint, and the packet size.

In other applications the sensors are passive, they are not battery powered but they rely only on the energy induced by the electromagnetic waves emitted by a powered unit (the reader), and they have a limited storage functionality. Typically, passive sensors are implemented as RFID (Radio Frequency IDentification) tags.

Although in some applications active and passive sensors may coexist, in most cases their applicative scenarios are different as active sensors can self organize into a network and offer autonomous, unmanned, environmental monitoring service, while passive sensors do not construct an autonomous network.

2. Ongoing Research on WSN

In the early approaches, WSNs implemented an external storage scheme, for example using *Directed Diffusion* [2], where the user queries the network by injecting interests. Data matching an interest is then drawn along a path to the user, and, during this path, it can be aggregated and united with other data sensed by other sensors. This paradigm has evolved on different directions. One important evolution has been towards the use of query languages modeled on the SQL used in data bases to express more complex queries aimed at the specification of data acquisition, aggregation and filtering [3,4]. Another evolution has been towards the so-called *data centric storage* (DCS) models [5,6] where the sensed (and possibly aggregated) data is stored within the network in order to be retrieved later by the user.

In DCS it is assumed that the sensors deployed in the sensing area are aware of their position (for example they are equipped with GPS), and that each sensed datum d is