Juraj Hromkovič
Richard Královič
Jan Vahrenhold (Eds.)

LNCS 5941

Teaching Fundamental Concepts of Informatics

4th International Conference on Informatics in Secondary Schools - Evolution and Perspectives, ISSEP 2010
Zurich, Switzerland, January 2010, Proceedings

Springer

Lecture Notes in Computer Science 5941

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Juraj Hromkovič
Richard Královič
Jan Vahrenhold (Eds.)

Teaching Fundamental Concepts of Informatics

4th International Conference on Informatics in Secondary Schools - Evolution and Perspectives, ISSEP 2010
Zurich, Switzerland, January 13-15, 2010
Proceedings

Springer

Volume Editors

Juraj Hromkovič
Richard Královič
ETH Zürich
Informationstechnologie und Ausbildung
CAB F 16, F 13.1
Universitätstrasse 6
8092 Zürich, Switzerland
E-mail: {juraj.hromkovic, richard.kralovic}@inf.ethz.ch

Jan Vahrenhold
Technische Universität Dortmund
Foundations of Computer Science and Computer Science Education Group
Chair of Algorithm Engineering
Faculty of Computer Science
Otto-Hahn-Str. 14
44227 Dortmund, Germany
E-mail: jan.vahrenhold@cs.tu-dortmund.de

Library of Congress Control Number: 2009941784

CR Subject Classification (1998): K.3, J.1, K.8, H.5.2, D.1, D.3

LNCS Sublibrary: SL 1 – Theoretical Computer Science and General Issues

ISSN 0302-9743
ISBN-10 3-642-11375-3 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-11375-8 Springer Berlin Heidelberg New York

springer.com

Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India
Printed on acid-free paper SPIN: 12830671 06/3180 5 4 3 2 1 0

Preface

The International Conference on Informatics in Secondary Schools: Evolution and Perspective (ISSEP) is an emerging forum for researchers and practitioners in the area of computer science education with a focus on secondary schools.

The ISSEP series started in 2005 in Klagenfurt, and continued in 2006 in Vilnius, and in 2008 in Toruń. The 4th ISSEP took part in Zurich. This volume presents 4 of the 5 invited talks and 14 regular contributions chosen from 32 submissions to ISSEP 2010.

The ISSEP conference series is devoted to all aspects of computer science teaching. In the preface of the proceedings of ISSEP 2006, Roland Mittermeir wrote: "ISSEP aims at educating 'informatics proper' by showing the beauty of the discipline, hoping to create interest in a later professional career in computing, and it will give answers different from the opinion of those who used to familiarize pupils with the basics of ICT in order to achieve computer literacy for the young generation." This is an important message at this time, when several countries have reduced teaching informatics to educating about current software packages that change from year to year. The goal of ISSEP is to support teaching of the basic concepts and methods of informatics, thereby making it a subject in secondary schools that is comparable in depth and requirements with mathematics or natural sciences. As we tried to present in our book "Algorithmic Adventures. From Knowledge to Magic," we aim at teaching informatics as a challenging scientific discipline, full of puzzles, challenges, magic and surprising discoveries. Additionally, this way of teaching informatics is also a chance to import the concept of engineering to schools, by merging the mathematical analytic way of thinking with the constructive work of engineers in the education of one subject.

To underline informatics as well as informatics didactics as scientific disciplines, ISSEP 2010 had two special tracks. The track "Contributions of Competitions to Informatics Education" was based on the fact that taking part in different kinds of competitions provides a valuable contribution to knowledge acquirement and supports the development of problem-solving skills in a creative way. Organizing a competition includes addressing the following two questions:

- Which kinds of competitions are especially well suited for achieving which goals?
- How should one create and choose tasks and rules for such competitions?
- What are the achievements of the competition participants, in particular in relation to their training process?
- What is the influence of competitions on the educational processes in secondary education?

The starting point to this track was provided by the invited talk "Sustaining Informatics Education by Contests" by Valentina Dagienė.

The second track, “Empirical Research,” pointed out that the community of computer science didactics has to strengthen its effort in empirical research in order to be as serious as the didactics of mathematics and physics are. The main questions posed were:

– What is “good empirical research?”
– Which rules should be followed to produce “good” empirical results?
– Which criteria can be applied to recognize “good” empirical results?
– What are the pitfalls of interpreting empirical results?

To make ISSEP 2010 attractive due to high-quality contributions, we increased the number of invited speakers to five. In addition to Valentina Dagienė (Vilnius), we invited the internationally leading experts David Ginat (Tel Aviv University), David Gries (Cornell University), Allen B. Tucker (Bowdoin College), and Amiram Yehudai (Tel Aviv University) to give talks about different aspects of computer science education.

I would like to express my deepest thanks to all members of the Program Committee for serving and thus contributing to the high standard of the ISSEP series among the conferences devoted to computer science education.

November 2009 Juraj Hromkovič

Conference Organization

Program Chairs

Juraj Hromkovič	ETH Zürich
Jan Vahrenhold	TU Dortmund

Program Committee

Peter Antonitsch	University of Klagenfurt
Owen L. Astrachan	Duke University
Ralph-Johan Back	Abo Akademi University
Harry Buhrman	CWI & University of Amsterdam
Valentina Dagienė	Institute of Mathematics and Informatics, Vilnius
Judith Gal-Ezer	The Open University of Israel
David Ginat	Tel Aviv University
Peter Hubwieser	TU München
Ivan Kalaš	University of Bratislava
Peter Micheuz	University of Klagenfurt
Roland Mittermeir	University of Klagenfurt
Wolfgang Pohl	Bundeswettbewerb Informatik
Ulrik Schroeder	RWTH Aachen
Jarkko Suhonen	University of Joensuu
Maciej M. Sysło	UMK Torun, University of Wroclaw
Tom Verhoeff	TU Eindhoven
Michal Winczer	UK Bratislava

External Reviewers

Brauner, Philipp
Leonhardt, Thiemo

Local Organization

Juraj Hromkovič (Chair)	ETH Zürich
Hans-Joachim Böckenhauer	ETH Zürich
Herbert Bruderer	ETH Zürich
Petra Hieber	ETH Zürich
Blanca Höhn	ETH Zürich
Lucia Keller	ETH Zürich
Dennis Komm	ETH Zürich
Richard Královič	ETH Zürich

Jan Lichtensteiger	ETH Zürich
Ueli Marty	ETH Zürich
Tobias Mömke	ETH Zürich
Giovanni Serafini	ETH Zürich
Andreas Sprock	ETH Zürich
Björn Steffen	ETH Zürich
Monika Steinová	ETH Zürich

Table of Contents

Sustaining Informatics Education by Contests (Invited Talk) 1
Valentina Dagienė

Impasse, Conflict, and Learning of CS Notions (Invited Talk) 13
David Ginat

K-12 Computer Science: Aspirations, Realities, and Challenges (Invited Talk) 22
Allen B. Tucker

Perspective on Computer Science Education (Invited Talk) 35
Amiram Yehudai

Didactics of Introduction to Computer Science in High School 36
Michal Armoni, Tamar Benaya, David Ginat, and Ela Zur

Software Design Course for Leading CS In-Service Teachers 49
Ofra Brandes, Tamar Vilner, and Ela Zur

The Effect of Tangible Artifacts, Gender and Subjective Technical Competence on Teaching Programming to Seventh Graders 61
Philipp Brauner, Thiemo Leonhardt, Martina Ziefle, and Ulrik Schroeder

The Difficulty of Programming Contests Increases 72
Michal Forišek

Didactic Games for Teaching Information Theory 86
Michal Forišek and Monika Steinová

Collaborative Initiatives for Promoting Computer Science in Secondary Schools 100
Irene Glendinning and Margaret Low

Teaching Public-Key Cryptography in School 112
Lucia Keller, Dennis Komm, Giovanni Serafini, Andreas Sprock, and Björn Steffen

Towards a Methodical Approach for an Empirically Proofed Competency Model 124
Johannes Magenheim, Wolfgang Nelles, Thomas Rhode, and Niclas Schaper

Having Fun with Computer Programming and Games: Teacher and Student Experiences 136
Clare McInerney

Showing Core-Concepts of Informatics to Kids and Their Teachers 143
Roland T. Mittermeir, Ernestine Bischof, and Karin Hodnigg

Object-Oriented Modeling of Object-Oriented Concepts: A Case Study in Structuring an Educational Domain 155
Michela Pedroni and Bertrand Meyer

Programming Camps: Letting Children Discover the Computer Science 170
Monika Steinová and Juliana Šišková

Mission to Mars – A Study on Naming and Referring 182
Michael Weigend

Long-Term Development of Software Projects – Students' Self-appreciation and Expectations 194
Cecile Yehezkel and Bruria Haberman

Author Index 207

Sustaining Informatics Education by Contests

Valentina Dagienė

Institute of Mathematics and Informatics
Akademijos str. 4, LT-08663 Vilnius, Lithuania
dagiene@ktl.mii.lt

Abstract. Three decades ago high school computing was highly consistent with academic and professional world. This consistency was destroyed when school curricula began to emphasize information and communication technology skills at the expense of computer science. Recently many countries began to think how to re-establish informatics education in schools and how to attract pupils to choose optional modules related to computer science. Although informatics is not taught as a discipline in many countries, pupils are invited to participate in different contests on informatics organized all over the world. When pupils get interested in programming contests, they are looking for training and gain some informatics education. Contests are exceptionally valuable for motivating and involving pupils in computer science. The current paper discusses the contests and olympiads in informatics arranged internationally and continuously. The main attention is paid to the model of International Olympiad in Informatics and International Contest on Informatics and Computer Fluency (named Bebras in Lithuanian, or Beaver in English).

Keywords: Teaching informatics, computer science education, teaching programming, olympiad in informatics, contest on informatics.

1 What? Why? How? – Questions That Should Be Reconsidered in Informatics Education

In one of the fundamental papers on teaching informatics, Juraj Hromkovic asked: What is informatics? What is computers science? Why teach computer science? What to teach and how to teach it? [1] These are core questions to everybody who has been thinking on bringing informatics to the school level.

Significant changes in society do not begin on one particular day or even in a particular year. Changes come slowly, especially in education. Teachers, policy makers and researchers should work continuously for decades in order to gain significant results on pupils' achievements in informatics.

The education achievements that were obtained in the eighties and the nineties of the 20th century might be explained by the implementation of computers and information technologies (IT) in schools and by forming of their impact to general education. In Europe and world wide, countries were tackling the problem in different ways: richer countries were buying computers on a mass scale and supplying their schools with educational software. They were also arranging training courses for

J. Hromkovič, R. Královič, and J. Vahrenhold (Eds.): ISSEP 2010, LNCS 5941, pp. 1–12, 2010.

teachers. Thus, they were using computers and modern technology wherever they could. The majority of other countries, on the other hand, were trying to develop theoretical well-grounded models of informatics and IT education, compose curricula, syllabi, tutorials, textbooks, and arrange trainings, i.e. at implementation to all students with moderate investment in equipment.

In the beginning of the 1980s, the informatics education, although in a different range, was established in the majority of schools of some countries, e.g. Austria [2, 3], Germany, Lithuania [4], Russia. The German computer scientist Klaus Haefner warned that education should be adopted quickly to avoid the risk of misqualifying people [5]. The human brain would be challenged by the growth of information technology and would be subject to competition of information processing systems. To overcome the crises mentioned in time, Haefner recommended bringing informatics into the classes and developing new curricula with information technology behind.

One of the early Russian pioneers in the field of theoretical and systems programming, a founder of the Siberian School of Computer Science Andrei Ershov, has declared a slogan: "Programming is the second literacy" [6]. It has become a popular metaphor, which has been widely used around the world. Politicians and educators in the industrialized countries, proclaimed "computer literacy' as an essential part of education and they demanded the integration of new technologies into the school curriculum.

Teaching informatics started with programming. Sometimes it was interpreted that machines at that time were miserable and that programming was the exceptional possibility to manage them. However, the goal of teaching programming is problem solving transfer, i.e., users are expected to be able to apply what they have learned to solving problems that they have not been taught [7]. Furthermore, programming is the best way to build a language for instructing (communicating) a machine. According to Hromkovic "We have to teach programming as a skill to describe possibly complex behaviors by a sequence of clear, simple instructions“ [1, p. 33]. Later, Avi Cohen and Bruria Haberman went further and declared computer science as a language of technology [8].

A significant role in designing methodology for teaching programming has been played by the scientists of Lithuania. Already in 1978–1979, a students' education in programming by using postal services was drafted. After accomplishment of certain experiments, the Young Programmer's School by Correspondence was established in 1981 [9]. This is one of the oldest schools for teaching programming and it continues to function nowadays. The activity of the Young Programmer's School in distance learning was one of the first examples concerning informatics and had a strong impact on many phenomena related to informatics' teaching, such as accomplishment of the UNESCO initiated project "Distance learning of informatics (programming)" in 1992–1993 [10] and development of the Contests and Olympiads in Informatics [11].

In the recent years, enrolments in the undergraduate programs of computer science have been dropping. There are many factors that have contributed to the decline in student interest, some of which relate to the lack of understanding the essence of computing at school [12]. These after-effects are very closely connected with what has been done in many western countries: computer science was exchanged for information and communication technology in schools. "… we, as computer scientists, are also responsible for this big misunderstanding…", declared Hromkovic [1, p. 25].

Nowadays more and more countries have been reconsidering the role of informatics in general education, e.g. France is discussing the curricula for teaching informatics in secondary schools, and Slovakia is developing new courses for teacher training in informatics.

Bringing informatics to schools through curriculum in a formal track is quite important, however it is necessary to support the informal ways of introducing pupils to informatics. The most famous informal way to introduce informatics are contests and olympiads on programming [13, 14].

Contests make teaching of programming more attractive for students. Furthermore, computer programming is one of the appropriate and effective ways to develop problem solving skills for computer science learners [15]. During contests students meet their peers from all around the country (or countries), make friends, and wait for the next competition ready to show their abilities which have improved since the last contest.

2 Contests on Programming for General Education

Developing abilities to master modern technologies and skills for solving problems is among the most important capabilities of an educated future citizen of an information society and it can be straight connected with informatics education. Problem solving by means of programming does not lose its importance in a contemporary school equipped with modern information technologies and it will remain as a very important part of understanding the information processing and running computer. Programming, with emphasis on algorithms, remains the core of several worldwide contests, e.g. International Olympiad in Informatics (IOI) and the USA Computing Olympiad (USACO). The USACO holds six internet-based contests each year and has several difficulty divisions [16].

In developing teaching of programming, we recommend considering the attractiveness of instructional methods and consolidation of pupils' motivation. The following aspects should be taken into account:

- For school students, practical activities are much more interesting and attractive than academic studies.
- Elements of contests and competition stimulate the learning process.

More time should be dedicated to the motivation, aims, connection between practice and theoretical concepts, and especially to the internal context of the presented theory.

Programming is an activity composed of several components: comprehension of the problem, choosing algorithm, encoding it, debugging, testing, and optimizing [17]. Since many of the skills required for successful programming are similar to those required for effective problem solving, computer programming and particularly choosing one of several possible solutions and later debugging in a short period of time, provides a fertile field for developing and practicing problem solving skills in an environment that is engaging for young students [18].

When students begin learning basics of programming, they soon try to find a place where they can demonstrate their skills, their projects, share their interests or compare themselves with others. This might explain the reasons why many students, soon after they have started learning programming, choose one of the areas where they are able to demonstrate their work immediately, e.g. creation of web pages, or computer graphics. For some areas, e.g. developing algorithms, it is not easy to find practical demonstration. The most powerful means which endorse students' motivation are competitions or contests.

There, the pupils meet their peers from all over the country and form other countries; they make friendships, wait for the next contest ready to show their abilities which have improved since last contest. In the programming contests, pupils use and develop, at the same time, their problem solving skills. Furthermore, pupils especially gifted can be challenged by problems that cannot be solved by applying learned mechanisms, but that require special talent, mental abilities, and probably extraordinary effort, too.

Pupils like to be involved in competition, they like to compete [19]. In education, it is important to find right place for competition: these can be contests or challenges. In a contest, the main interest is the quality of the individual performance; contestants are confronted with problems, not with each other. Contests are extracurricular activities that allow students to acquire their knowledge and, understanding from the classroom, apply it within a competitive environment. These types of activities provide ways of challenging students in creative and innovative ways.

There have been many academic competitions and contests in computer science throughout allover the world. Most of them are programming contests with focus on algorithmic problem solving. There are several contests covering other scientific areas, most prominent examples are contests in robotics: Robocup Junior and First League. There are mixed contests that cover different areas, for example, the American Computer Science League (ACSL). The contests of the ACSL mostly consist of a short answer test and a programming problem. A short answer test contains five questions from categories like number systems, logic, Lisp, data structures, graph theory, digital electronics and WDTPD (What Does This Program Do). Typically answers are very short. The programming problem is solved by submitting a program source code within 72 hours. Framework of classification on computer science contests for secondary school students is provided by Wolfgang Pohl in [20].

There are two main paradigms for implementing contests: from an international level to the local one (top-down strategy), and vice versa, from local activities to an international promotion (bottom-up strategy). The first paradigm is a challenge to find some suitable international contests, analyze, train students, and join them after intensive work. The second paradigm stresses an opportunity to establish the local contest and attempt to develop it to an international level. The IOI is a contest referred to the first competition paradigm while the Bebras International Contest on Informatics and Computer Fluency [21] belongs to the second paradigm.

2.1 International Olympiads in Informatics

The IOI is one of the five international science Olympiads initiated by UNESCO in 1987. It is an annual international informatics competition for individual contestants from many countries around the world, accompanied by social and cultural programs [22].

These competitions focus on informatics problems of algorithmic nature. In the scope of IOI the concept *Informatics* means a domain that is also known as computer science, computing science and information technology.

Yet, the high-level goal of the IOI is to promote computer science among the youth, and to stimulate their interest in programming and algorithms. The contest brings exceptionally gifted pupils from various countries together and renders them an opportunity to share scientific and cultural experiences. Thus, one of the main objectives in each country is to discover, encourage and train exceptionally talented young people in computer science.

The IOI is managed by the General Assembly, which is a temporary, short-term committee composed of the leaders of all the participating countries and by two long standing committees. The International (Steering) Committee consists of representatives of the past, present, and future IOI's as well as several elected representatives. Its task is to retain the continuity of the IOI by finding future host countries. The second committee is the IOI Scientific Committee, the task of which is to ensure continuity and quality control of the IOI competitions [23].

The IOI is organized in and by one of the participating countries. Each participating country typically sends a delegation of four students accompanied by two leaders. Students are usually selected in the national olympiads in informatics or programming contests. Each of the two competition days lasts for five hours with 3 or 4 tasks to be solved.

The students compete individually and try to maximize their score by solving a set of problems. The IOI contestants are required to express their algorithms in one of the allowed programming languages (currently Pascal and C/C++) and they must engineer their programs to run flawlessly, because marking is based on automated execution [24].

Organized in 1989 in Pravec, Bulgaria, the IOI celebrated 20 years anniversary again in Bulgaria, this time in Plovdiv. The 101 tasks were presented for students during 20 years. Tom Verhoeff, one of the leading persons in developing tasks for the IOI, analyzed the 20-year history of IOI tasks and summarized task type and difficulty level, and classified them according to concepts involved in their problem and solution domain [25]. Difficulty level is determined on the basis of what percentage of contestants were able to 'fully' (a submission should be scored 90 % or more) solve the task. According to Verhoeff, many of the tasks are too difficult to use 'as is' in regular computer science courses for secondary education [25].

The most significant contribution of the IOI to computer science education can be considered olympics movements in many countries and regions. Only 13 countries participated in the first IOI, whereas already 82 countries were involved in the 21st Olympiad (actually 79 countries with participating teams and 3 countries observers). Almost all these countries organize national contests or olympiads in informatics and train pupils and teachers. Some of these contests were implemented following the IOI model (with some adaptation to national peculiarities), although some countries are concentrated on their own infrastructure of contests. Additionally there are regional olympiads in informatics, e.g. African, Asian, Arabic, Balkan, Baltic, Central European; usually they are organized in the same manner as the IOI.