

Sustaining Informatics Education by Contests

Valentina Dagiene

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

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 from 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 all over 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 IOI 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.

2.2 The IOI Conference on Olympiads in Informatics

The IOI community consists of about 80 participating countries. We face mainly the same problems: how many of the countries have national olympiads in informatics? Do they have some other contests on programming? How do we pick our students? How do we train them? The IOI presents an ideal forum for discussing these experiences and associated issues. It was decided to establish conferences during IOIs.

The first IOI conference "Olympiads in Informatics" was held in Zagreb, Croatia in 2007 during the first and second competition days. The 17 selected papers discussed the running of and issues facing several national olympiads: Brazilian [26], Canadian [27], Chinese [28], Croatian [29], German [30], Italian [31], Polish [32], Russian [33], etc.

Next IOI conference was organized in Cairo, Egypt. It concentrated on training and task types, and many of the ideas and experiences were drawn from the national olympiads. Tasks are perennial issue for contests, their most visible aspect and, for many contestants, the primary reason for participation. The IOI community strives for quality, variety and suitability. We endeavour to make tasks interesting, understandable and accessible.

Fourteen research papers were published and discussed during the third IOI conference held in Plovdiv, Bulgaria in 2009. The major part of the papers were focussed on training and task types, and some of the ideas and experiences are drawn from the national olympiads.

Tasks are perennial issue for contests, their most visible aspect and, for many contestants, the primary reason for participation. We strive for quality, variety and suitability. If to the contestants, tasks seem to be the main purpose of an olympiad, from an educator's perspective there is often an equal interest in training the contestants. This is not only a question of how we choose the best, or enable to show their true ability. We seek to enthuse them with a passion for the subject.

The IOI conferences are followed by the journal "Olympiads in Informatics" (http://www.mii.lt/olympiads_in_informatics). It is a refereed scholarly journal that provides an international forum for presenting research and development in teaching and learning informatics through competition. Three already published volumes have been closely connected with the IOI conference. Starting from this year, submissions of papers are flexible and there are no requirements to participate in the conference while paper is accepted.

2.3 Regional and National Olympiads in Informatics

The national olympiads exist in a wider community – of course, it is also true for the international olympiads. In order to ensure better preparation for the IOI and to strengthen regional relations, various regional olympiads are organized. While the national olympiads represent informatics teaching traditions of each country, the regional olympiads are usually a mini model of the IOI, allowing the participants to experience what they will come through in the IOI.

We shortly present one of the regional olympiads, the Baltic Olympiad in Informatics (BOI); more detailed view can be found in [34]. The BOI was established on the initiative of three Baltic countries (Estonia, Latvia, and Lithuania) in 1995, and few years later it was opened to all countries around the Baltic Sea. The main goals concentrate on

providing the participating students with experience of an international olympiad, encouraging communication and exchange of ideas between the developers of national contests in informatics, as well as assisting delegation leaders in selecting participants for the IOI.

The BOI is a short-term (lasting 3-4 days) and inexpensive event. It can be distinguished for cozy and good neighborly atmosphere, which is highly important when motivating students for self-help.

Even though the BOI is a mini-model of the IOI, it has significant differences from the cultural and learning perspectives. The organization of the scientific part of BOI's is based on mutual trust of the participating countries. The leaders of all the participating countries offer problems for the nearest BOI. At first draft task texts are offered, then the ideas are exchanged via e-mail, discussed, some problems rejected, while other problems are suggested to be modified and later are accepted. Most of the problems are translated to the native languages by the leaders before going to the olympiad. This is a unique possibility for country representatives to gain experience in organizing the scientific part of a small international olympiad as well as to raise their qualifications in algorithms.

The BOI is also a form of learning for its participants. On one hand, they come to the event ready to gain some international experience after participating in the domestic contest. On the other hand, they know that their final destination is the IOI, and they try to learn as much as possible in the BOI. The organizers of BOI's try to follow as close as possible the newest IOI trends in problem types, compilers, platforms, and contest systems. Even though all the tasks are of the algorithmic nature, they represent cultural and methodical differences. Since in the BOI much preparatory work has been done in advance, team leaders can discuss the tasks, possible solutions and technical issues and the BOI can be considered as a pre-arranged international way of learning.

National olympiads have more complex infrastructure. Countries have different educational systems but there are as many similarities as differences. Usually there are several rounds, e.g. school, regional or state round. It is also common to use difficulty levels e.g. junior and senior.

3 International Contest on Informatics and Computer Fluency

Olympiads in Informatics focus mainly on developing algorithms and programming. Programming is an important area of computer science but it is not enough. What else could we suggest to pupils at secondary schools so that they would gain more complete view of computer science? Which topics of computer science are "compulsory" to general education?

We agreed that teaching informatics in secondary or even primary schools has to start with programming. The programming course can be followed by the introduction of some basic concepts of data structures, recursions, procedures, and fundamental methods for designing algorithms.

Fromkovic suggested to start teaching fundamentals with automata theory [1, p. 34]. Using simple mathematics, pupils can learn modular design methodology of hardware systems and some important verification concepts.

These and other computer science concepts can be introduced to pupils again by contest. The contest should be the key to present various topics of computer science in an attractive way [35].

The idea of the Bebras Contest on Informatics and Computer Fluency originated in Lithuania in 2003 (the name Bebras – in English “beaver” – is connected with a hard-working, intelligent, goal seeking and lively animal). It took almost a year to create the tasks and to prepare the technology for implementing that: the first contest started in October 2004 [36, 37]. The Bebras contest started in a coordinated way: run of contests at schools, where solutions may be submitted to some central authorities or some local organizers.

The Bebras contest addresses all lower and upper secondary school dividing pupils into three age groups: Benjamin (age 11-14), Junior (age 15-16), and Senior (upper secondary level). Some countries divide Benjamin’s group further in Cadets (age 13-14).

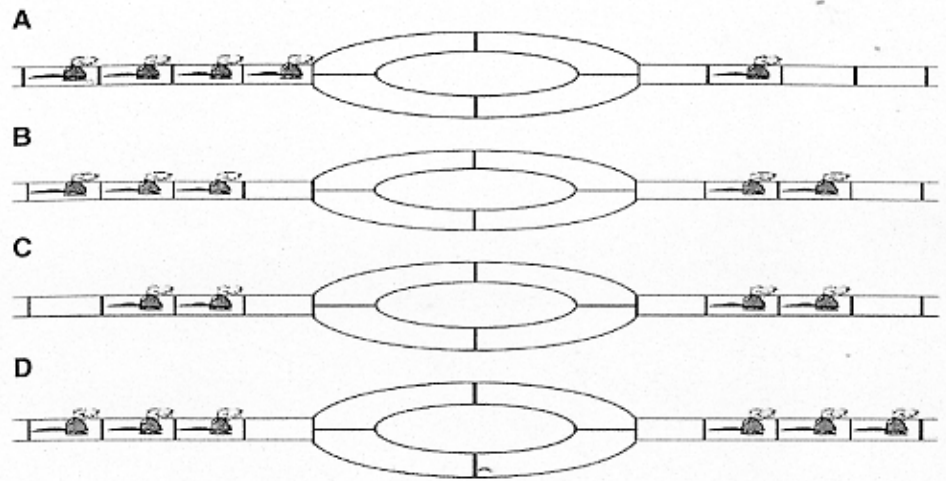
Any contest needs to have first, a challenging set of tasks and second, a grading procedure. Tasks are the most important. The Bebras tasks’ developers are seeking to choose interesting tasks (problems) for motivating students to deal with computer science and to think deeper about technology. Some agreements on tasks development criteria have to be settled [38, 39].

In the past few years, the number of the Bebras participants has been notably growing. In 2007, the Bebras contests took place in seven countries, with about 50 000 participants total. In 2008, more than 90 000 students from 10 countries played the game, world-wide [21]. Estonia had the strongest relative participation with 4 039 contestants, whereas Germany had the highest total number of participants, exactly 53 602. Seven further countries are going to run Bebras contests (Bulgaria, Czech Republic, Egypt, Finland, Italy, Israel, and Macedonia).

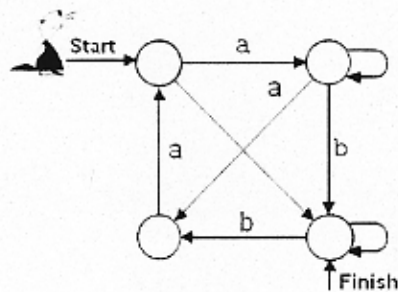
In [39], the six task topics (types) are discussed: (1) information comprehension, (2) Algorithmic thinking, (3) Using computer systems, (4) Structures, patterns and arrangements, (5) Puzzles, (6) Social, ethical, cultural, international, and legal issues. The descriptions of these task types involve also concepts of informatics although this was not the goal of this classification. It gives anyway a rough idea what kinds of problems and what topics of computer science we have in mind for Bebras contest. In short, Bebras tasks can involve concepts of informatics like algorithms and programs: sequential and concurrent; data structures like heaps, stacks and queues; modelling of states, control flow and data flow; human-computer interaction; graphics; etc.

Task example 1. Beaver Creek (Bebras Contest, 2008; type: algorithmic thinking; level: medium, juniors). There are tracks in Beaver Creek. Since beavers do not go backwards, there are some parallel tracks to give the way. Look at the figures. There can be only one beaver in each cell. In which situation is a total traffic jam *unavoidable*?

To solve this task one has to imagine how the beavers walk from one field to the next, where the beavers do their steps, and whether they do one step after each other or walk in parallel! With this task one can learn that concurrent activities may end up in a deadlock where no further move is possible. This task allows an insight into autonomous agents that act in parallel and may run in a so-called deadlock.



Task example 2. Paperchase. (Bebras Contest, 2007; type: structures, patterns, arrangements; level: hard, seniors). Peter writes on a paper the letters he finds on his way following the arrows. Some of the arrows have no letter. Which one of the following sequences of letters *cannot* be written by Peter on his way from *Start* to *Finish*?

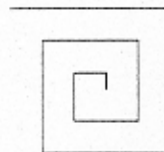


- A abaabba
- B ba
- C abaaaab
- D aab

Task example 3. Drawing spiral. Using Logo, a simple programming language, Anna has drawn a rectangular spiral with the help of the following commands:

- forward 10 – the turtle moves forward drawing a line of 10 steps (dots) long;
- left 90 – the turtle turns left making an angle of 90 degrees.

Which of the following numbers expresses the length of the whole spiral in dots?



- A 550
- B 170
- C 300
- D 250

Since 2004, the Bebras contest runs every autumn, usually at the end of October or at the beginning of November. Before the annual contest, the Bebras workshop is

organized. The main goals of the workshops are to develop a set of tasks for the upcoming contest, to discuss them and to come to an agreement among the countries with different (or without) curricula and traditions of teaching computer science in general education. The workshop for developing tasks is organized every spring.

Two general types of problems have been used in the Bebras contest: interactive tasks and multiple-choice tasks. Creating interesting and attractive tasks, which motivate and introduce pupils to basic concepts of computer science, is very challenging for researchers as well as teachers.

4 Conclusions and Challenges

Informatics, the science of algorithmic processing, representation, storage and transmission of information, is important discipline in the knowledge society and should be introduced in to secondary or even in to primary school. Informatics or Computer Science is a fascinating research area with a big impact on the real world, full of spectacular ideas and great challenges [40].

There is a lack of recognition for the value of informatics in general education and the mistake of over-focusing on computer driving skills. There is a shortage of the informatics teaching methodology for schools. We have very few reasonable didactical approaches for teaching informatics, and one of them could be introducing informatics through contests.

Contests play an important role as a tool of motivation and inspiration. In order to encourage students to learn computer science, we should look for attractive didactic forms. The olympiad in informatics and the Bebras contest on informatics and computer fluency could serve as useful examples.

Olympiad is a globally recognized way to involve pupils into informatics and a very important motive to improve their programming skills. There is a steady infrastructure of international olympiad in informatics. A community of scientists, teachers, and other professionals in education has been formed too. The regional olympiads are organized following the same principles. Similar olympic movements exist in many countries. Various difficulty levels in national olympiads render a possibility for the students with different experience and knowledge of programming to participate in the event. Even the beginners in programming can acquire motivation to participate and to learn. Olympiads are mainly focused on algorithms design and programming. Actually, tasks are rather difficult for many pupils and require continuous efforts.

The Bebras international contest on informatics and computer fluency is established with an idea that it should fit to each pupil regardless of whether she or he is taught informatics at school or not. The main goals of the Bebras contest are to evoke interest in computer science for everyone, to motivate pupils to understand its fundamentals and to be fluent with the technology, e. g. to be able to communicate with a machine. The contest should help children to get interested in informatics and to stimulate thinking about contributions of informatics to science at the very beginning of school.

Let's take olympiads and contests in informatics as a serious didactical approach of computer science education and support them with learning material, tools, teacher training courses, etc. Olympic and contest communities are open for proposals and

ideas for collaboration and future developments. Involving pupils in recognition of informatics as a science discipline should be our target, and we should try to achieve it together. Well-organized contests with interesting, playful, exciting problems, and attractive awards will involve pupils into the essence of the computer science world and will help to understand the realities, possibilities, and failings of the technology.

References


1. Hromkovic, J.: Contributing to General Education by Teaching Informatics. In: Mittermeir, R.T. (ed.) ISSEP 2006. LNCS, vol. 4226, pp. 25–37. Springer, Heidelberg (2006)
2. Micheuz, P.: 20 Years of Computers and Informatics in Austria's Secondary Academic Schools. In: Mittermeir, R.T. (ed.) ISSEP 2005. LNCS, vol. 3422, pp. 20–31. Springer, Heidelberg (2005)
3. Reiter, A.: Incorporation of Informatics in Austrian Education: The Project "Computer-Education-Society" in the School Year 1984/85. In: Mittermeir, R.T. (ed.) ISSEP 2005. LNCS, vol. 3422, pp. 4–19. Springer, Heidelberg (2005)
4. Dagienė, V.: The Model of Teaching Informatics in Lithuanian Comprehensive Schools. *Journal of Research on Computing in Education* 35(2), 176–185 (2002–2003)
5. Haefner, K.: Die neue Bildungskrise. Herausforderung der Informationstechnik und Bildung und Ausbildung, Basel (1982)
6. Ershov, A.P.: Programming is a second literacy. *Pocit e umela intel* 1(6), 457–471 (1982)
7. Mayer, R.E.: Teaching for Transfer of Problem-solving Skills to Computer Programming. In: De Corte, E., Linn, M.C., Mandl, H., Verschaffel, L. (eds.) *Computer-based Learning Environments and Problem Solving*. NATO ASI Series, pp. 193–206. Springer, Heidelberg (1991)
8. Cohen, A., Haberman, B.: Computer Science: A Language of Technology. *ACM SIGCSE Bulletin* 39(4), 65–69 (2007)
9. Dagys, V., Dagienė, V., Grigas, G.: Teaching Algorithms and Programming by Distance: Quarter Century's Activity in Lithuania. In: *Proc. of the 2nd Int. Conference on Informatics in Secondary Schools: Evolution and Perspectives*, Vilnius, November 7–11, pp. 402–412 (2006)
10. Grigas, G.: Distance teaching of informatics: motivations, means and alternatives. *Journal of Research on Computing in Education* 27(1), 19–28 (1994)
11. Dagienė, V.: *The Road of Informatics*. Vilnius, TEV (2006)
12. Clark, M.A.C., Boyle, R.D.: Computer Science in English High Schools: We Lost the S, Now the C Is Going. In: Mittermeir, R.T. (ed.) ISSEP 2006. LNCS, vol. 4226, pp. 83–93. Springer, Heidelberg (2006)
13. Bowring, J.F.: A New Paradigm for Programming Competitions. In: *Proc. of the 39th SIGCSE technical symposium on Computer science education*, Portland, OR, USA, March 12–15, pp. 87–92 (2008)
14. Kears, I.B., Hardnett, C.R.: Computer Science Olympiad: Exploring Computer Science through Competition. *ACM SIGCSE Bulletin* 40(1), 92–96 (2008)
15. Dagienė, V.: Programming-based solution of problems in informatics curricula. In: *Communications and Networking in Education: Learning in a Networked Society*, IFIP WG 3.1 and 3.5, Finland, pp. 88–94 (1999)
16. Kolstad, R., Piele, D.: USA Computing Olympiad (USACO). *Olympiads in Informatics* 1, 105–111 (2007)
17. Dagienė, V., Skapiene, J.: Learning by competitions: Olympiads in Informatics as a tool for training high grade skills in programming. In: Boyle, T., Oriogun, P., Pakstas, A. (eds.) *2nd Int. Conf. on Information Technology: Research and Education*, London, pp. 79–83 (2004)

18. Casey, P.J.: Computer programming: A medium for teaching problem solving. In: *Computers in the Schools*, vol. XIII, pp. 41–51. The Haworth Press, Inc., New York (1997)
19. Verhoeff, T.: The Role of Competitions in Education. In: *Future World: Educating for the 21st Century: a conference and exhibition at IOI 1997* (1997) (retrieved June 2007). <http://olympiads.win.tue.nl/ioi/ioi97/ffutwrlld/competit.pdf>
20. Pohl, W.: Computer Science Contests for Secondary School Students: Approaches to Classification. *Informatics in Education* 5(1), 125–132 (2006)
21. Bebras International Contest on Informatics and Computer Fluency home page. <http://www.bebbras.org>
22. IOI home page. <http://www.ioinformatics.org>
23. Verhoeff, T.: The IOI is (not) a science olympiad. *Informatics in Education* 5(1), 147–159 (2006)
24. Verhoeff, T., Horvath, G., Diks, K., Cormack, G.: A proposal for an IOI Syllabus. *Teaching Mathematics and Computer Science IV*(1), 193–216 (2006)
25. Verhoeff, T.: 20 Years of IOI Competition Tasks. *Olympiads in Informatics* 3, 149–166 (2009)
26. Anido, R.O., Menderico, R.M.: Brazilian Olympiad in Informatics. *Olympiads in Informatics* 1, 5–14 (2007)
27. Kemkes, G., Cormack, G., Munro, I., Vasiga, T.: New task type at the Canadian Computing Competition. *Olympiads in Informatics* 1, 79–80 (2007)
28. Wang, H., Yin, B., Li, W.: Development and Exploration of Chinese National Olympiad in Informatics (CNOI). *Olympiads in Informatics* 1, 165–174 (2007)
29. Brodanac, P.: Regular Competitions in Croatia. *Olympiads in Informatics* 1, 15–23 (2007)
30. Pohl, W.: Computer Science Contests in Germany. *Olympiads in Informatics* 1, 141–148 (2007)
31. Casadei, G., Fadini, B., Vita, M.G.: Italian Olympiads in Informatics. *Olympiads in Informatics* 1, 24–30 (2007)
32. Diks, K., Kubica, M., Stencel, K.: Polish Olympiads in Informatics – 14 Years Experience. *Olympiads in Informatics* 1, 50–56 (2007)
33. Kiryukhin, V.M.: The modern contents of the Russian National. *Olympiads in Informatics* 1, 90–104 (2007)
34. Poranen, T., et al.: Baltic Olympiads in Informatics: Challenges for Training Together. *Olympiads in Informatics* 3, 112–131 (2009)
35. Dagienė, V.: Information Technology Contests – Introduction to Computer Science in an Attractive Way. *Informatics in Education* 5(1), 37–46 (2006)
36. Dagienė, V.: Competition in Information Technology: an Informal Learning. In: *EuroLogo 2005: the 10th European Logo Conference, Warsaw, Poland, August 28-31*, vol. 31, pp. 228–234 (2005)
37. Dagienė, V.: The BEBRAS Contest on Informatics and Computer Literacy – Students’ Drive to Science Education. In: *Joint Open and Working IFIP Conf. Kuala Lumpur*, pp. 214–223 (2008)
38. Opmanis, M., Dagienė, V., Truu, A.: Task Types at “Beaver” Contests Standards. In: *Proc. of the 2nd Int. Conf. on Informatics in Secondary Schools: Evolution and Perspectives*, Vilnius, pp. 509–519 (2006)
39. Dagienė, V., Futschek, G.: Bebras International Contest on Informatics and Computer Literacy: Criteria for Good Tasks. In: *Mittermeir, R.T., Syslo, M.M. (eds.) ISSEP 2008*. LNCS, vol. 5090, pp. 19–30. Springer, Heidelberg (2008)
40. Hromkovic, J.: *Algorithmic Adventures: From Knowledge to Magic*. Springer, Heidelberg (2009)

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