

Programming Camps: Letting Children Discover the Computer Science ^{*}

Monika Steinová¹ and Juliana Šišková²

¹ Department of Computer Science, ETH Zurich, Switzerland,
`monika.steinova@inf.ethz.ch`

² Department of Informatics Education, Comenius University, Bratislava, Slovakia,
`juliana.siskova@fmph.uniba.sk`

Abstract. Daily and summer camps for children are certainly well known to every adult. They provide many activities that encourage interests, create opportunities to start new friendships and, last but not least, help parents with the child care when they are unable to take holidays. However, camps designed as a support for contests in the form as we are presenting them here are unique, established in former Czechoslovakia by Vít and Milan Hejný in the late 70s. In this paper, we provide a basic outline of the structure of these programming camps for high school students, we describe the daily program, and introduce the games and activities used to attract and educate participants. Our hope is to provide a comprehensive overview that can serve as a guideline for organizing similar camps in other countries, and in this way to draw the attention of the talented students, to help them to prepare for the contests like the International Olympiad in Informatics (IOI) and to encourage them to choose a career in computing.

1 Introduction

In Computer Science (CS), just as in any other area, training future generations of prospective experts is a vital task. The sooner one gets acquainted with CS, the better results he can achieve in pursuing it. That is why introducing children to CS is a worldwide trend.

Different countries have their own strategies to encourage the students in exploring CS. A method used worldwide to encourage them is by contests [7]. The best known and most prestigious is the International Olympiad in Informatics [1] (and the regional contests it has inspired). At present the contest Bebras [4] is also gaining in popularity.

In Slovakia, we stimulate the interest for CS and train young people for programming contests by organizing correspondence based contest on the local level and through the programming camps. For various activities focused on preparing the students in Slovakia, see Forišek, Winczer [5]. In this paper, we present

^{*} This work was partially supported by FILEP grant 351 of ETH Zurich.

our experience in preparing the camps, not only by discussing the structure and activities of the camp, but also the reasons why and the ways how it helps to teach children about the CS, develop their social skills and hunger for knowledge.

In Section 2, we outline the basic information about our scientific competition camp; in Section 3, we present the schedule of the camp. Section 4 and Section 5 are devoted to the camp's educational and social activities, respectively. Some statistics of popularity of the camp games and activities with participants are presented in Section 6. The paper is concluded in Section 7.

2 Overview

Scientific contest camps for talented youth have a long tradition and respected standing in Slovakia. The most elementary camps are oriented towards primary school students from the age of 12 years and are focused mainly on mathematics. Their objective is to popularize the field and to provide impulses beyond the scope of the school curricula.

The Correspondence Seminar in Programming (KSP) is a correspondence contest concerned with algorithmic programming with more than 25 years of tradition, in which secondary school students regularly receive letters with contest tasks. The target group of the contest are young people up to 19 years, i.e. the age when young people enter universities in Slovakia. It covers mainly the field of algorithm design through problem solving. To solve the contest problems, the participants must show that they are capable of implementation, clear explanation of the underlying ideas and reasoning about the complexity of the solution. The organizers of KSP are usually former participants that became students of Computer Science at Comenius University in Bratislava and they continue to run KSP on the voluntary basis. This is the case of both authors of this article as well.

The competition consists of four correspondence series per year with 10 tasks each. Participants send the written solutions that are returned back corrected together with sample solutions and the ranking list. The contest is handled by the postal service (recently, emails became acceptable, too). Therefore, the camps offer the only opportunity for the participants as well as the organizers to get to know each other personally.

The reward for the best 32 participants of KSP is the 6-day long programming camp organized twice per year (after a period of two correspondence series) by 10 organizers from KSP. The age of the participants varies and is usually between 15 and 19 years. A significant number of participants comes from schools without additional lectures in programming and/or mathematics. Female students are encouraged to participate in KSP. The best 4 female participants are invited to the camp, regardless of their overall ranking.

Camp location. The camp is located in a camp resort with catering in a remote area, usually without internet access. The location is selected with a purpose of relieving the participants from the influence of modern technologies and letting

them instead spend their time socializing. It also allows the organizers to prepare activities which the participants can hardly experience elsewhere.

The camp team competition. At the beginning of the camp, participants are divided into 4 teams that are then competing with each other during the entire camp. The competition is more or less symbolical and the final award is only a cake for each team. The only individual competition is a series of mental exercise (see Section 4.5 for details) which is evaluated separately at the end of the camp.

The underlying benefit of the team structure is that the team members are learning from each other while answering to the impulses in the games and activities. This corresponds with Vygotsky’s principles of how children learn being tutored by older children [8]. Moreover, the team structure allows the introverts to feel safe in the team whereas more extrovert people gain the space for self-realization.

Storyline. The uniqueness of each camp is achieved by creating its storyline with the usual themes include fantasy (wizards, spells), science fiction (space-ships and aliens), history (knights, prehistorical), literary tales (Harry Potter, Musketeers) or modern adventure (journalists, spies, detectives). During the camp week, the storyline gradually uncovers through a number of theme games, activities and by other means such as regular role-playing within the scope of the story, daily “printed” camp newspapers, appearance of the messages on the camp pin-board or appearance of special items in the camp area. The storyline develops until the last day, when, in a concluding main game, a final common objective has to be achieved (save the Earth/princess, return from the deserted island, etc.). The storyline gives a very special tint to every camp and makes the participants return home with positive memories.

3 The weekly schedule of the camp

The overview of the weekly schedule of the camp is shown in Figure 1. For detailed description of particular activities see Sections 4 and 5.

The camp start-up games are usually placed in the schedule around the time of the first dinner. During the first part of these games, participants and organizers are introduced and a game is played to help them remember the names of each other. The second part utilizes the distribution into the teams. The teams need to be well-balanced with respect to their physical prowess and CS skills of their members.

The “Grand Prix”, i.e. the main concluding game, is a climactic point of the camp storyline. Usually, it is divided into several parts that resemble the games from the social program section (see Section 5 for details).

The overall evaluation of the team competition, series of mental exercise and also the awarding of the best KSP-solvers takes place in a final closing ceremony.

The schedule of the camp might be re-arranged when a special game or activity (e.g., a whole-day hiking trip) from Section 5 is planned. Possible sports games between lectures, the period of relaxation after the lunch and the short breaks between the timeslots in the schedule are not depicted in the Figure 1 for clarity of presentation and are variable in accordance with the needs of the participants.

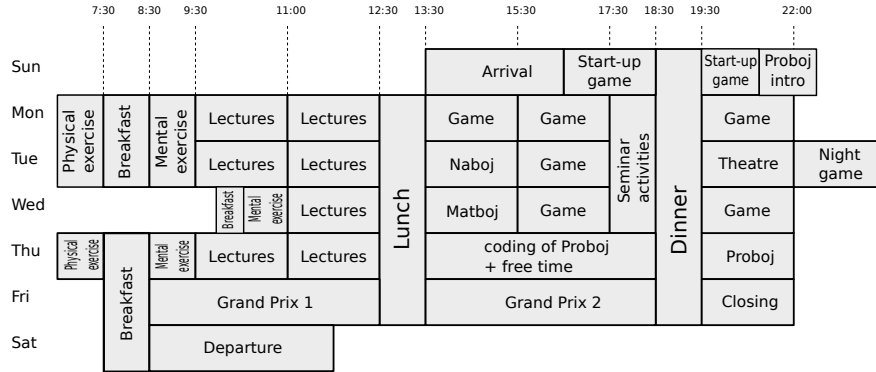


Fig. 1. The week overview of a camp schedule

4 Educational program

In this chapter, we provide an overview of the educational activities of the camp. Besides lectures and seminars, as described in Section 4.1, these activities consist of various contests which are described in more detail in Sections 4.2 to 4.5.

4.1 Lectures and seminar activities

Lectures and seminar activities are the essential part of CS-teaching in the camp. They are arranged by the organizers of the camp who are all university students of CS. The fact that the lecturer was once a participant himself gives him/her a better perspective for detection and solving of the problems connected with the comprehension of the topic by the participants. The lecturing competence of the less-experienced lecturers is mitigated by careful selection of suitable topics and by the supervision of the older students from the KSP organization board. No two camps are organized by the same set of people which brings uniqueness and diversity into the content as well as into the overall realization.

The length of a lecture ranges from 45 minutes up to one and a half hour. There are usually two parallel lectures – one intended for the beginners and

one for the advanced participants – and they usually take place in the morning. The content of the lecture is prepared so that the people at different levels of knowledge are able to follow it. The lecturer is ready to provide extra information if the lecture is progressing better than expected. The topics of the lectures vary from CS on the theoretical level to the presentation of the newest trends, multimedia and technologies.

In contrast to the lectures, the seminar activity is shorter, working with a small number of people only. Out of five (or more) parallel seminars at various levels of difficulty, every participant selects and attends one seminar that is a series of 3 or 4 one-hour meetings per camp. Since the number of participants of a seminar is usually limited to eight people, the presentation of the topic is very individual and allows for the use of different forms of presentation methods including short internal contests, games, etc. The topics discussed during these sessions are more oriented towards CS and algorithm design.

Some beginner lecture topics that will prove to be successful are: *simple complexity* (big O-notation), *data structures* (array, linked list, stack, queue), *basic graphs* (definition, representation, depth- and breadth-first search), *sorting* (sorting algorithms like quick sort and merge sort), *dynamic programming* (introduction with some examples), *greedy technique* (introduction with some examples) and *divide and conquer technique* (introduction with some examples).

Other possible topics discussed in the lectures/seminars include *computational geometry* (convex hulls, localization of points in the plane, sweeping technique), *complexity theory* (amortized complexity, Master theorem for solving recurrence equations), *string algorithms* (Knuth-Morris-Pratt and pattern matching, suffix trees), *cryptology* (protocols, digital signatures, time stamps, RSA), *coding theory and compression* (Huffman, Fano, Shannon codes, compression in bzip2), *formal languages* (finite automata, parsing, nondeterminism, Turing machines, NP-completeness), *graphs* (coloring, Euler tour, spanning trees, bridges and articulations, shortest paths, topological sort), *data structures* (heap, binary search trees, interval trees, trie, hashing), *mathematics* (Euclid's algorithm, game theory), *popular software* (TeX, Linux, various programming languages, viruses), *computer architecture* (file systems, operating systems, cache), *ACM training* and many more.

4.2 Proboj – The programming competition

Proboj³ is a team programming contest spanning almost the whole period of the camp. It is based on a simple multiplayer computer game that several organizers develop for the camp. During the camp, each team develops a player program. In the end, the final versions of these programs are launched simultaneously, competing against each other. The organizers prepare the Proboj's environment, i.e. the interface via which players obtain information about the state of the game, whereas teams have to develop a kind of artificial intelligence that is going to play on their behalf. Proboj is graphically very attractive and the interface

³ abbreviation for “the programming fight” in Slovak

used to control a player is simple – usually a single procedure is executed as a move of the player. During the week, participants often have a textual version of the game only, but then the tournament is presented with all the glamor – intro with titles of producers, supporting music and then several games in nicely rendered environment.

Since there are many different ways in which members of the team can contribute, the contest is very appealing to the participants. Usually, several people analyze strategies and possible reactions to the opponent moves. Then, a few selected members of the team transform the ideas into the code. Here, the participants can experience teamwork on a project that includes many different aspects of development.

The game rules are introduced to the participants on the day of their arrival allowing them to discuss Proboj during their free time. The afternoon of the day before the last day is devoted to the development of the player programs and then, in the evening, the tournament of the final versions of players is staged.

In the previous years, several different types of Proboj were used. The most remarkable type of the Proboj programs is the turn-based strategy, where, given a map, the player has to collect objects by employing units within the game. Other types include card games (Hearts, Poker), board games (Clue, Risk, Stratego) and real-time games (car racing, tennis, ice-hockey).

4.3 Naboj – the mathematical team contest

The problems are usually derived from various areas of mathematics and CS. This type of contest stimulates competitiveness and allows the participants to match their skills, that is why it is very popular. It is not physically demanding and therefore suitable as the afternoon program.

The competition usually lasts one hour, and a further half of an hour is reserved for the discussion of the solutions. Out of about 30 tasks of increasing difficulty the team always has an access to 5 tasks at one time. Each time the team solves a problem, it gets a new one. Usually the rules of the game allow the team to drop a task and get another instead. The ratio between the mathematical and the CS tasks is approximately 2:1.

Example 1. Some problems used in Nabojis include:

1. Find the smallest positive number x , such that x modulo 47 = 5 and x modulo 42 = 14.
2. You are given a program with task description. Replace one character to make the program run correctly.
3. Prove: $\forall n \in \mathbb{N} : \frac{1}{n+1} + \frac{1}{n+2} + \dots + \frac{1}{2n} > \frac{13}{24}$
4. Decide on the validity of the following statements:
 - (a) The complement of a disconnected graph is a connected graph.
 - (b) The complement of a connected graph is a disconnected graph.
5. Prove or disprove: A graph is connected if and only if its edge-graph is connected.

Solving as many tasks as possible does not have to be the main goal of Naboj. The competition is often covered by a guise of a game where solving a task is reflected into one or several moves in a strategic game, revealing nodes while searching a graph, gaining clues to a Zebra puzzle⁴ [3] or a Smullyan-like logic problem [6]. The covering game has to be well-balanced so that the more tasks a team solves the more points it gains. However, it should be designed in such a way that devising a good strategy can affect the overall results slightly.

The following competencies are developed during Naboj.

Brain-tingling. Mathematical and CS problems introduce the participants to the enjoyment of learning and discovering own potential and the thrill of finding the answer.

Learning by observation. Participants have different skill levels in solving the problems and different interests in topics. Therefore, the team cooperation causes participants to observe the ways of solving the tasks as done by other team members. The discussion and the sample solutions provided after the contest are another example of learning by observation.

Task management. Given 5 tasks at one moment, the team has to manage its tasks and distribute them among its eight members according to their abilities.

Skill estimation. Due to the number of tasks and the covering game, most of the tasks are being solved by one member of the team only. Therefore, the right estimation of skills is important and overestimating is expensive – it costs time and also points.

Development of strategies. Besides the need of a strategy within the covering game, the solving of the tasks opens the room for tactics. Participants are aware that the difficulty of the tasks present is increasing and that once a task is given up it cannot be solved later.

4.4 Matboj – the game tournament

Matboj⁵ is a tournament in a given combinatorial game (e.g., variation of the NIM) or board game. The game used for the tournament is usually a two-player game, preferably with complete information; it should take only a few minutes to play, and the optimal strategy needs to be complex enough so that the players cannot detect it easily.

The tournament proceeds as follows: After the explanation of the rules of a particular game, each team has up to 30 minutes to devise a strategy and play several sample games. Then the four teams play the all-play-all tournament, i.e. the members of the two opposing teams form pairs that are playing two rounds of the game – each player starting once. The winning player receives 2 points

⁴ The Zebra puzzle consists of N people, several different types of sets of N items and a number of statements about the relations of these people and these items. The task is to assign one item of each type to every person so that the statements given hold and so that none of two people have the same item of one type.

⁵ abbreviation for “math fight” in Slovak

for his/her team, in the case of a draw, both players receive 1 point. After the tournament, all points gained by a team are summed-up and the final ranking of the teams is determined.

4.5 Mental exercise

The program of the standard camp day begins with a session of mental exercise after the breakfast. This is a regular activity where participants warm-up their brains before the lectures by solving simple algorithmic tasks individually, using pen and paper only. The series of mental exercise is the only individual competition of the camp. The best solvers of the whole week are awarded at the end of the camp.

Each session of the mental exercise consists of two algorithmic tasks. The tasks are a lighter version of the KSP tasks. They are simplified to the utmost level and given without a story just as pure formulations of problems. The participants are asked to solve both tasks in 30 minutes. Their solution should contain their core idea, pseudocode and discussion of the complexity. Mental exercise may contain tasks where knowledge gained in the beginner lectures is applied.

Example 2. Some tasks used in the mental exercise.

- Given a two-dimensional array of 0s and 1s, write a program that finds the largest square containing only 1s.
- Given two sorted arrays of different integers, write a program that computes the median⁶ of the elements of both arrays.
- Given an array of integers, find the longest subsequence with the largest arithmetic average.
- Write a program that calculates the smallest number of trails necessary to draw a given graph. Each edge of the graph must belong to exactly one of the trails. (A trail is a sequence of distinct edges with each connected to the preceding one.)

5 Social activities

Since the stay in the camp is a reward for the participants of KSP, the camp program cannot consist solely of teaching and learning about new topics. Instead of making the camp schedule school-like, the educational program is interlaid with several other activities, the most significant representatives of which are mentioned below. Except for the academy, the hiking trip and the fighting games, all other games are carried out on the team basis.

The games and activities are intended to be an enjoyable part of the camp for all the participants – by simple being a game and also by the addition of side games that can affect the course of the large game. These side small games are diverse so that each member of the team can select the activity that suits him/her best. While playing these games, participants are confronted with the competencies pointed out in the text.

⁶ the middle element out of sorted sequence

Academy. Academy is an activity requiring more time, consisting of several series of lectures that imitate the university environment. The organizers of the camp prepare several mock-lectures (e.g., "How to open bananas") which they present in 10 minutes and then examine students in the following 5 minutes. Participants attend the lectures and are graded.

Whenever the participants passed an exam, they are allowed to lecture on the topic (but they are not allowed to stage exams and therefore to train new lecturers). In the end of the academy, all participants are examined from all lectures they attended. In the elaborated closing ceremony participants receive academic or scientific titles. This whole activity develops the *learning* and *presentation skills* and *trains the concentration*.

Board games. The central component of the game is the board, where one or several team members move the team tokens. The moves on the board or the commodities needed for boosting of the tokens are earned by physical/mental side games. The team has to *plan, find strategies* and *adjust to the individual talents* of its members in order to maximize the gain.

Fighting game. The fighting game is a voluntary game that might be scheduled at the very end of the camp program. It takes place outdoors and its main idea is a battle between the participants and the organizers. It is physically demanding but not as harsh as its name might suggest. A good example of such a game is the so-called sock game. Here the objective is to strip the players from the other group of their socks. The last player wearing his/hers socks wins the game for his/her group. The aim of the game is to *relieve tension* and to *dissolve the distinction between the organizers and participants*.

Graph game. The central goal in this game is to be the first to acquire a certain item. This is not a straightforward process, but rather there are several criteria that need to be met first. Some of the criteria might be dependent on other sub-criteria. The criteria dependencies form a graph⁷, which has to be identified by the participants. The standard strategy that the teams come up with is to draw the directed graph and divide its subtrees among the team members. The game promotes abilities such as *cooperation, resource management* and *communication*.

Hiking trip. Depending on the setting of the camp, a hiking trip can be scheduled in the program. During the hike, participants and organizers are getting to know each other, they chat in friendly atmosphere about topics not necessarily connected with computing and thus *strengthening their relations*.

Investigating game. The game is based on solving a Zebra-like puzzle [3] or Smullyan-like logic problem [6]. The outcome of such solution is a password or a meeting place that is needed to proceed in the camp storyline. In addition to the puzzle or the logic problem, a list of extra, quite often funny activities is given. By completing these activities, teams get puzzle clues. The game supports the forming of the *logical thinking* of the participants. Team members have to *make a tactic* when to make an attempt at the puz-

⁷ usually a directed acyclic graph

zle/problem directly and when to use the time and the team-members to obtain a clue.

Night game. This game is not announced in advance, which throws in the element of surprise and thrill. The participants are awakened in the middle of the night, usually because an unexpected event occurred in the camp storyline. They might be asked to rescue a storyline character, follow clues or a map or play an outdoor game to resolve the unfavorable situation. The important rule is not to force the participants to participate. Apart from the thrill of the game, this activity also teaches the participants to *work under pressure and in unforeseen situations*.

Quiz. A typical quiz team game on various topics, which have mostly nothing to do with CS or mathematics. This is an example of purely relaxing activity, which *relieves tension* and allows for *socialization*.

Role playing game. In the role-playing game [2], the participants impersonate roles in a fictional environment in order to achieve certain goals. Every team plays a separate game session with an organizer that is the game master, i.e. he directs the storyline, provides reactions to the player actions and accounts for all the non-player characters in the game. Participants often try to challenge the game master by creating unexpected situations in which game master needs to improvise to a great length. The game is usually scheduled after the dinner since it is a rest indoor program. Role-playing trains the *communication skills*, *boosts team working* and *encourages the introvert team members* to present their ideas.

Theatre. Theatre is a an evening activity in which the teams prepare a short performance. The performance is often restricted by defined title, topic, artistic style, or it must contain certain phrases or sentences. The aim of the theater is to *relax* and *enjoy the atmosphere through creative brain-storming*. It is therefore rarely counted into the team ranking.

Trading game. The game is usually played outdoors with several outposts approximately 100 to 200 m apart. The goal of the game is to accumulate the wealth which is done by trading different (usually very funny) commodities. During the game the trading outposts change prizes of commodities regularly. The standard strategy that is sooner or later employed by the teams is based on acquiring the commodity prizes at the moment, finding out the formulas to determine how the income can be maximized and employ these until the prizes undergo change. This trains the *strategical thinking* and *time management*.

6 Popularity of activities

The most important concern during the camp is that the participants associate the fun with CS and learning. To know whether the participants liked the games prepared, we ask them to fill in a questionnaire. This also helps us to prepare subsequent camps. In Figure 2 we present the questionnaire data for the last nine years of camps that we were able to evaluate.

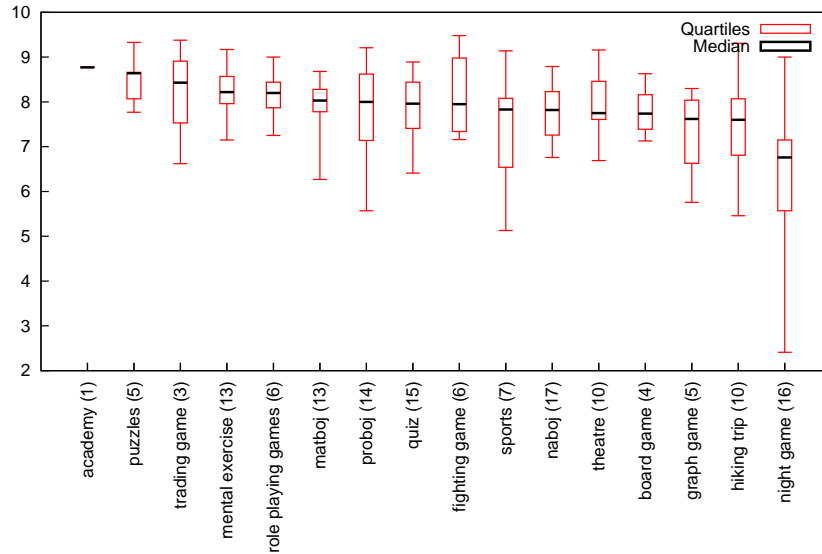


Fig. 2. Popularity of activities in camps between years 1999 and 2008.

The participants award the activities 0 to 10 points, 10 being the highest mark. For each game in a camp, we use the average of the rating in the visualized graph⁸. The figure for each activity contains the median, quartiles, minimum and maximum of the average, taken from all 18 camps. The number in the brackets represents the number of camps in which the activity was graded.

The games based on physical activity obtained generally lower grades. In particular sports and the hiking trip might be rather unpopular due to their nature of outside exhaustive activities. This would not be surprising since the participants often tend to staying inside doing non-physical activities. Another game of this type is the night game. The game often obtains very mixed grading. This is caused for most of the part by the nature of the game, which many participants might not find attractive. On the other hand, the fighting game is more popular. This is probably due to it being rather short and due to the opportunity to challenge the organizers.

Activities that are related to mental aspects seems to gain a bit higher marks. Their popularity is likely a result of the participants being chosen on the criterion of good ranking in a mental activity of related sort, i.e. KSP.

Academy was arranged only once since its realization is time-consuming, but was received very positively. This is not surprising, since an imitation of academic environment is interesting for the prospective students. Moreover, the game is

⁸ The median value, that would be more adequate, was not preserved from all the camps.

not physically or mentally demanding and participants can relax while enjoying (or paying no attention to) the lectures.

Since we did not record all data for board games, trading games, graph games, academy and puzzles, these are mentioned in the figure only for the informative purpose.

7 Conclusion

This paper provides a general overview of how a successful CS camp for young people can be prepared. While offering the opportunity for the young people interested in CS to meet and encouraging their learning, the camps also build a number of valuable competences. In the first place, there is the specialized CS knowledge, which forms the core of the educational part of the camp program. Other competences that are trained within the camp program involve various team-building, team-work and social skills, such as time and task management and cooperation between the teams. Last but not least, there are other elements of social nature that the camp program incorporates, e.g., creativity, sports, communication. If well-balanced, the combination of the impulses provided through the camps can successfully help to insure CS community growth.

Acknowledgment

We would like to thank our friends for valuable comments in preparation of this paper. We are grateful to anonymous reviewers for their feedback. We are also greatly indebted to all the generations of organizers that contributed to KSP and camps and who helped to make them what they are today.

References

1. International Olympiad in Informatics. <http://ioinformatics.org/>.
2. Role playing game. *Wikipedia*®. http://en.wikipedia.org/wiki/Role-playing_game.
3. Zebra puzzle. *Wikipedia*®. http://en.wikipedia.org/wiki/Zebra_puzzle.
4. V. Dagiene. The bebras contest on informatics and computer literacy – students’ drive to science education. In *Joint Open and Working IFIP Conference. ICT and Learning for the Net Generation*, pages 214–223. Kuala Lumpur, 2008.
5. M. Forišek and M. Winczer. Non-formal Activities as Scaffolding to Informatics Achievement. In V. Dagiene and R. Mittermeir, editors, *Information Technologies at School*, pages 529–534, 2006.
6. R. M. Smullyan. *What is the name of this book?* Prentice Hall, 1978. ISBN 0-13-955088-7.
7. T. Verhoeff. The role of competitions in education. 1997. *Future World: Educating for the 21st Century*.
8. L. S. Vygotsky. *Mind in society: The development of psychological processes*. Harvard University Press, 1978.