

Bridging the Gap Between Bebras and Olympiad: Experiences from the Netherlands

Willem van der VEGT

Dutch Olympiad in Informatics

Windesheim University for Applied Sciences

PO Box 10090, 8000 GB Zwolle, The Netherlands

e-mail: w.van.der.vegt@windesheim.nl

Abstract. While the number of contestants in the Dutch Olympiad in Informatics was declining, the number of participants in the Bebras contest grew rapidly. In order to reach these Bebras participants for joining the Olympiad, several steps were taken. We analyzed the differences between the contests. We offered Bebras contestants an introductory course in programming. And we changed the contest format of the first round of the Olympiad, introducing two new types of tasks. As a result, the number of contestants increased and girls returned to the Olympiad.

Key words: Bebras contest, Olympiad in Informatics.

1. Introduction

The Dutch Olympiad in Informatics was initiated in 1991 as a contest for selecting participants for the International Olympiad in Informatics. In the early years of its existence, the number of participants grew to just below 200. Since 2000 the number of contestants in the first round of the contest has varied, but it is proven to be hard to attract more pupils in high school for the Olympiad.

In 2005 the Netherlands was among the first countries to join Lithuania in the Bebras Challenge. Over the last 10 years the number of participants has grown rapidly to over 21000. A lot of contestants of the Olympiad also participated in the Bebras contest. We started looking into how we could persuade other Bebras-contestants into joining the Olympiad as well.

Naturally, there are large differences between the Olympiad and the Bebras contest. It is a bit like comparing the contestants in a marathon run that are aiming for a national championship with the grand total of recreational runners; the participation of contestants serves different goals. However, there are also similarities; both of our contests are about algorithmic and computational thinking, and they aim to challenge the participants to show what they are capable of.

In section 2 we will give a short history of the Olympiad and Bebras in the Netherlands and elaborate on the similarities and differences between the two contests. In sec-

tion 3 we will show the measures we took to bridge the gap between these contests. In section 4 a sample problem in our new approach will be shown. In section 5 we will present the first results of the new contest format and section 6 presents some discussion.

2. Olympiad and Bebras in the Netherlands

From 1999 till 2014 the Dutch Olympiad in Informatics (2016) had a fixed format with three stages.

Three or four programming tasks for the first round are published on our website in September; the final task is to design a program that can enter a tournament. Contestants can register themselves and they are able to submit their solutions till January 15th. Pupils are allowed, even encouraged, to co-operate. Our submission system is able to handle over 10 different programming languages; evaluation is also done by this system. However, all submissions that failed the test are inspected manually, and if the jury is able to fix a small bug, like an IO-format, the program is re-evaluated and a small amount of points will be subtracted (like 3 out of 100 for every fixed error). Contestants that score at least 50% of the maximum score are allowed to join the second round. All contestants that proceed to the second round get a certificate, and there are special prizes for early submitters and the winner of the game tournament (Codecup, 2016).

This second round is in March at a university; two or three problems with subtasks (van der Vegt, 2009) have to be solved and sometimes there are one or two more theoretical questions (van der Vegt, 2012). All languages that our system can handle are allowed. Only in very rare cases the results of the automated evaluation are overruled. The best performing contestants are invited for a trainings course in April on algorithms and problem solving. They will need to switch to C++ or Pascal for the training course and to prepare themselves for the IOI. Finally, the third round in May or June with a limited number of contestants is used to determine the team for the next IOI.

Like in many countries, organizing the Bebras contest is done by the organizers of the Dutch Olympiad in Informatics. The Bebras Challenge was first held in Lithuania (Dagienė, 2006). The Netherlands started with a test contest in 2005. The contest grew rapidly, in 2015 over 1.3 million pupils from more than forty countries participated in their national Bebras. The questions used in these contests are chosen from an international task pool. The contest is about computer science, algorithms, structures, information processing and applications. No prior knowledge is required. Criteria for good Bebras tasks are formulated by Dagienė and Futchek (2008). Dagienė and Stupurienė (2016) give an overview of current research on Bebras.

Contestants compete in their own age division. In the Netherlands contestants have 40 minutes to complete 15 tasks. These can be multiple choice questions, questions where an answer has to be given in the form of an integer or a short string, or interactive questions. The contest runs for a week; the best performing contestants for every age division are invited at a university for a second round (Beverwedstrijd, 2016).

All contestants in the second round get a certificate. In the IOI-style, 1/12 of them get a gold certificate, 1/6 a silver one and 1/4 bronze. The overall winner in each age category wins a gadget with a text inscription showing he or she was the winner of this year's Bebras contest in a specific agegroup.

Performing in the Olympiad takes a reasonable amount of time; our research suggests that a typical contestant can use 20 to 80 hours to write the programs for the first round, though experienced pupils can easily, within a few hours, create solutions that score well enough to proceed to the second round. A contestant in the Bebras contest does one round in 40 minutes.

The questions differ accordingly. Most of the tasks in the Olympiad require writing either a batch or a reactive program. So knowledge of a programming language is required. For Bebras, no prior knowledge is needed. The tasks however tend to test the perception of concepts of computer science and computational thinking. Barendsen et al. (2015) showed that it is possible to use Bebras tasks to assess the understanding of these concepts; in their research they focused especially on algorithms.

Another difference was the participation of girls. Until 2004 the Dutch delegation for the IOI used to be a mixed team (Maggiolo, 2015). In later years there was no possibility to create such a team, because the girls almost completely disappeared out of the contest. In Bebras we reached many girls, around 40% of the contestants every year. In the highest age group this is around 20%.

3. Bridging the Gap

We decided to work on different changes to try to get more Bebras contestants in the Olympiad. Now we are connected with the contestants in Bebras, we can invite them as talented pupils to try the tasks of the Olympiad. But since most of them will not be able to write a program, we now offer a programming course for interested pupils. We give contestants a link to the Dutch translation of the Canadian Computer Science Circles (2016). This is an interactive website designed for student to learn programming in Python.

To give these newcomers a challenge, we had to add a series of tasks on a lower difficulty level. We call these tasks the A-tasks.

We also introduced a new kind of tasks in the first round of the Olympiad, tasks that have the look and feel of a Bebras task, but with a problem that we think will not be solved within a few minutes. You can solve the task without using a computer, but it could also be possible to write a computer program to help solving the problem for you. These tasks are called the B-tasks.

The three tasks of the type we used to offer in the Olympiad are now the C- and D-tasks. An overview is given in Table 1. So we still offer a few hard programming tasks,

Table 1
Tasks in the first round of the Dutch Olympiad in Informatics

Task type	Description	Number of tasks	Points per task	Points per group
A	Introductory programming tasks	5	40	200
B	Theoretical, Bebras-like tasks	4	50	200
C	Advanced programming tasks	2	100	200
D	Game programming task	1	100	100

but we want to attract new contestants in two ways: as starting programmers, after taking the course, solving A-tasks and getting curious about the other tasks we are offering, or as experienced Bebras-contestant, with a few challenging, more time consuming puzzles. We hope that this entices the pupils to try a few programming tasks.

When we started this approach in 2014, we had no clue how many contestants we would be able to attract. We stated in our contest description that the top 100 contestants with a certificate would be invited to the second round, though the last year we had over 100 contestants in the Olympiad was 2003.

We decided to combine the certification system of Olympiad and Bebras, but we set boundaries in advance, so a contestant will know where to aim. We give every contestant that solved all A-tasks or all B-tasks bronze certificate. Performing well for both types is enough for silver. The boundaries are stated in Table 2.

4. Output only Problems: an Example

The Bebras-like theoretical tasks are like the output only tasks at the IOI. On the Wiki of the International Olympiad in Informatics (2016) this definition is given. “A task is of type ‘output only’ if the contestant is provided with the input files, and must only submit the corresponding output files. The contestant can solve each test case by hand or by writing one or more programs in the language of his choice, and doesn’t need to submit these programs.” In our contest we will not use input and output files; the contestant has to download the problem description and to submit the answer in our contest system.

For the first tasks we used exercises that were based on the tasks Dungeon, Palindromes and Cities from Burton (2010) and Coins from Kubica and Radoszewski (2010). Since there is a large period in time that these tasks can be solved in the first round, we could not simply use one instance of these tasks. So we decided to make these tasks personalized. The contest ID of a contestant was used as a random seed to create a problem instance.

Two other restrictions were made:

1. For a full score (50 points) a submission had to be made within a week after the problem statement was generated by the contest system. Each day of delay gives a penalty of 1 point.
2. To discourage guessing, submitting a wrong answer gives a penalty of 10 points. A new submission is only allowed after 24 hours.

Table 2
Boundaries for certificates

Type of certificate	Total score
Bronze	200–399
Silver	400–599
Gold	600–700

The header of a downloaded problem instance is shown in figure Fig. 1. It shows the date and time of production, as well as the username of the contestant.

For task B4: “Woorden leggen”, Crosswords, a contestant had to place six give words on a 10 by 10 grid, each letter in a different cell, in the way a crossword puzzle or a Scrabble board is filled. All words had to be connected and no unintended short words should appear. The number of cells in the smallest enclosing rectangle had to be submitted.

Four of the six given words were the same for all contestants. The other two were selected out of a dictionary; the total length of the two words was 13 and each of the words had at least a length of 5. If our program discovered that this problem instance had no valid solution, another one was created.

This is of course a tricky problem. There are many suboptimal solutions, so a contestant has to be really convinced before submitting. The average score for this task was much below that of other B-tasks, as shown in Table 3.

In our second round, we take one of the B-tasks of the first round and we extend it to a programming task with several subtasks. This way contestants are already familiar with the problem behind the task. In the second round in 2016 we had a programming task Crosswords with 6 subtasks. Input were a diagram with a filled in crossword and a dictionary file with allowed words.

- A. Count the empty cells.
- B. Count the number of different letters used.
- C. Count the number of different words used.
- D. Make a list of all words, in a specific order.



Fig. 1. Header of a problem instance.

Table 3
Scores for B-tasks

ID	Title	No solution	Incorrect	<=20%	<=50%	<=100%	100%	Average
2014–2015 B1	Maze	26	45	0	4	24	83	78,94%
2014–2015 B2	Palindromes	36	14	1	2	42	87	72,52%
2014–2015 B3	Cities	39	8	0	1	24	110	74,99%
2014–2015 B4	Coins	43	8	0	1	23	107	72,37%
2015–2016 B1	Radio mast	29	26	1	1	56	107	78,41%
2015–2016 B2	Connections	42	15	6	3	37	117	72,43%
2015–2016 B3	Subsequence	47	17	0	5	30	121	73,97%
2015–2016 B4	Crossword	63	53	8	3	56	37	53,11%

E. Make a list of all words in the dictionary that can be added in the diagram.

F. Try to fill the diagram, minimizing the number of empty cells.

A seventh subtask was output only: Produce a 10 by 10 diagram with only valid words, and minimize the number of empty cells.

5. Results

After applying these changes in the Olympiad, the number of participants grew rapidly. Fig. 2 shows the number of contestants that earned points with their submissions.

The boundaries we used for giving certificates proved their value. A part of the participants with a bronze certificate was invited to the second round. Table 4 provides an overview of certificates and participation.

In 2015 only 76 out of the 100 invitees competed in the second round, in 2016 we invited 124 pupils and 84 of them joined the contest.

The introductory course in programming was a success. Between 2011 and 2014 the average number of submissions using Python was 18%. In the last two contests it was 45%. So a lot of the new participants use the language that they were trained in. Other contestants kept using the languages they knew already; we had submissions in C++, C#, Java, Pascal, PHP, Visual Basic and Haskell.

It was nice to see how some teachers use the A-tasks as part of their assessment for computer science education. We had at least three classes that submitted some of these tasks, with unique solutions for the participants. This is a form of collaboration that we encourage; we think that especially the A- and B-tasks should find their way to the classroom.

Some of the contestants started with the B-tasks. About 10 % of the contestants restricted themselves to these tasks. Since the cut-off for the second round was 240 points

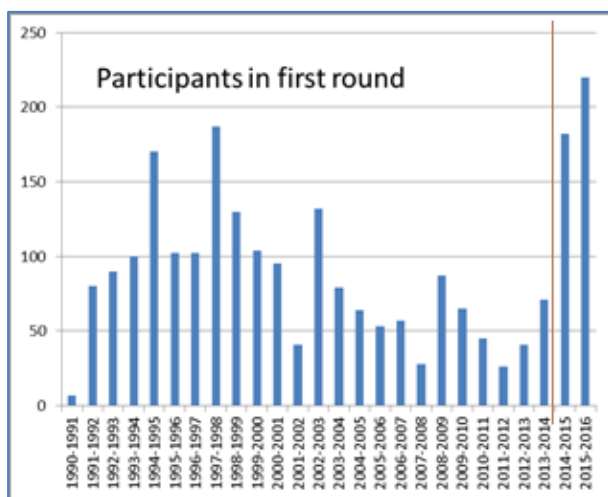


Fig. 2. Participation in the Olympiad before and after the new format.

Table 4
Results of the first round of the Olympiad

Results first round	2014–2015		2015–2016	
	Total	Girls	Total	Girls
Gold	13	0	23	0
Silver	35	0	65	1
Bronze, proceed	54	9	36	9
Bronze, not proceed	9	2	25	6
No certificate	71	12	71	9
Total with score	182	23	220	25
No score	47	12	94	16
Total users	229	35	314	41

in 2015 or 297 point in 2016, competing only with A- or with B-tasks could not get you an invitation for the second round.

And the girls returned to the Olympiad! Between 2005 and 2014 we had only three girls in our contests. None of them performed well enough that we could consider to let them advance to the final round. Table 4 shows that we reach a reasonable number of girls in this contest, about 12% of the participants. Alas, last year only 2 of the girls actually joined the second round. One of them scored very reasonable, the other one is still young and has a lot of years ahead to improve.

6. Discussion

We changed the contest format for the Olympiad in order to attract more Bebras-contestants to the Olympiad. This turned out well. The number of participants was at least tripled, the girls returned to the Olympiad and we welcomed many newcomers in programming, due to the introductory course. The certification method worked out well.

Which challenges remain?

1. We want to attract still more contestants. Given the discussions on the role of programming in education and the emphasis on computational thinking, both Bebras and the Olympiad offer possibilities to discuss tasks and backgrounds in a classroom. In Bebras we have a good working relation with many teachers. Getting the computer science teachers involved in the Olympiad, using for instance A-tasks as part of the assessment, can attract more participants.
2. New forms of tasks will be needed in the near future. Informatics as a subject is changing and developing all the time, for instance by introducing physical computing (Przybylla and Romeike, 2014) and the use of constructivists learning environments (Weigend, 2014). The contest format of the Olympiad gives a focus on algorithms. Other topics need to find a place. So we need to keep experimenting with new question types.

3. The Olympiad is still mostly a man's world. Finding partners, like focus groups on girls and technology, is a condition to improve the participation of girls in the Olympiad.

The tasks we introduced in our new contest were based on the work of colleagues in the international community. We found ready-to-use ideas, that we only had to fit into our new approach.

Exchanging experiences within this community is and will be an important base for further improvements.

References

- Barendsen, E., Manilla, L., Demo, B., Grgurina, N., Izu, C., Mirolo, C., Sentence, S., Settle, A. Stupurienė, G. (2015). Concepts in K-9 computer science education. In: Dagienė, V. (Ed.), *ITICSE'15: Proceedings of the 2015 ACM Conference on Innovation and Technology in Computer Science Education*. ACM, New York, NY, USA, 85–116. DOI: <http://dx.doi.org/10.1145/2858796.2858800>
- Bebras website* (2016). <http://bebras.org/>
- Beverwedstrijd* (2016). (In Dutch). <http://www.beverwedstrijd.nl/>
- Burton, B. (2010). Encouraging algorithmic thinking without a computer. *Olympiads in Informatics*, 4, 3–14.
- Codecup* (2016). (In Dutch). www.codecup.nl or nio3.codecup.nl
- Computer Science Circles* (2016). <http://cscircles.cemc.uwaterloo.ca/>
- Dagienė, V. (2006). Information technology contests – introduction to computer science in an attractive way. *Informatics in Education*, 5(1), 37–46.
- Dagienė, V., Futschek, G. (2008). Bebras international contest on informatics and computer literacy: criteria for good tasks. In: R.T. Mittermeier, M.M. Syslo (Eds.), *ISSEP 2008*, LNCS 5090. Springer-Verlag Berlin Heidelberg, 19–30.
- Dagienė, V., Stupurienė, G. (2016). Bebras – a sustainable community building model for the concept based learning of informatics and computational thinking. *Informatics in Education*, 15(1), 25–44.
- Dutch Olympiad in Informatics* (2016). (In Dutch). <http://www.informaticaolympiade.nl>
- International Olympiad in Informatics (2016). *Output only*. http://wiki.ioinformatics.org/wiki/Output_only
- Kubica, M., Radoszewski, J. (2010). More algorithms without programming. *Olympiads in Informatics*, 4, 157–168.
- Maggiolo, S. (2015). An update on the female presence at the IOI. *Olympiads in Informatics*, 9, 127–137.
- Przybylla, M., Romeike, R. (2014). Physical computing and its scope – towards a constructionist computer science curriculum with physical computing. *Informatics in Education*, 13(2), 241–254.
- van der Vegt, W. (2009). Using subtasks. *Olympiads in Informatics*, 3, 44–48.
- van der Vegt, W. (2012). Theoretical tasks on algorithms; two small examples. *Olympiads in Informatics*, 6, 212–217.
- Weigend, M. (2014). The digital woodlouse – scaffolding in science-related Scratch projects. *Informatics in Education*, 13(2), 293–305.



W. van der Vegt is teacher's trainer in mathematics and computer science at Windesheim University for Applied Sciences in Zwolle, the Netherlands. He is one of the organizers of the Dutch Olympiad in Informatics and he joined the International Olympiad in Informatics since 1992. He was involved in the IOI-workshops on tasks in Dagstuhl (2006, 2010) and Enschede (2008). He also is one of the task designers for the Bebras contest.