CZU: 37.016:004

THE TASKS IN COMPETITIONS AS AN ELEMENT OF THE TRAINING OF PUPILS COMPETITORS IN INFORMATICS

Biserka YOVCHEVA, PhD, professor honoris causa
https://orcid.org/0000-0003-1096-6135
"Konstahtin Preslavski" University of Shumen

Abstract. In the paper are displayed the vision of the autor for the role of the selection and the grading of the problem in different informatics contests for the education and bulding of important professional habits in the competitors – future programmers. The position that problems shoud not be self-serving, that is not be given for the purpose of only creating ranking of the competitors, but should also have additional educational effect is defended.
Keywords: education, competitive informatics, tasks, competition, grading system, tests.

PROBLEME DE CONCURS CA ELEMENT CHEIE ÎN PREGĂTIREA ELEVILOR CONCURENȚI LA COMPETIȚIILE DE INFORMATICĂ

Rezumat. În lucrare este prezentată viziunea autorului asupra rolului selecției și notării problemei în diferite concursuri de informatică pentru educarea și dezvoltarea unor obiceiuri profesionale importante la concurenții – viitori programatori. Se susține poziția conform căreia problemele nu trebuie să fie autoservitoare, care nu sunt date doar în scopul realizării unui clasament al concurenților, ci ar trebui să aibă și un efect educațional suplimentar.
Cuvinte cheie: educație, informatică competitivă, sarcini, competiție, sistem de notare, teste.

Introduction

Competitive Informatics is a popular field of study for future programmers. It develops several useful skills that are necessary for the successful realization and efficiency of those working in the IT sector.

Above all, this type of training leads to an excellent development of the algorithmic thinking of the learners. Moreover, they learn not only to apply the correct algorithms in a particular situation, but also to strive for their maximum efficient implementation.

Secondly, the computer science contestant successfully handles "programming under pressure", which is very useful in critical situations when developing any projects in the IT industry.

Perfect logical error detection skills in a short time are also a product of this kind of training.

Add to this the competitive nature that makes competitive informatics an interesting and inspiring activity for adolescents, and it is clear that this type of programming education is preferable at school age.

The performance of students in various regional, national and international competitions serves as an assessment of the learning outcomes. New competitions have emerged in recent years to give more children the opportunity to compete against their

peers. Different contests have different objectives, but the purpose of this article, is to examine the educational and training element in competitive tasks. Much more useful and applicable in practice would be competitions that include tasks, after solving which competitors learn and consolidate new algorithms and programming techniques. The selection of tasks with such a purpose has become even more complex, especially in recent years, when it has become increasingly difficult for competition organizers to create original and interesting tasks, tailored to the abilities and ages of the competitors.

1. Types of informatics competitions and the technique of their conduct

Several classifications of computer science competitions can be made, based on several different attributes of those involved.

1.1. *Types of competitions, according to the age of competitors*

- *Competitions for students up to 15,5 years - junior age*

The first such international competition was the Balkan Youth Olympiad in Informatics, initiated by Serbia and held for the first time in Belgrade in 2007. It was followed by the International Autumn Informatics Tournament in Shumen, which was held for the first time in 2009 in two separate age groups, one of which was the youth group - up to 15,5 years. The European Youth Olympiad in Informatics, created in 2017 on the initiative of Bulgaria and the interest in it, showed the necessity of competitions for this age. An international competition for young people that has been gaining momentum in recent years is the Info(1) Cup, held annually in the city of Ploiesti, Romania. When talking about the training effect of competitive tasks, it is especially important for this age group.

- *Competitions for high school students - senior (here the age restriction is only related to whether the competitor is still a high school student)*

This group includes all popular computer science competitions, the first of which is the International Olympiad in Informatics (IOI), first held in 1989 in Bulgaria. Following are a number of international competitions in Informatics, such as - the Balkan Informatics Olympiad, the Central European Informatics Olympiad, the Baltic Informatics Olympiad, the International Autumn Informatics Tournament (IATI), the Romanian Masters in Informatics, various open Olympiads, such as the Innopolis Open, the Moscow Open, the Zhautykov Olympiad (in its informatics part), etc.

1.2. *Types of competitions, according to the range of participants*

- regional competitions – these are regional competitions that are usually selective in nature, like the first stage of another larger competition. Such competitions are held wherever there is enhanced training in competition informatics to select participants for competitions of greater scope.
- national competitions – in most countries, it is the National Informatics Olympiads that play a major role in selecting the national teams that participate in international Olympiads. In addition, there are various national tournaments, such as the autumn,

spring and summer tournaments in Bulgaria.

- international competitions – most of them have already been listed above. In different international competitions, different numbers of students are allowed to participate. The International Olympiad in Informatics (IOI) and eJOI, for example, allow only 4 competitors from a given country, while some international tournaments allow for much wider participation - RMI, IATI, Zhautikov Olympiad, INFO(1)Cup, etc. Others, such as the Innopolis and Moscow Open Olympiads, as well as the international CodeIT competition, are open to all interested competitors, but have pre- selection rounds and a certain number of the best-performing competitors are admitted to the finals.

1.3. ***International online competitions***

These competitions do not require on-site attendance and usually have no age or number restrictions - these are competitions like Code Forces, Top Coder, etc. The educational role of tasks in this kind of competitions is particularly valuable, because they are held regularly, and do not require much organizational resources other than the creation of tasks and the maintenance of a competition system available on the Internet.

Each of the competitions listed above have their role in building and training programmers at an extremely high level, who are rare internationally, and are some of the most sought after by IT companies because they are flexible, easily deal with difficult problems and mobile when moving from one technology to another, from one programming language to another. To play this role, the tasks on offer in competitions - their subject matter, comprehensiveness and assessability are particularly important.

2. Elements of the competition tasks

Competition tasks have a characteristic structure and each of them must allow for automated checking by the grading system. According to these requirements, contestant tasks have the following elements:

2.1. ***Condition***

In recent years, a standard for the conditions of competitive programming tasks has been established. They are composed of several parts:

- legend – a description of the problem to be solved, usually a hypothetical life situation that informally represents the problem.
- input data – describes the correct format of the input data and the sequence of their input.
- output data– describes the expected output, similar to the input data.
- restrictions – in recent years, constraints are set as subtasks.

Depending on the constraints, the solution to the problem may have different efficiency.

2.2. ***Tests***

The tests should be able to verify the different subtasks. They can be divided into groups, with each group checking one or more subtasks.

2.3. ***Checker***

The checker is a program that is assigned to the grading system in order to implement a correct check of the contestants' decisions. A standard checker simply compares the answer returned by the program of the contestant with the correct answer. There are cases, however, where multiple correct answers are possible, and then the checker must verify that the contestant's answer is correct.

2.4. ***Analysis***

Of course, the problem can be proposed without analysis, but given the topic of this article - analysis is a very important part of the learning, that would take place by considering the specific problem.

2.5. ***Author's solutions***

There can be several author's solutions that solve the different subproblems, and it is mandatory to provide the solution that the author considers correct. Considering the educational purpose of all competition tasks, it would be good to analyze the participants' solutions after the competition and apply the best ones to the authors' solutions.

3. Evaluation of the competition tasks

In recent years, the sub-task scoring technique has become established, whereby points for a sub-task are only awarded if all the sub-task tests are passed. With full feedback, this is somewhat a good approach, but it completely changes the organization of the competitor's work compared to the recent past, when evaluation was by test and there was no feedback. Whereas then the contestant had to write at least two solutions and checkers, now testing is entirely through the grading system.

Under the new organization, the contestant again has to write several solutions under different constraints, and sometimes under an almost completely changed condition. Actually, in this way, student competitions are similar to ACM-style competitions, i.e. the competitor has to solve a large number of problems with a complete solution (i.e. a solution covering all tests). In ACM-style there are no problems with having a large number of competitors with the same problems solved, because there the time factor is included in the ranking. In school competitions, however, the same results are increasingly observed for different competitors, even if the competition is held over two days and this implies some differentiation. One way of trying to avoid this problem is to grade each subtask with a different number of points, but this does not always lead to the desired result. Increasingly, at various competitions, we see entire groups of competitors, who have solved specific subproblems and as a result have the same number of points, making it difficult to determine the best.

In sub-task scoring, it matters a great deal how the sub-tasks are separated and what role each will play in the contestant's score. In many cases, misjudging the complexity of the subproblems can lead to anomalies like the contestant writing some very hard algorithm for 5 points, whereas if, after some observation, he finds that he could apply some particular case of that algorithm (which is the point of the condition) where he suddenly ends up with at least 10 times the score. Such were the tasks of the first day of eJOI 2022(3). These are sometimes anomalies that do not stimulate children to learn serious algorithms, and even in the general case to try to "cheat" the system by separately processing supposed tests.

There is also another paradox related to the considered evaluation- a relatively easy intuitive solution brings most of the points, while in order to get the full result it is necessary to write some heavy algorithm that takes a lot of the contestant's time. An example is the ***addk*** task from the first competition day of eJOI 2021(4), where 92 points were obtained for an elementary, trivial solution, but to get the remaining 8 points required writing a lot of code, implementing work with tree structures, etc.

Since the main purpose of this report is precisely the educational and training purpose of the tasks, the balance in the evaluation is particularly important and will be discussed in the next paragraph.

4. The competitive tasks as part of programming education

The problem posed in this article (as already mentioned) is the selection of tasks for the competitions so that they effectively enforce some basic concepts in programming education.This is especially true for the junior competitions and the regional and national competitions. However, tasks from international contests can also be used for the same purpose. This paragraph discusses examples of tasks providing such opportunities.

Depending on who the organizers of the competitions are and what their goals are, the tasks are selected in different ways. And although in important international competitions, such as the IOI, for example, it is the ranking of the competitors that is at the forefront, and not the stimulation of their training, here the principle is defended that we should still think about the tasks being selected in such a way as to have some educational and training effect.

In recent years, so-called **ad hoc tasks** have become popular. These are problems in their own right that require no special theory to solve. For their solution, sometimes effective observation is sufficient, which, if it leads to the correct result, provides the contestant with a solved problem. This type of assignments allows the truly brilliant students to excel and take their rightful first places. The problem with these tasks is that they do not admit a partial solution. Usually, their competition results are in the two opposites - either the problem is completely solved, or not solved at all.

Such was the **kdag** task given at RMI 2022(2). Those who guessed the unique solution are counted on fingers, with the majority of participants having 0 points. In practice, such tasks, apart from highlighting the winners, do not play a particularly effective role in

ranking the other participants. The effect would have been the same if this task had not been given. It is easy to see from the rankings that those who ranked first, would have done so without having solved the problem in question. Many more such assignments could be listed, including those given to IOI, but it is not our purpose at this time to make such statistics. It is not necessary to negate these tasks altogether, but it is also unwise in a competition to have all tasks be ad hoc.

Another type of problems can be called "textbook" problems - they require the recognition of a standard algorithm and its implementation. If the goal of the competition is training, these tasks are fine, but if it's a competition where rankings need to be made and most participants are expected to be well-prepared in all the standard algorithms, the task again makes for inefficient rankings. In this case, most competitors have the full number of points and very few are those who failed to do the task. Such is the **SubsetMex** problem from the just passed European Girls Olympiad or again an example from RMI 2022(2) - the **revstring** problem.

However, these tasks are a bit more useful because they give the competitors a chance to believe in their abilities and not end up with a zero score at the end of the competition. This also matters, especially at the age referred to in the article - high school students and especially junior high school students. In terms of ranking, however, this kind of task is just as unnecessary as ad hoc tasks.

Here the established thesis is that the best way to select the problems is to make each sub- problem require a different level of knowledge of the contestant - to have a naive solution that can be written by everyone and bring a certain number of points. The subtasks should follow the level of training, with each successive task applying increasingly difficult algorithms and data structures, while the maximum number of points requires ad hoc observation and solution.

Such are a large part of the tasks that have been given in recent years at IATI, Moscow Open finals and other competitions. As an example I can point to the task **news** from IATI - 2021(1). A careful reading of both the condition and the analysis, and of the contestants' results, reveals that the problem involves a large number of algorithms and data structures, with each subgroup requiring increasingly in-depth knowledge and advanced skills on the part of the contestant. In spite of all the requirements, however, the task gives a good distribution of competitors, having quite a difference as number of points. Here, in addition to achieving the goal of obtaining an equal distribution in the ranking, the other goals that have already been commented- training and educational- are also achieved.

It is good to pay attention to one more problem when creating the tasks- the grading system. There are examples of tasks that meet the requirements described in the previous paragraph, but the distribution of points is not very appropriate. There are many instances of such uneven distributions of results, but some of the most recent are some of the eJOI - 2022 tasks. Many interesting tasks were offered, but in most of them the "easy" points were

very difficult to achieve and were too few. There was an opportunity in some of the tasks to have a smoother transition to the difficult sub-tasks, which would have stimulated children to struggle to solve them. When the work to be put in is very low "paid" the competitor becomes demotivated and generally gives up fighting until the last moment, and this is one of the qualities that competitive programming builds in kids.

Ultimately, it should be noted that creating good assignments is an extremely difficult, creative job, and the professionals who do it usually do it as an extra commitment. Because there is a lot of work involved in creating a task, authors often get "worn out". At some point, ideas are born less and less, and the next race is coming up.

One way in which the problem of task creation can be solved is to pool the efforts of colleagues from more than one country and, especially when it comes to training or selection competitions, to take tasks of several different teams. In this way, more and more quality tasks will be created and the selection of competitors in different regions and countries could be more objective.

Bibliography

1. http://www.math.bas.bg/infos/Shumen_2021.html
2. https://rmi.lbi.ro/rmi_2022/
3. https://ejoi2022.org/
4. https://sepi.ro/ejoi/2021
5. MOLLOV, Anton; YOVCHEVA, Biserka; PETROV, Peter. Internal contests as an element of the training of pupils for competitions in informatics. *Математика и математическо образование*, София, 2009. с. 217-233, ISBN: 978-954-8212-01-4. 2009 г.