МАТЕМАТИКА И МАТЕМАТИЧЕСКО ОБРАЗОВАНИЕ, 2009
MATHEMATICS AND EDUCATION IN MATHEMATICS, 2009
Proceedings of the Thirty Eighth Spring Conference of
the Union of Bulgarian Mathematicians
Borovetz, April 1–5, 2009

INTERNAL CONTESTS AS AN ELEMENT OF THE TRAINING OF PUPILS FOR COMPETITIONS IN INFORMATICS*

Anton Mollov, Biserka Yovcheva, Peter Petrov

The article presents the experience of the Private School of Mathematics and Informatics A&B (Shumen, Republic of Bulgaria) in the training of pupils for competitions in Informatics. The attention is focused on different aspects of the organization and the conducting of internal training contests. According to the authors, this is a direction with a very important contribution to the training of good competitors. A sample contest is presented and an analysis of the different stages of its preparation and conducting is made.

1. Training pupils for competitions in Informatics. The purpose of the paper presented is to share the experience and the conclusions of the authors from their long years of practice in the training of pupils for competitors in Informatics and to challenge different points of view on the problems in discussion.

The training of pupils for competitors in Informatics varies in different directions (organization, content, methodology, etc.), united in a system. Some of them are discussed in other publications of ours. Without commenting them in details we will point out the following:

The authors' experience is obtained in the School of Mathematics and Informatics A&B (town of Shumen, Republic of Bulgaria [1]) during the period 1998–2008.

The training in the School is conducted with children at the age of 8 to 19 years old. It starts with training in information technologies and building up of skills for practical work with computer. This initial training continues up to two years and on a descriptive and operational level it includes some basic concepts of informatics (program, operational system, file system, computer networks, internet work etc.). To get involved in these occupations, no special training or selection of the trainees is required. Further, the training continues with introduction in informatics and programming (on the basis of the programming language C/C++), with objective for the children to be able to determine as soon as possible the direction in which they will continue their training – information technologies or programming. Thus the systematic training in programming starts at the age of 10–11 years.

*Key words: Programming contest, Olympiads in Informatics, IOI, training, preparation.

For the above mentioned 10-year period about 800 pupils are taught in the School, as the average number of pupils per year is between 60 and 100. They are distributed in groups, corresponding to the competitive groups, approved by the National Commission of Informatics formed each year by the Ministry of Education and Science, which in the last years have been the following: E (4–5 grade), D (6 grade), C (7–8 grade), B (9–10 grade) and A (11–12 grade). On a national scale these groups are formed on the basis of the age. The participation in a group non-corresponding to the age of the contestant is also allowed, but if the pupil is older than the age of the group she/he is not allowed to take part in the rating. Taking all this into consideration, the groups in the School A&B are formed on the basis of the age, the acquired knowledge and, above all, of the proven results.

The training in the School is conducted on the basis of curricula, materials and handouts made by the team of School's teachers, in compliance with the national program for training for competitions. In our opinion, some of the issued school manuals have no analogy in the world school literature of Informatics [2].

A part of the system are the regular conducting of internal for the School training competitions and including the results from each of them in the current rating of each student.

Apart from the system of national competitions in Bulgaria [3], one of which is created in the School, according to an offer of the School, and is conducted traditionally in the town of Shumen, our students participate in international competitions (TopCoder, Usaco, Timus, BOI, IOI, etc.) also. One of the last participations of a team of the School was in the final competition of TopCoder High School, which took place on May 19th 2007 in the University Pardu, USA.

The traditionally good results of the teams and individual competitors from the School (Rostislav Rumenov – 3 silver and 1 gold medals from Balkan OI, 1 silver and 2 gold medals from IOI, a special award of IOI'2006 for the only solution to the most difficult problem, etc.; Rumen Hristov – 2 gold medals from Balkan Olympiads in Informatics for students up to 15,5 years old, etc.) show the efficiency of our work and they are good reasons to be shared in the present forum.

The training of the competitors in the school has the following components: theoretic training, analysis and solving of problems, individual work in class, homework, and training competitions. Not all of the components are included in each educational period. During the training competitions, the acquired knowledge and skills from the class activities are combined with elements from the other components and make possible the precise assessment of the work of the students.

2. Organizing and leading of training contests. The competitions in programming are two types: the first ones are organized according to the rules of the International Olympiad in Informatics, the other according to the rues of ICPC of ACM. Does not mater what kind of contest will be the internal competitions in the School it pass through the following stages:

1. Selection of the tasks.
2. Conducting of the competition.
3. Checking and assessments of the results.
4. Analysis of the competition as a whole and of the individual presentation of each one of the participants.

The first stage requires significant efforts of the team, preparing the competitions. The efficiency and the teaching effect of the competition depend to a great degree on this stage. The problems which the team must solve are:

- whether the problem set is thematically oriented or a general training;
- what the type of the competition will be (according to the rules of ACM ICPC or of IOI). This will determine the number of the tasks and the range of the knowledge and the skills, which will be trained during the competition;
- the character and the level of complexity of the theoretical material included into the tasks;
- intriguing, attractive, even funny, and at the same time, sufficiently precise formulating of the tasks;
- selection of sufficiently indicative (typical), and in compliance with the set of objectives, tasks, including tasks from other competitions and authors;
- deciding the eventually including of tasks, the solving of which requires unknown to the pupils theoretical knowledge or programming skills and which could be considered at the soon forthcoming classes;
- writing of a good authors' solutions (sufficiently effective from the author's point of view and in compliance with the pupils' knowledge and skills, creating useful and preventing harmful programming habits);
- creating of appropriate test cases (corresponding to an all-comprising analysis of the possible algorithmic mistakes) and checkers.

At the second stage of the conducting of the competition it is necessary:

- to provide the computer laboratory with the necessary for the competition software – operating systems, compilers, Integrated Development Environment (IDE), net software, as well as a grading software system. The last one is not obligatory, but its use not only increases the efficiency of the organization and conducting of the contest, but teaches discipline, getting the pupils involved in a real competitive atmosphere. The choice of grading system depends on the kind of conducted competition – in the School we use PC 2, for ACM ICPC-styled contests, and an authors' system, based on the so called "Korean System", for conducting of competitions according to the rules of IOI;
- to provide competent and exact answers to the questions on the tasks statements, asked by the pupils.

  If a grading system is used the third stage of the competition is completely automated.

  Without any doubt, the efficiency of an internal competition depends enormously on the completeness and the depth of the conducted at the fourth stage analysis. For the purpose the analysis must include:

- theoretical comments;
- author's explanation of the solution with all possible nuances;
- a detailed analysis of the approach to the solution and the eventual mistakes of each of the competitors.

We consider the practice to include pupils from the upper age groups in the teams for the conducting of the competitions at each one of the stages adopted in our School to be extremely useful.

3. A sample contest. We present a sample competition with illustrations of the different stage and analyse the results as well as the mistakes of the pupils and the objectives achieved.

The objective of this sample contest is the thematic training for consolidation of the pupils' knowledge of arrays – going around an array, sorting the elements through counting, finding a set of elements with given properties, etc. The problem set is for the level of group D, i.e. for six graders of the Bulgarian school – twelve years old students.

Task 1. Prize. The Swiss company "Sweet Life" put another kind of chocolate sweets on the market. They were luxuriously packed and arranged in one line in a rectangular box. On the occasion of the new product the firm decided to surprise its clients with a prize and it put a small piece of cardboard with a number written on it under each one of the sweets. The winners were those two clients who first took sweet with the same number hidden under them. The sweets were taken in a row – the first client took the first sweet from left to the right in the row, the second took the second from left to the right, etc. Write a program **PRIZE**, which determines the awarded clients.

Input. On the standard input an integer N is given followed by N positive integers not larger than 100000 – the numbers that are hidden under the sweets ($N \leq 100000$).

Output. On the standard output two integers have to be printed – the numbers of these two clients, who must take the prize. It is always guaranteed that they are two clients who win.

Examples:

Input	**Output**
10 3 4 5 6 19 3 2 5 2 4	1 6
4 20 30 30 20	2 3

Task 2. Warming. Tired from the unusually warm winter, the habitants of the village of Nevsha decided to find out whether this was the longest warming period of the winter in the history of the weather observation in this village. The mayor of the village asked meteorologists to explore the statistics of the previous years. They were interested in how many days the longest warming period had continued.

A period is called warming if in every day of the period the average day temperature exceeds 0 degrees. Write a program **HOT**, helping the meteorologists with their work.

Input. On the first line of the standard input an integer N is given – the total number of the days in consideration ($1 \leq N \leq 100$). On the second line N integers are given, divided by spaces. Each one of these numbers is the average day temperature of the correspondent day in consideration. The temperatures are integers within the range from -50 to 50.

Output. On the standard output only one number has to be printed – the length of the longest warming period, i.e. the greatest number of consecutive days, during which the average day temperature exceeded 0 degrees Centigrade. If the average days temperature was not positive, then the program has to print 0.

Examples:

Input	Output
6 −20 30 −40 50 10 −10	2
8 10 20 30 1 −10 1 2 3	4
5 −10 0 −10 0 −10	0

Task 3. Triangular Sequence. A sequence with N elements ($3 \le N \le 20$), containing integers between 0 and 1000, is called "triangular", if there exists an integer M ($1 < M < N$), such that the first M elements of the sequence form an increasing subsequence (i.e. each of this M elements is less than the next in the row), and the rest $N - M$ elements form a decreasing subsequence. The M-th element is the end of the increasing and the beginning of the decreasing part of the sequence.

Write the program **TRISEQ**, which determines whether the sequence is triangular or not.

Input. On the first line of the standard input an integer N is given, that is the number of the elements of the sequence ($1 \le N \le 100$). On the next row the N integers in consideration are given, divided by spaces.

Output. On the standard output the program has to print out Yes and the found number M, divided by one interval, if the sequence is triangular. Otherwise the program has to print out No.

Examples:

Input:	Output:
6 3 40 67 200 123 112	Yes 4
7 56 34 34 115 450 345 211	No

The first task requires an ***efficient solution***. There is an obvious solution, which finds the searched index through one scanning of the array for each of its elements – algorithm with complexity $O(N^2)$. In most cases this is the first thing pupils think about. But for this task a solution with complexity $O(N)$ is required, which is achieved through sorting of the numbers trough counting. When a greater limitation for N is specified, it has to show to the pupils that they must seek the linear solution. Similar tasks are considered with them in the classes before the training contest, so they are expected to think of an efficient realization.

The second problem requires a ***working with arrays' technique***. Here the limitations are weak and the efficiency is not the first priority. Almost any algorithm for finding the length of a maximal platform in the array will be linear. So the pupil has to concentrate on the technical problems of the realization.

The third problem gives student an opportunity for to write a solution without using of an array. However, by using an array the problem can be solved much easier and the probability to make a technical mistake is considerably smaller.

The training contest is prepared for conducting in the style of IOI. For each one of the three problems ten test cases are prepared, each of them assessed by 10 points. The standard duration of the internal training contests for this group is 2 hours.

As it was already mentioned, the role of the test examples is particularly important for the efficiency of the training. That is why they are well selected for each one of the tasks. For the first task in 60% of the tests cases N is sufficiently big, so that a non-efficient but true solution may get not more than 40 points. In another kind of competition for this age group the emphasis of the assessment would not fall on the efficiency but here it is the main objective of the task. Yet, for this task it is easy to make various mistakes in the realization of the algorithm and by that reason there should be foreseen tests cases to catch them. Such a case is the second example given in the statement of the task, where mistakes could be made with symmetrical rows or with rows in which there are many equal elements.

For the second task it is more important to emphasize the technique of working with arrays. There must be test cases which consider all the particular occasions: cases, where the maximal platform is at the beginning or at the end of the array, as well as cases, in which there are no elements corresponding to the condition. It would be appropriate to have a test case in which there is only one platform of length 1, and so on.

The third task has been taken from the first round of the Bulgarian National Olympiad in Informatics for 2005 [4]. In the test cases with positive answers, the number M has to be also printed out in order to prevent the attempts to give randomly true answers for a part of the cases. In tests cases with negative answer it is good to be considered all possible variants, as the presence of equal elements, the cases in which the row constantly increases or decreases, etc.

Our observations indicate that 20% of the pupils succeeded in solving the tasks correctly and they deserved 100% of the points. Common were the mistakes due to problems with technology of programming. In a great percentage of the cases the pupils did not consider well enough the length of the arrays. We also noticed that the particular cases were often mistreated. The careful analysis of the solutions of each contestant is of great importance for the consolidation of knowledge and improving the skills of the whole group. Another very important factor for preventing mistakes of this type is the systematic offering of problems of the same kind in few internal competitions. The experience showed that after making the mistake several times, the pupils stopped doing it.

REFERENCES

[1] `http://ab-bg.com`.

[2] Б. Йовчева, И. Иванова. Първи стъпки в програмирането на C/C++, Издателство КЛМН, София, 2007.

[3] Kr. Manev, E. Kelevedjiev, St. Kapralov. Programming Contests for School Students in Bulgaria, *Int. J. Olympiads In Informatics*, **1** (2007), 112–123.

[4] `http://infoman.musala.com`.

Anton Mollov, Biserka Yovcheva
Konstantin Preslavsky University of Shumen
115, Universitetska Str.
9700 Shumen, Bulgaria
e-mail: a_mollov@abv.bg, bissy_y@yahoo.com

Peter Petrov
A&B Private School
13, Hristo Botev Str.
9700 Shumen, Bulgaria
e-mail: peshoto_bg@yahoo.com

ВЪТРЕШНИТЕ СЪСТЕЗАНИЯ КАТО ЕЛЕМЕНТ ОТ ПОДГОТОВКАТА НА УЧЕНИЦИ ЗА СЪСТЕЗАНИЯ ПО ИНФОРМАТИКА

Антон Моллов, Бисерка Йовчева, Петър Петров

Представен е опитът на Школата по математика и информатика А&Б (гр. Шумен, Република България) от подготовка на ученици за състезания по Информатика. Вниманието е съсредоточено върху различни аспекти на организацията и провеждането на вътрешни тренировъчни състезания. Според авторите това е направление с много важен принос за възпитаване на добри състезатели. Представено е примерно състезание и е направен анализ на различните етапи от неговата подготовка и провеждане.