Olympiads in Informatics, 2012, Vol. 6, 158–169

IOI Training and Serbian Competitions in Informatics

Aleksandar ILIĆ[1], Andreja ILIĆ[2]
[1]*Facebook Inc*
1601 Willow Road, 94025 Menlo Park, California
[2]*Faculty of Sciences and Mathematics, University of Niš*
Višegradska 33, 18000 Niš, Serbia
e-mail: {aleksandari, andrejko.ilic}@gmail.com

Abstract. This paper summarizes the story of the Serbian competitions in informatics. The annual cycle of these competitions is described along with the selection and training process. The structure and work of the Serbian Scientific Committee for high school competitions in informatics, which is responsible for selection of Serbian IOI team, is given. We address content and problems of our general education in computer science in schools. Examples of problems are also provided in Appendix.

Key words: informatics competitions, Serbia, Serbian Olympiad in informatics, SIO.

1. Introduction

This paper summarizes the story of the Serbian competitions in informatics. Our annual cycle is divided in five stages. In the next section we described these stages and included both technical and organizational part of these competitions. We described the selection and training process for our national team. The structure and the work of the Serbian Scientific Committee for high school competitions in informatics, which is responsible for selection of Serbian IOI team, is given in Section 3. Selection process and preparation of the tasks is, probably, the hardest job for the Scientific Committee. This process is divided in stages and every stage is described in this paper. We addressed curriculum and problem of our general education in computer science in schools in Section 4. Some examples of problems are also provided in Appendix. The main web pages with all important information on Serbian competitions in informatics are `http://www.dms.rs/` and `http://www.yuoi.nis.edu.rs/` (currently only available in Serbian). For the national reports of other countries see references.

2. The Selection and Training Process in Serbia

The annual cycle of Serbian competitions consists of five levels, in the increasing order of the task difficulty: online Qualifications, Regional, National, Serbian olympiad in informatics (SIO) and IOI Team Selection competition. The fifth level is not mandatory and

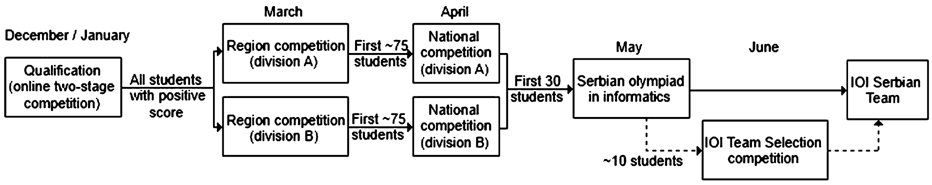

Fig. 1. Diagram of the annual cycle of Serbian competitions.

it is organized in cases where the selection of the national team for IOI is not clear from the previous rounds.

In the last four years, the Scientific Committee introduced one additional online competition – **Qualifications**. The tasks are set up on Z-training site (we will address this online-judge later in this report). This competition is a two-stage competition (December and January). In each stage, students are given five problems and they have one week to solve them. Difficulty of the problems for qualifications varies from trivial one (e.g. primality testing) to the ones similar to National competitions. The only condition that students have to meet to participate in Regional competition is to have positive score on Qualification. The main motivations for these online qualifications are popularization of the programming competitions and algorithmic type of problems to the beginners (input and output, testing on data sets, scoring . . .). Around 350 students have submitted at least one problem at last year's Qualifications.

For Regional and National competition students are divided in **two divisions: A and B**. The division A consists of students which are second, third and fourth grade in schools that are working by the program of the Mathematical Gymnasium. All the other students are in the division B. The main difference between the problems in these two divisions is difficulty. Also, the minimal number of points needed for advancing to the next stage can be different between divisions. The reason for two divisions is to give chance for everybody to win a national award – without this the awards will be mostly won by the students from division A. Usually the number of competitors that receive diplomas and awards on National competition and SIO is one half of the number of all competitors and the awards are distributed by the ratio $1 : 2 : 3$.

The Regional competition, usually held in the beginning of March, is the first stage of the Serbian competitions. It is organized in every region, around 30 of them, where students are struggling with the three algorithmic problems for which they have three hours to solve. Before the competition, the Scientific Committee provides the problem statements, test data and a simple console application for evaluation that works with executable files. Using this, the regional committee can evaluate contestants' sources and get the preliminary results for their students. Exactly one of the problems (the most difficult one in division B and the easiest in division A) is the same for both divisions. After the competition, all source codes are sent to the Scientific Committee for final evaluation and the results from this evaluation are final. These results are united and, approximately the first 75 students per division are selected and invited to the National competition.

The National competition has the same structure as Regional, except that the students have five hours to work on problems. The competition is held in three different

Table 1
Statistics of school grades for students in Regional competition and SIO

Grade	Percentage of students on the Regional competition	Percentage of students on SIO
1	19	14
2	22	23
3	35	30
4	24	33

sites (Belgrade, Niš and Novi Sad). In these university towns the Scientific Committee has representatives, so the communication and evaluation is much simpler. The first 30 students in this completion are invited to SIO, usually the first 24 from A category and 6 from B category.

SIO has a similar structure with that of the IOI. The competition is organized in the capital city of Serbia, so all competitors are in the same place. In this stage of competition there are no divisions – all students are solving the same tasks. This competition is a two-day competition (without break days), where students are given three problems and five hours each day. Solutions are graded in real-time, but students can not see the results until they leave the competition room.

Table 1 contains statistics for student grades in high school competitions from Regional and final level. After adding Qualification stage, we notice an increase of the number of contestants from the first grade and the number of contestants having positive score on the Regional competition.

For the Regional and National competition there is no global training. Some schools organize preparation, which are different from town to town. For SIO, depending on the budged, there can be additional competition during the training camp, where the national team is being selected.

IOI training camp for our national team, plus some additional competitors from the first or second grade, is the only official training that is organized every year. Most of the times IOI training camp is organized in some mountain camp where students have lectures all day. General structure of the lecture is divided into two parts: 4 hours of theoretical lecture (on a blackboard) in the morning and 4 hours of coding in the evening. Typically, this training lasts one week. We present part of the last year training schedule in Table 2.

All types of tasks on Serbian national competitions are designed to be of algorithmic nature and all of them have to be solved by directly programming on the computer. Similar to IOI's rules, there are three general categories: batch tasks, interactive tasks and output-only tasks. On the other hand, there are no restrictions for known algorithms on the last level of competition (SIO).

The knowledge needed for each stage of the annual cycle is presented to the students in the beginning of every year. This list is not strict and it is given to represent some sort

Table 2

Part of the last year training schedule

Day	Time	Lecture
01	10h – 14h	Graph theory
	15h – 19h	Graph theory (coding)
	21h – 23h	IOI tips and tricks
02	10h – 14h	Data structure
	15h – 19h	Data structure (coding)
	21h – 23h	Interview questions
03	10h – 14h	Game theory
	15h – 19h	Game theory (coding)
	21h – 00h	Output only problems
04	10h – 14h	Problems from the other national competitions
	15h – 20h	IOI practice competition

of guideline for students and their professors. Of course, if a student knows all of the given algorithms, that does not imply that he / she will do well on the competition.

- **Regional competition**
  - simple data structures (arrays, strings, matrices, sets, lists);
  - simple mathematical algorithms (number systems, prime numbers, Euclid's algorithm);
  - backtrack and recursion;
  - elementary sorting algorithms (selection sort, insertion sort, bubble sort, count sort);
  - basic geometric objects (points, lines, segments, circles) and their manipulation.
- **National competition**
  - fast sorting and searching algorithms (quick sort, merge sort, binary search);
  - representing simple graphs and trees (DFS, BFS, connected components, topological sort);
  - dynamic programming and combinatorial enumerations;
  - simple geometric algorithms (area of the polygon, point in polygon);
  - greedy algorithms.
- **Serbian Olympiad in informatics**
  - graph algorithms (Dijkstra's algorithm, MCST, SCC and BCC, Eulerian path, Flow algorithm, Bipartite matching, ...);
  - data structures (Hash tables, Segment tree, Cumulative tables, Disjoint data set, ...);
  - geometric algorithms (convex hull, closest points, triangulation, polygon intersection);
  - advanced algorithms for pattern searching (KMP, suffix tree, ...).

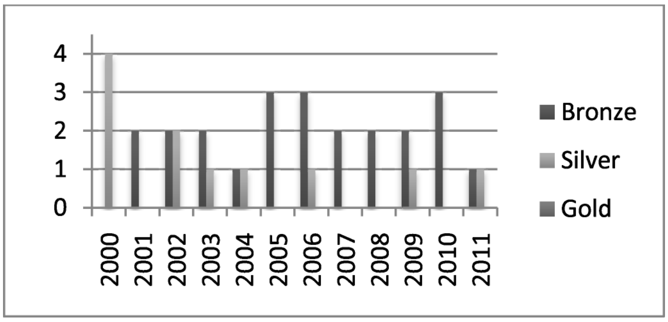

Fig. 2. The summary of IOI medals achieved by Serbian students.

All tasks on Regional and National competitions are evaluated offline in one centralized place and the results are afterwards distributed in each place where the competition was held. For Serbian Olympiad in Informatics we use online evaluating system written by Nikola Mihajlović in Java and we started to use it from 2004. The system supports programming languages C, C++, Pascal and Java. All mentioned types of problems are included in this system. Every problem is tested on some data corpus, where test case(s) from the problem statement is precondition for accepting the code for evaluation and the authors must provide checkers for testing. On these competitions, every problem has equal maximal number of points (100) and the number of test cases can vary.

3. Serbian Scientific Committee for the High School Competitions in Informatics

The Scientific Committee for the high school competitions in informatics, which is responsible for the technical organization and management, consists of the chairperson, Dragan Urošević, and Serbian former IOI and SIO competitors, which are mostly students or software developers in various companies around the world (around 20 active members). The Scientific Committee is a part of Mathematical Society of Serbia.

The task creation and selection are the most important jobs of the Scientific Committee. This process has three of stages:

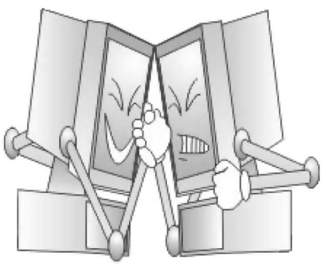

Fig. 3. The Serbian Scientific Commitee logo.

- *Submitting the problem proposals.* In this stage, more or less, every member of the SC submits some problem proposals without solution or hints. This is done before the actual formal meeting for any competition, so the members can think a little bit about the problems and get some intuition about their difficulty and possible approaches.
- *Problem selection phase.* After the discussion of the proposed problems, Scientific Committee choses the problem set (five or three tasks). The rest of proposals are saved for some other competitions or rejected as known or inadequate.
- *Problem creation.* For every problem we select one main author and one tester. The author of the problem is responsible for writing the task statement, creation of the test data and writing the official solution for this problem. On the other side, tester is writing another optimal solution (not necessarily the same algorithm) and some other possible approaches which are not optimized in terms of memory or time constraints. This way we want to be sure in the problem and test data correctness and have good distribution of points depending on the algorithms.

Besides this, Scientific Committee for the high school competitions in informatics is involved in many other events / projects. The members of SC are also active on various forums and prepare problems for other competitions, such as TopCoder. Here we will address the main three.

Z-Trening is an online resource `http://www.z-trening.com/` of challenging programming exercises (mostly compiled from the previous international programming competitions) that serves for training for the future contests. Z-trening is created by Serbian former competitor, Aleksandar Zlateski.

Each problem is accompanied by a series of tests that can be used to assess the correctness of a given solution. Most of the problems are from the Serbian competitions. Problem archive also includes problems from various other competitions with official data set. Users are free to submit their solutions in *Pascal*, Java, *C*, or *C*++ and are graded based on the number of tests passed. Each user is associated with a history of his / her results and later awarded medals. Z-Trening also hosts online programming competitions, which usually last for 3–4 hours and include several new problems. Most of the material on this online judge is collected and organized by Scientific Committee of our National competition.

BubbleCup is a coding contest created by Microsoft Development Center Serbia in 2008. It is a team contest, similar to the ACM Collegiate Contest, aimed at university and high school students. The original goal of the contest was popularization of computer science among students and additional training for international competitions. Soon that idea was outgrown and the vision was expanded to attracting talented programmers from the entire region and promoting the values of communication, companionship and teamwork. The contest consists of two parts – qualifying rounds in the spring and the final round in September. Qualifications are organized in two stages as an online competition where students have twenty five days to solve 10 problems. Problems are selected from Timus online judge. The final is on-site round with the same structure as ACM finals and the best teams are rewarded with valuable prizes.

With each new iteration the contest has been growing. In its first year close to 100 participants (only from Serbia) took part and in 2011 it attracted more than 500 participants from 8 different countries in the region. Besides the competition itself, one of the great outcomes is BubbleCup Booklet. The booklet contains detailed analysis and solutions for all problems, both from qualifications and finals. The official BubbleCup website is `http://www.bubblecup.org/` and it contains detailed information about the competition and BubbleCup Booklets from previous years.

Junior Balkan Olympiad in Informatics is a new competition, established in 2007 in Serbia and the first JBOI was held in July 18 – 23. in Belgrade. The competition is open for students aged 15.5 and younger, residents of Balkan countries. The competition consists of two 3-hour rounds. All problems are algorithmic in nature and available programming languages for the competition are C, C++, C# and Pascal.

Each competing team consists of two leaders and four students – competitors. The following countries participated so far: Bosnia and Herzegovina, Bulgaria, Croatia, FYR Macedonia, Greece, Romania, Montenegro, Serbia and Turkey. The primary motive and goal of this competition is identifying talents among young students. We have invested much effort and resources to support young programmers in their education, personal development, work and on their road to success. Apart from popularization of informatics and programming among students, the goal of this event is to create friendly environment in which students can make new friends and exchange information in elementary school.

The third and fourth JBOI was held under International tournament in informatics in Shumen, Bulgaria. The fifth one in 2011 was held jointly with BOI in Bistrita, Romania.

In the last five years, Mathematical Gymnasium in Belgrade, Niš and Novi Sad started experimental class of students of seventh and eighth grade with some more intense courses in mathematics and informatics. Some of the best students from these classes also took part on high school competitions in division B with nice results (three students even qualified for SIO last year).

4. Short Overview of General Education of Informatics (Computer Science) at Schools

In Gymnasiums, informatics is thought with one or two classes per week, which is really low compared to other important courses. The aim of the course is the acquisition of basic computing literacy and training students to use computers in their further education and work. Unfortunately, the emphasis is far from the algorithmic nature. The tasks of teaching computer science and information technologies are (from the first to the fourth grade):

- introducing students to the internal organization of computer systems;
- introducing and training students to use the operating system;
- introducing and training students to use word processing programs and multimedia applications;
- introducing students to the principles of representation and processing of drawings on the computer;

- introducing students presentations on the Internet and Internet search principles;
- training for primary use of one of the programs to work with the tables;
- develop the ability to fully, accurately and concisely define the problem;
- introduction to the algorithmic way of solving problems;
- the basic data types and algorithms;
- introduction and practical use of Pascal programming language for solving problems on a computer;
- mastering the principles of creating modular and well-structured program;
- introduction to the windows-based programs;
- introduction and practical use of Delphi or Java programming environment for solving problems on the computer;
- learning and mastering the basic operations of databases.

On the other hand, in Belgrade there is a special school for gifted and talented students in mathematics and informatics – Mathematical Gymnasium. In other university towns, there are also branches of this school (one department per school class) that are working by this program. Most of the professors in these classes are professors at the Universities in Serbia. The program differs from the one mentioned above. Students gain knowledge in programming languages (Pascal, Delphi, C, C++, C#, Java; Fortran; Prolog), operating systems (Windows, Linux, Unix), databases and DBMS (Oracle Database, Microsoft SQL Server, MySQL), and in various IT and ICT fields.

5. Conclusion

In spite of the non-algorithmic content of programming courses in our schools, Serbian IOI team shows solid results. In our opinion, this is the largest problem that we are facing. This gap between school and competition type of problems is something that we want to put emphasis on in the next years. Achieving better results and higher interest of young students in this type of contest in the initial level can definitely reflect at the IOI results (that is why we introduced qualification round and involve students from seventh and eighth grade in high school competitions). Another big problem is a lack of the materials for the beginners. There are only few books of this type in Serbian. Collections of tasks with full analysis are very hard to find. This will enable us to train students for a longer period of time, which is the main point.

References

Cepeda, A., Garcia, M. (2011). Mexican olympiad in informatics. *Olympiads in Informatics*, 5, 128–130.

Combefix, S., Leroy, D. (2011). Belgian olympiads in informatics: the story of launching a national contest. *Olympiads in Informatics*, 5, 131–139.

Italiani, M. (2011). Italian olympiad in informatics: 10 years of the selection and education process. *Olympiads in Informatics*, 5, 140–146.

Kelevedjiev, E., Dzenkova, Z. (2009). Tasks and training the intermediate age students for informatics. *Olympiads in Informatics*, 3, 26–37.

Koivisto, J. (2011). The national computer olympiads and the IOI participation in Finland. *Olympiads in Informatics*, 5, 147–149.
Merry, B., Gallotta, M., Hultquist, C. (2008). Challenges in running a computer olympiad in South Africa. *Olympiads in Informatics*, 2, 105–114.
Malaivongs, K. (2011). Preparing students for IOI: Thailand country report. *Olympiads in Informatics*, 5, 150–154.
Pankov, P.S., Oriskulov, T.R. (2011). Kyrgyzstan national report on olympiads in informatics. *Olympiads in Informatics*, 5, 155–160.
Verhoeff, T. (2011). 20 years of IOI competition tasks. *Olympiads in Informatics*, 3, 149–166.
Yovcheva, B., Momcheva, G., Petrov, P. (2009). jBOI – one more possibility for increasing the number of competitors in informatics. *Olympiads in Informatics*, 3, 167–173.
Zur, E., Benaya, T., Ginat, D. (2010). IOI Israel – team selection, training, and statistics. *Olympiads in Informatics*, 4, 151–157.

Appendix

Problem: The Hamming distance

Here is an example of the problem from the **National competition 2011.** This was the second problem for students in division B. The average number of points for this problem was 26.3 (out of 100) and nine contestants solved this problem (out of 85).

Statement. The Hamming distance of two integers a and b is the number of bits in binary representation at which these numbers differ. For given n integer numbers $a[1], a[2], \ldots, a[n]$ calculate the sum of Hamming distances for all pairs of numbers, i.e.,

$$\sum_{i<j} \mathit{ham}(a[i], a[j]).$$

Input. In the first line of the input is the number of elements n ($1 \leqslant n \leqslant 100,000$), and in the second line are n integer numbers $a[i]$ ($1 \leqslant a[i] \leqslant 2,000,000,000$).

Output. In the first line of the output print the sum of the hamming distances among all pairs of numbers $a[i]$.

Sample input	**Sample output**
4	14
1 9 4 10	

Explanation. The Hamming distances for all pairs of the given array are:
$\mathit{ham}(1, 9) = 1$, $\mathit{ham}(1, 4) = 2$, $\mathit{ham}(1, 10) = 3$, $\mathit{ham}(9, 4) = 3$,
$\mathit{ham}(9, 10) = 2$, and $\mathit{ham}(4, 10) = 3$. The sum of these Hamming distances is 14.

Naive solution. Traverse all pairs of numbers and calculate the Hamming distance using bitwise operations – one can do this in many ways: using bitwise operators, strings, arrays (bitwise operators are fastest). This solutions runs in $O(n^2 \log M)$, where M is the largest number among $a[i]$'s.

```
unsigned int sum = 0;
for (int i = 0; i < n; i++)
    for (int j = i + 1; j < n; j++)
    {
        unsigned int x = a [i] ^ a [j];
        while (x > 0)
        {
            sum++;
            x = x & (x - 1);
        }
    }
return sum;
```

Optimal solution. Note that each bit is independent. For ith bit, $0 \leqslant i < 32$, let it count represents how many numbers have 1 on ith position. Then on the aggregate sum, we just add *count* $\cdot$ $(n -$ *count*$)$. The complexity of this algorithm is $O(n \cdot \log M)$, since we traversed the array a exactly $\log M$ times.

```
unsigned int max_bit = 32;
unsigned int sum = 0;
unsigned int pow = 1;
for (int i = 0; i < max_bit; i++)
{
    unsigned int count = 0;
    for (int j = 0; j < n; j++)
        if ((a [j] & pow) != 0)
            count++;
    sum = sum + count * (n - count);
    pow = pow << 1;
}
```

Problem: Division Query

Here is an example of the problem from the **Serbian IOI Team Selection competition 2011.** The average number of points for this problem was 18.46 (out of 100) and, unfortunately, none of the contestants had maximal number of points. For this problem, because of its difficulty, we present only the main idea for the optimal solution.

Statement. You are given an integer array a of the length n. There are two types of operations that you can perform on this array:

- $1\ i\ x$ – sets ith element of the array to x ($a[i] = x$);
- $2\ i\ k$ – return the longest subarray (consecutive elements) of the array a such that it contains element $a[i]$ and all elements from it are divisible by k. This length can be equal to zero.

There are q operations of the above types in the input. Write a program that preforms these operations and for every query (an operation of the second type) outputs the corresponding result.

Input. In the first line of the input there are two numbers n and q ($1 \leqslant n,\ q \leqslant 200,000$) which represent the number of elements in the array a and the number of operations, respectively. The next line contains n positive integers, separated by one space –

the elements of the array a. The next q lines describe the operations. Values of the elements in array a and the value for k in queries are from 1,000,000,000.

Output. For every query in the input, print the longest subarray with the described property (in the same order as from the input).

Sample input	**Sample output**
6 4	3
2 25 20 30 19 5	1
2 3 5	5
2 4 6	
1 5 100	
2 3 5	

Remark. In 40 percent of the test cases, the number k for all queries in the input will be the same.

Naive solution. Trivial solution is to preform each query by traversing all elements, starting from $a[i]$ in both directions, and checking whether they are divisible by k. When we find the first element that is not divisible by k, we can stop. The complexity of this operation is $O(n)$, so overall complexity is equal to $O(n \cdot n)$. This solution gets only 10 points.

Subproblem solution. If the number k is equal for all queries, we can maintain an additional array *mark* such that *mark*$[i] = 1$ if k divides $a[i]$; otherwise *mark*$[i] = 0$. Now, we can translate the query for the ith element as follows: find the longest subarray of 1's containing the element with the index i. It suffices to find first-left and first-right element of the array mark, starting from ith, with value 0. This can be implemented with binary search and cumulative sums in $O(\log^2 n)$ per operation. Faster $O(q \cdot \log n)$ solution can be obtained by exploiting binary structure of cumulative table or disjoint set data structure rather than using binary search.

Optimal solution. For the optimal solution we are going to use a variation of the segment tree data structure. The structure is represented as a complete binary tree with the root node 1. The leafs correspond to the elements from the array a, starting from left to right in the last level. By definition, it follows that the number of nodes in segment tree is at most $2n$. For every node v we are going to store the greatest common divisor of values in its subarray. This way, when the element $a[i]$ is set to new value x, we are going to change only $\log n$ values in the segment tree. Using the property that every subarray can be covered with at most $\log n$ different nodes, the total complexity of this algorithm is $O(q \cdot \log^2 n)$, where we have $\log^2 n$ because of the complexity for finding the greatest common division and as every subarray can be covered with at most $\log n$ nodes of binary tree.

A. Ilić (1984) holds a PhD in computer science from Faculty of Science and Mathematics, University of Niš from 2011. He published over sixty papers on combinatorial and spectral graph theory and design of algorithms. From July 2011 works as a software engineer in Facebook on friends ranking and optimization. Aleksandar represented Serbia as a contestant on two IOIs and two IMOs wining two silver and two bronze medals, and wrote problems for IOI 2006, JBOI 2007, IOI 2008, BOI 2011, HackerCup 2012. He was the main organizer of international high school mathematics competition "Math Kangaroo" in Serbia and the first Junior Balkan Olympiad in Informatics for elementary school students in 2007.

A. Ilić (1987) is a final year student at Faculty of Sciences and Mathematics, University of Niš. Since 2007 Andreja is a member of the Serbian Committee for Competitions in Informatics and the main organizer of many training camps of Serbian national team. He won more than twenty medals as competitor in informatics and mathematics on national and international olympiads, representing Serbia. He wrote problems for BOI 2012, BubbleCup 2010, BubbleCup 2011. His areas of research are spectral graph theory, social networks and machine learning.