Agnes Erdosne Nemeth, Laszlo Zsako

Online Training and Contests for Informatics Contestants of Secondary School Age

2015

MUZEUM HISTORII POLSKI

Ágnes Erdősné NÉMETH, László ZSAKÓ

Eötvös Loránd University, Hungary

Online Training and Contests for Informatics Contestants of Secondary School Age

Introduction

Like in the 15th's century, when the books changed the world and the books became the tool of information nowadays the digital communication and the computers are the main source of knowledge. The new generation is the part of computers' world and we need to use this fact. Nowadays the processing of information, modeling and making interactions is becoming the main goal of education. The classical classroom and teacher model is changing to individual learning with the help and instruction of the teacher. The rivalry dropped out of the classroom to global, and communication has become permanent in all subjects, but especially in informatics.

The role of the teachers is changing: they have to be up-to-date and able to offer a wide range of learning methods and materials to the children. A lot of practice is crucial for the successful training of informatics contestants as well as a very large number of interesting, creative and personalized series of tasks. Teaching children of many different algorithmic and programming levels you face to have very few resources to make new exciting problems to everybody. They need to learn new techniques and algorithms, practice a lot and they need a lot of motivation.

Making the unique, interesting, appropriate-level new task with test cases against all possible mistakes lasts for days, sometimes a week, while selecting a task from a well-structured and reliable problem set with immediate evaluating and online test cases takes just takes a few hours.

It is a good way to recommend your students websites with online materials:
− show them the way of individual learning online and check their work regularly or
− recommend them carefully selected online training sites' problems or
− encourage them to take part in online contests.

To do this you have to know which site is good for practicing a given area on the appropriate level and motivates your students as well. Some sites are good for beginners and for national level contestants, quite others for IOI-level students and for those who are ready for ACM ICPC.

Nowadays children live in permanent online communication and use online social connections regardless of where they live in the world. The additional advantage of online materials is to motivate your students as they see codes and ranks of others and can communicate with them even from the opposite side of the world if they have the same field of interest.

Great advantage of the online materials is not being tied to lessons and being able to use them in students' spare time depending on their mood.

Related work

The first computer science contests classification was approached by W. Pohl [2006]. His classification was applied and supplemented [Combefis 2014] for IPSC, ACM-ICPC, IEEEXtreme, IOI contests and online contests of Google Code Jam, ProjectEuler and Codechef. N. Ragonis [2012] explored the large variety of questions proposed in the main programming contests and classified them. G. Garcia-Mateus [2009] discussed that programming contests also help make learning programming fun. M. Forišek [2013] made a succinct programming contests overview focusing on contests to find efficient algorithms.

Aim of this article is to complete the classification with new aspects and to perform the new classification of some online contests and training sites. It concentrates on the needs of secondary-school-aged informatics contestants of the IOI-style: finding the most efficient algorithm and modeling challenging problems and final work is writing a computer program to solve it on the given test cases.

The new aspects arise from the changing role of the teacher and the changing needs of students. So that the teacher can suggest tasks to be solved to practice and advises motivating contests in the preparation process, he should know exactly what he can expect from the current online material.

When you, as a teacher suggest a material you should know what algorithms you want your student to practice, on which level your student is, which step of program development process you want to strengthen. For example if you teach a new algorithm on some level you can choose which site offers you tasks of the appropriate level, on that site you can choose the appropriate type of tasks without the large amount of work of creating them. If you are teaching the making of test cases for checking the correctness of a program you have to know on which sites the test cases can be viewed.

Classifying criteria
Previously defined and used criteria

From Pohl [2006] classifying dimensions this article concentrates only on

Scientific area	Algorithms
Style	IOI-style: finding the most efficient algorithm and writing a computer program to solve it

Duration   Short time (exam-style) or long time (homework-style)
Grading   Automatic or manual via test cases
Submission   Executable program or code or output file
Divisions   High school age students.

That article describes just only the contests. These actual values of the contests classification are the main scope of this article.

Using online materials for teaching and learning algorithms and to practice implementing the code needs more detailed classification as described in the S. Combefis and J. Wautelet [2014] classification. Their first set of criteria is general information about the contests:

I1   Team   Single or team
I2   Age & gender   Age range of contestants
I3   Language   Accepted programming language
I4   Duration   Contest duration to submit solutions
I5   Frequency   Contest frequency or it is available any time (open)
I6   Scoring   A score calculating method.

The second set of criteria is the criteria of the tasks:

T1   Submission   Code source, executable program, output data
T2   Type   Solving a problem
T3   Limitations   Time, memory, the number of submissions
T4   Feedback   Submissions feedback
T5   Level   Difficulty level or partitioning of the tasks

Additional criteria to general information of contests
The supplement of the contests' general information criteria:

I7   ranking

Whether the score of the actual contest is integrated in the site's ranking or not. Some of the sites only have separate contests and the score of the current one doesn't matter in any ranks (or only counts in the background in the national or group ranking system outside the webpage) (USACO, COCI).

Some sites have ranking lists based on the number of problems solved or competitions completed: the more tasks you have solved or the more contests you have taken part in, the higher rank you earn. On these sites the ranks are categorized from the beginner through advanced till the grandmaster (Project-Euler, SPOJ).

There are some sites where the actual ranking score is calculated with a complicated formula based on your previous rank and the performances of the contestants of the same level. In this case the rank may increase or even decrease (Timus Online Judge, Codeforces)

There are sites on which you can step up to the next level group and get higher level tasks only if you have reached a certain result (USACO, Codeforces Div1&2).

The ranks can motivate your students, give them better self-knowledge and show them the way of self-improvement and the way of being better.

I8 task language

If you want to use a contest for students who are younger or speak very few languages you have to know in which languages the tasks are written. Maybe your students need translation or they need some clarification if they don't understand the task exactly. It means that you as a teacher have to sit in front of the computer during the whole contest and help the students with the translation. However, for a fair competition you can't help them to solve the tasks or give them any idea of the method of solution.

Additional task criteria of contests:

T4 the feedback criteria need some identification and detailed specification

The simplest feedback is correct or failed after running all the test cases or just one test case written in a task description. Even if only one test case is failed then the feedback says not accepted. (COCI, USACO)

More detailed feedback is whether the program is good or bad with runtime or compiling error. (SPOJ)

The next category of feedback is an IOI-style feedback, when the test cases are in groups. In every group there are varied test cases. The group can be accepted or – even if one test is failed from the group – the whole group is failed.

The most detailed feedback helps to find and correct an error. It is detailed for every single test case: accepted (AC), compile error (CE), wrong answer (WA), time limit exceeded (TL), memory limit exceeded or run-time error (EX). The run-time error may be detailed: sigkill, illegal reference, non-zero exit code. (CodeChef)

The supplement of the task criteria:

T6 test cases availability

Some contests have single test cases and every test case has its own value.

If the task can be divided into several parts by the size of the input then the small test cases have a previously specified percentage of the points and the larger test cases have another given percentage of the whole task's points. The test cases adjust to these subtasks. (CodeChef, COCI, UVa)

T7 description of the wanted algorithm

Detailed description of the correct algorithms is available after the contest or not.

T8 code visibility and sample code

Sample code is available for everybody after the contest and other contestants' source codes are available or not. If you are learning programming or a new programming language it is a good strategy and very useful to examine others' source code.(COCI, SPOJ, CodeChef)

T9 predictability of the level of the contest

If the teachers want to recommend a contest to their students it is important that they face tasks of the appropriate level. Furthermore the level of the contest can be predictable or may be chaotic.

Additional criteria from the teacher's point of view:

If you want to use online materials not only for checking your students' knowledge but for teaching algorithms or practicing as well then a lot of new criteria are important for you.

P1 tasks outside the contests

The tasks for practicing have to be downloadable or available online any time, 24 hours a day, 7 days a week. Are the tasks of the contest available for practicing after the competition? How many tasks are available without a contest?

P2 categorized tasks

If the tasks are categorized according to difficulty level and to the expected algorithm type then it's easier to choose good, for-your-students-tailored exercises. (UVa, SPOJ, USACO)

P3 online judge or offline test cases are available.

It is easy and you can serve a large number of students with a practicing environment if you can upload a source code or solution and an online judge evaluates the submission. (SPOJ, UVa, CodeChef) It's more complicated if they can download test cases and test solution offline individually. In that case the checking of the expected work is complicated too for you as a teacher. (COCI)

P4 availability of detailed test cases.

If the exact and detailed test cases are available you can show your students the process of creating them too. When the number of submissions is limited or there is a penalty for uploading a wrong submission on a contest it is very important for the students to test solution offline before uploading. To test their own solution they have to make their own test cases. The students must be taught about it and some tests must be observed before doing their own test cases. If you can analyze test cases online or offline you can teach them how to analyze a problem and their own code. (COCI, USACO)

P5 learning material

Some of the students like to discover the world of programming alone at their own pace and want to discuss only if they have some problems with the understanding or the coding. Some of the online sites offer distance learning material with series of exercises. (USACO, UVa)

P6 limitations of practicing tasks

As in the contests the tasks for practicing have time limits, memory limits and sometimes limit of number of the submission or penalty for wrong submission. May be the tasks are available only if you submit the previous level tasks (USACO).

P7 language of tasks

Sometimes the number available languages of the description of the task in practice is larger than on contest. If your students speak only their mother tongue than the tasks' availability in that language is a very important question. Even if the tasks are not available in student's language while making exercise you can translate them and take the advantages of automatic submissions, rank and carefully selected test cases.

P8 programming language

If you want to teach one specific programming language or your students are in conversion from one programming language to another one, you have to know whether the new one is accepted or not.

P9 algorithm's description and sample code

Besides the distance learning material it is important and instructive to look at the problem setter's vision, the best algorithms and program codes. Some sites offer the possibility of sending in your algorithm or your test cases too.

P10 communication

The children like talking and asking about problems. It is important for them to have a forum to create a space for asking, talking about the methods or clarifying tasks. The forum makes it easy to contact other people who have the same problem. The editorial means that the site's operators can send messages, can clarify tasks and write the implementation of the most efficient code and algorithm. Some site has FB page or send a regular newsletter.

P11 teacher's supervision

If you teach students and recommend them a site for competition or practicing then you want to observe and follow up their work. Some sites offer forming groups from your students and check their performance, allow you to download their solutions for discussing. (ProjectEuler) Sometimes you can see your students work by their nickname on the site. (SPOJ, CodeChef)

Some site's classification and future work

The classification of seven sites with this system of criteria is available at www.microprof.hu/classification:

CodeChef: http://www.codechef.com/

Croatian Open Competition in Informatics: http://hsin.hr/coci

Project Euler: https://projecteuler.net/

Sphere online judge: http://spoj.pl/

Timus online Judge: http://acm.timus.ru/

UVa Online Judge: http://uva.onlinejudge.org

USA Computing Olympiad: http://www.usaco.org/

More sites will be classified in the near future.

Conclusion

The classification helps you as a teacher to choose contests for your students depending on what prior knowledge is required, in which language it can be accessed, what types of problems a contest deals with and at what level. Using online materials you can offer your students a very large number of interesting, creative and personalized tasks without the incredible amount of work to create the tasks and test cases. To make a well-tailored personalized series of tasks the classification helps you which sites offer you tasks corresponding to your current educational goals. The classification tells you on which site you can get ideas for solutions or see detailed test cases or the results of your students. On-line contests can also be categorized regarding whether students can see each other's solutions; whether algorithms corresponding to the running time and memory limits will be known, and whether students are given inspiration adequate to their own learning style.

Literature

Combéfis S., Wautelet J. (2014): *Programming Trainings and Informatics Teaching Through Online Contests*, „Olympiads in Informatics" vol. 8.

Cuba-Ricardo G., Leyva-Figueredo P.A., Mendoza-Tauler L.L. (2014): *Learning Strategies of Informatics Contestants*, „Olympiads in Informatics" vol. 8.

Forišek M. (2013): *Pushing the Boundary of Programming Contests*, „Olympiads in Informatics" vol. 7.

Garcia-Mateos G., Fernandez-Aleman J.L. (2009): *Make Learning Fun with Programming Contests*, Transactions on Edutainment II. Lecture Notes in Computer Science 5660.

Hiron M., Février L. (2012): *A Self-Paced Learning Platform to Teach Programming and Algorithms*, „Olympiads in Informatics" vol. 6.

Pohl W. (2006): *Computer Science Contests for Secondary School Students: Approaches to Classification*, „Informatics in Education" vol. 5.

Ragonis N. (2012): *Type of Questions – The Case of Computer Science*, „Olympiads in Informatics" vol. 6.

Abstract

If you prepare students for programming contests carefully selected and widely available online training and contests offer help and diversity. If you teach about testing programs you have to know which sites offer downloadable tests or feedback with detailed test cases. If you want to make series of tasks for practicing you have to know which sites offer you categorized tasks of the appropriate level. In order to be able to choose from the available materials we need to categorize them. The previously defined and used criteria need some supplement criteria for better and sophisticated use of categorization from the teacher's point of view. Online resources can be classified in general: what programming languages can be used, how often the contests are organized, in which

languages they can be accessed, what types of problems a website deals with and at what level, what prior knowledge is required. We can group sites according to whether they help teachers to set tasks for their students, or get ideas for solutions or see the results of their students. Online contests can also be categorized regarding whether students can see each other's solutions. The aim of this paper is to supplement the categorization and describe some major portals according to the previously defined and supplemented criteria.