

Analyzing Programming Contest Statistics

Shahriar Manzoor

Southeast University, Dhaka, Bangladesh

Problem setter and System Administrator

ACM Valladolid Online Judge

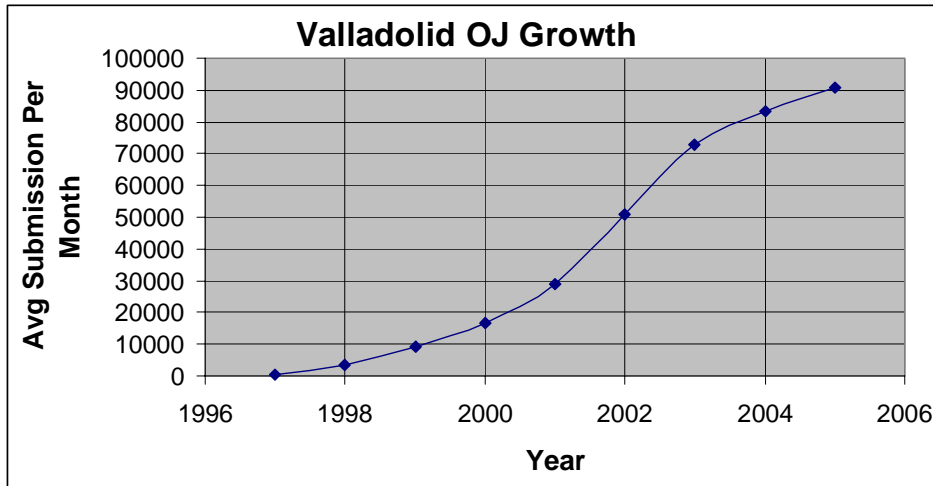
(<http://online-judge.uva.es/p>)

In this paper we will try to analyze a large database of submission of contestants from different parts of the world and provide some important results which was never presented before in this manner and at this scale. These results may enable us to find out what aspects of programming contests need to be changed to make it more meaningful, how practice and experience improves the performance of a contestant and it will also create new openings on which we can continue our study in future. It will also help us to identify the areas which are not lightened by programming contest and then we can take initiatives for those regions.

Introduction:

Programming contest is probably the fastest expanding co-curricular activity related to computer science. The main reason is because arranging a programming contest does not require too much change in infra structure. Generally each type of programming contest has an annual international event, but teams for these contests are selected via many preliminary contests which takes place in national level, regional level or sometimes as online contests. While ICPC (ACM International Collegiate Programming Contest), IOI (International Olympiad for Informatics) and Topcoder (www.topcoder.com) are the three most prominent programming contests of the world, there are some other web sites which provide adequate training materials for these contests. These web sites are known as Online Judges to contestants and most of these sites are prepared based on the rules and restrictions of ICPC – the largest programming contest for University Students.

Some well known online judges are ACM Valladolid Online Judge or UVaOJ (<http://acm.uva.es/p>), Zhejiang University Online Judge (<http://acm.zju.edu.cn/>), Saratov State University Online Contester (<http://acm.sgu.ru>) and Ural State University Problem Set Archive (<http://acm.timus.ru>). Of these sites the Valladolid Online Judge is the oldest Online Judge and arguably the most popular Online Judge as well. It started in 1997 and since then it has become more and more popular keeping pace with the popularity growth of programming contest throughout the world. The following chart shows how the submission per month (on average) has grown in UVa judge from its beginning. This site has received about 4089635 submissions from different corners of this world from its beginning till October, 2005. We will use this huge number of submissions to analyze different aspects of programming contest. To imagine how huge this number of submissions is, let's assume there are 8 problems and 100 teams in a programming contest and all the teams solve all problems with 4 submissions per problems. So the total number of submissions is $100 * 8 * 4 = 3200$. So we can assume (relaxed assumption) that a five hour programming contest has 3200 submissions in total. So the number submission of UVa OJ is equivalent to $4089635 / 3200 \sim 1278$ real-time programming contests. Before analyzing the huge data, we will give you a short introduction on online judge.



What is UVa Online Judge?

An online judge is in general a server which contains problem description of different contests. The judge also contains the judge data to judge most of the problems. All the problems have an unique ID. Any user from anywhere in the world can registrar himself with an online judge for free and solve any problem he likes – he just has to mention the problem ID and his own membership number while submitting the problem. Most online judges allow its users to submit solutions in at least three languages C/C++, JAVA and PASCAL. Although C and C++ are two different languages but it is hard to distinguish between them because they are supported by the same compiler, they have many things in common, contestants use both the features of C and C++ when it is more convenient and many contestant codes and C and submits them as C++ program to avoid some type conversion complications of C. For example some contestants use STL, codes in C++ but uses printf() function to produce output as output formatting is much easier with printf() function.

UVa online judge is one such online judge whose main responsible person Prof. Miguel A. Revilla has received DeBlasi Award from ACM for its outstanding contribution in popularization and training of programming contest. It has about 1800 problems to solve and about 64000 users world wide. It is also mentioned before that the site has received around 4 million submissions up to October 2005, which are the data for our analysis. When a user submits his solution for a problem, the online judge tests it with an specified input and matches its output with the specified output to test its correctness. Of course some problems have more than one solution and to judge those problems some special judge programs are used. According to the outcome of this test the user is given any one of the following responses

Table 1

Short Notation	Meaning	Detailed Meaning	Percentage
AC	Accepted	The output of the program matches correct output.	29.33%
PE	Presentation Error	The output is correct but it produces some extra space or blank line.	5.01%
WA	Wrong Answer	The output of the program does not match the correct output.	34.92%

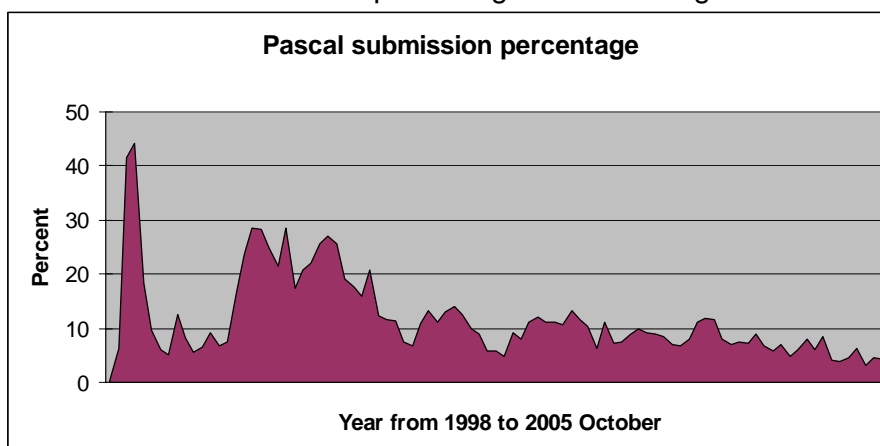
TL	Time Limit Exceeded	The program does not terminate within the specified time limit.	7.83%
CE	Compile Error	The program does not compile with the specified language's compiler.	9.39%
RE	Runtime Error	The program has crashed.	7.58
SE	Submission Error	The user ID, Problem ID is wrong.	3.06
ML	Memory Limit Exceeded	The program requires more memory to run than what the judge allows.	0.81
OL	Output Limit Exceeded	The program produces more than 4 MB output within the time limit.	0.93
RF	Restricted Function	The program uses some system function call or tries to access files.	0.81
Others	Uncommon verdicts	Some rare verdicts produced by the judge which are not so important.	0.33

The last column shows which error occurs at what percentage. So the percentage of "accepted" verdict is 29.33% means of the total verdicts produced by the judge 29.33% are accepted. In the next section we will see what causes can effects this rate of acceptance, wrong answer etc. Another interesting things from this table is that the principle verdicts produced by the online judge are Accepted (AC), Wrong Answer (WA), Compile Error (CE), Time Limit Exceeded (TL), Run Time Error (RE) and Presentation Error (PE). So in this article we will focus more on these five types of errors. Also these five errors are more common in realtime contests. Errors like SE, ML, OL, RF etc are generally not considered in realtime contest or considered within the primary five errors. For example in most contests Memory Limit Exceed is considered within Compile Error, Output Limit Exceeded is considered within Time Limit Exceeded or Wrong Answer.

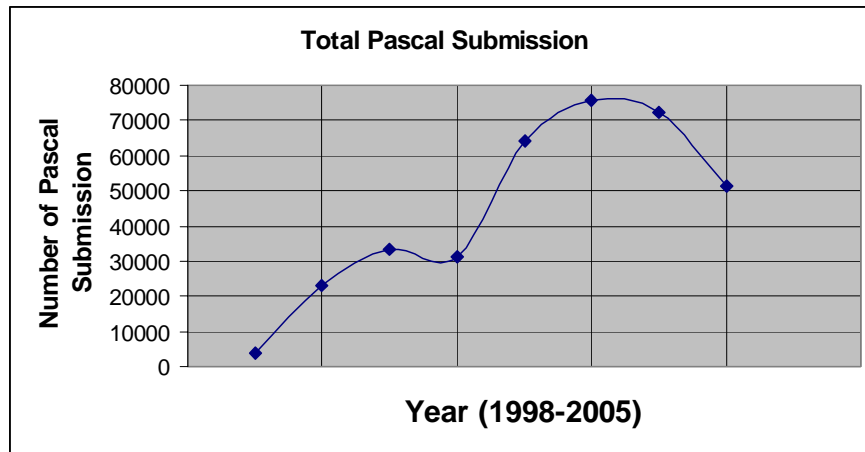
In the next section we will consider different aspects of programming contest based on the huge submission history of UVa Online Judge. As the submission history of UVa Online Judge is so big that these can actually represent the general characteristics or errors, successes in a programming contest.

Is PASCAL losing its popularity among contestants?:

The following graph shows how much of the total submission is actually pascal submission in UVa Online Judge from start of 1998 to October 2005. The graph shows that the submission percentage in PASCAL language changes abruptly with time but it also shows that submission percentage is decreasing.



The percentage of submission is decreasing but is the number of submission decreasing?



The graph above shows that the number of submission is also increasing but is it enough to drop Pascal from different contests? Is this drop in submission indicates that lesser people like Pascal or because Pascal is not being used in different programming contests remains so people are being discouraged to use it? All these remain in front of us as food for thought. We also have plan to produce a map that shows the regions where Pascal is still popular.

How does practice change things?

The table below shows the error rate of people with different experience.

Table 2: Based on all problems

Authors considered	AC Rate	WA rate	TL Rate	CE Rate	PE Rate	RE Rate
All Authors	29.33 %	34.92 %	7.83 %	9.39 %	5.01 %	7.58 %
Less than 50	22.15 %	34.77 %	7.99 %	16.24 %	4.85 %	6.96 %
At least 50	31.75 %	34.97 %	7.77 %	7.07 %	5.06 %	7.79 %
At least 75	32.31 %	34.92 %	7.68 %	6.75 %	5.02 %	7.77 %
At least 100	32.74 %	34.87 %	7.62 %	6.54 %	4.98 %	7.76 %
At least 125	33.08 %	34.81 %	7.58 %	6.37 %	4.94 %	7.75 %
At least 150	33.36 %	34.78 %	7.56 %	6.24 %	4.89 %	7.72 %
At least 175	33.60 %	34.75 %	7.52 %	6.15 %	4.85 %	7.69 %
At least 200	33.81 %	34.73 %	7.49 %	6.08 %	4.80 %	7.68 %
At least 225	34.01 %	34.68 %	7.47 %	6.01 %	4.75 %	7.67 %
At least 250	34.18 %	34.67 %	7.46 %	5.94 %	4.71 %	7.65 %
At least 275	34.33 %	34.66 %	7.44 %	5.89 %	4.68 %	7.65 %
At least 300	34.48 %	34.66 %	7.43 %	5.83 %	4.64 %	7.63 %
At least 325	34.62 %	34.62 %	7.42 %	5.78 %	4.61 %	7.62 %
At least 350	34.75 %	34.62 %	7.40 %	5.72 %	4.58 %	7.61 %
At least 375	34.88 %	34.62 %	7.38 %	5.68 %	4.54 %	7.60 %
At least 400	34.99 %	34.61 %	7.37 %	5.64 %	4.51 %	7.60 %
At least 425	35.10 %	34.58 %	7.37 %	5.62 %	4.48 %	7.59 %
At least 450	35.19 %	34.58 %	7.36 %	5.59 %	4.46 %	7.58 %
At least 475	35.27 %	34.58 %	7.35 %	5.54 %	4.44 %	7.58 %
At least 500	35.35 %	34.57 %	7.34 %	5.50 %	4.42 %	7.59 %
Change from less than 50 to 500	+13.2 %	-0.2	0.49	-10.74	-0.43	0.63

The above table indicates that with practice acceptance rate increases a lot and also compile error decreases a lot but surprisingly wrong answer percentage does not change. So does this indicate no matter how experienced you are you can always get wrong answer?

The above table has a problem, as people solve more problems they have less easy problems to solve (assuming that people tend to solve easy problems first) so when someone has already solved 500 problems he has no more easy problems to solve and so his acceptance rate can go down a little but as he is more experienced the acceptance rate does not go down but it remains similar.

In the table below we have put the same results but this time only based on the problems with low acceptance rate.

Table 3: Based on problems with low acceptance rate

Authors considered	AC Rate	WA rate	TL Rate	CE Rate	PE Rate	RE Rate
All Authors	15.78 %	41.77 %	13.53 %	8.77 %	1.97 %	11.68 %
Less than 50	9.41 %	40.33 %	14.60 %	15.42 %	1.59 %	10.78 %
At least 50	17.17 %	42.08 %	13.30 %	7.32 %	2.05 %	11.88 %
At least 75	17.63 %	42.11 %	13.08 %	7.05 %	2.07 %	11.88 %
At least 100	18.02 %	42.13 %	12.92 %	6.86 %	2.08 %	11.84 %
At least 125	18.35 %	42.13 %	12.84 %	6.69 %	2.09 %	11.77 %
At least 150	18.61 %	42.11 %	12.75 %	6.59 %	2.09 %	11.72 %
At least 175	18.84 %	42.13 %	12.64 %	6.51 %	2.10 %	11.66 %
At least 200	19.03 %	42.15 %	12.54 %	6.46 %	2.09 %	11.62 %
At least 225	19.22 %	42.11 %	12.49 %	6.40 %	2.09 %	11.59 %
At least 250	19.38 %	42.14 %	12.43 %	6.34 %	2.09 %	11.55 %
At least 275	19.55 %	42.17 %	12.34 %	6.29 %	2.08 %	11.52 %
At least 300	19.72 %	42.16 %	12.31 %	6.25 %	2.07 %	11.48 %
At least 325	19.87 %	42.07 %	12.29 %	6.21 %	2.07 %	11.48 %
At least 350	20.00 %	42.10 %	12.23 %	6.16 %	2.06 %	11.44 %
At least 375	20.15 %	42.14 %	12.15 %	6.13 %	2.06 %	11.39 %
At least 400	20.27 %	42.13 %	12.11 %	6.10 %	2.06 %	11.36 %
At least 425	20.39 %	42.14 %	12.08 %	6.07 %	2.03 %	11.33 %
At least 450	20.51 %	42.14 %	12.02 %	6.04 %	2.03 %	11.32 %
At least 475	20.61 %	42.11 %	11.97 %	5.99 %	2.04 %	11.32 %
At least 500	20.70 %	42.09 %	11.94 %	5.94 %	2.05 %	11.35 %
Change from less than 50 to 500	11.29%	1.76%	-2.66	-9.48	0.46	0.57

The most interesting result of this table is that the wrong answer rate and run time error percentage is actually less when you are inexperienced for problems with low acceptance rate.

Another table with problems of high acceptance rate will be added. Also we have plans to produce such tables for easy problems and hard problems.

If we can generate such tables for different types of problems then we can design problemset that will give experienced coders less or more advantages (the one that this required). Also all these can make programming contest more interesting in future. If we want to give rating to contestants, find out the probability of their winning based on the problemset and previous history these types of statistics can help. Also coaches can find from these statistics whether his team is performing better than average or less than average. One problem is that all these results are based on 24-hour judge, this can be a lot different in real-time contests. So in future we may produce results based on UVa online contests.

Drop out rate!!!:

Programming contest is not something that is very easy and generally people with average intelligence struggle in this field. It is quite impossible to find out how the drop out rate of programming contest (How many people hopes to participate in programming contest but then never does so). Of the 65000 users so far in Valladolid Site we have found 739 people who have submitted 20 or more times but have failed to get one problem accepted. People who have got something accepted probably would have learnt something, but these people have left the arena without probably learning anything or probably learning something bitter (They are not fit for contest). There is one person who have 2945 times and has not got anything accepted and another person who has submitted 540 times but have not got anything accepted. But these are very extreme cases and there are not too many like this.

Where is programming contest more popular?

With the nine years submission history of UVa we have found which regions have used this site more, and we can safely assume that the people of regions which uses this site more like programming contest but the vice versa may not always true. We have defined the acceptance ratio for a country C as:

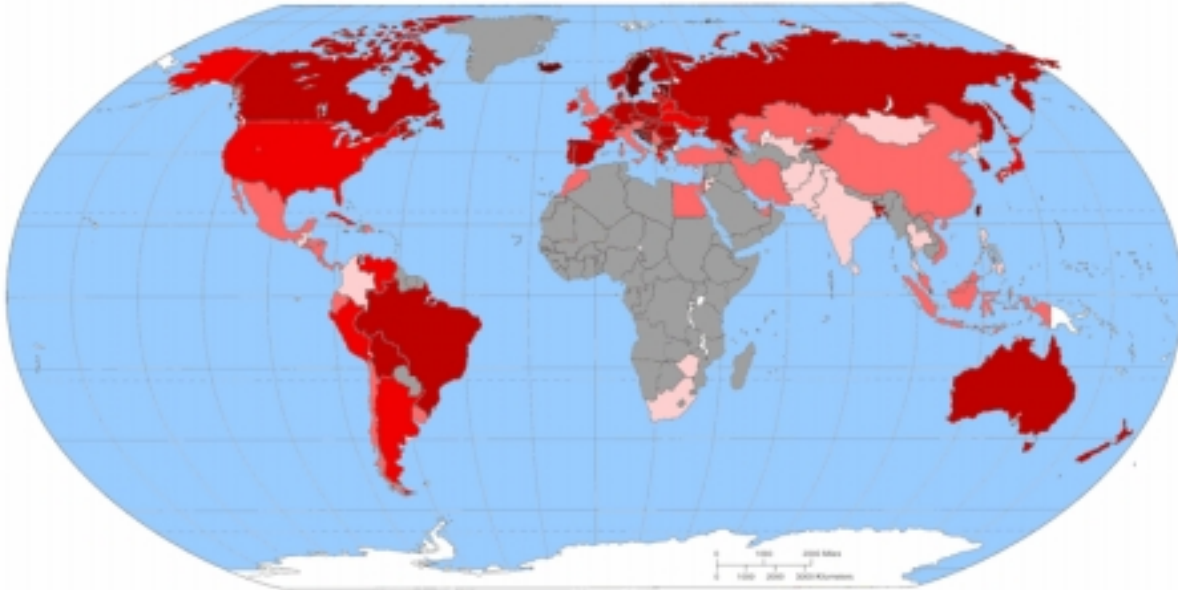
$$\frac{\text{Number of submissions from country C}}{\text{Population of Country C in million}}$$

According to this ratio some of the countries where programming contest is most popular are:

Table 5

Rank	Country	Ratio	Rank	Country	Ratio
1	Hong Kong	30524	21	Hungary	2404
2	Iceland	29610	22	Latvia	2303
3	Taiwan	27004	23	Slovak Republic	1991
4	Estonia	21193	24	New Zealand	1978
5	Singapore	14272	25	Spain	1870
6	Macedonia	8990	26	Bolivia	1863
7	Slovenia	8843	27	Czech Republic	1691
8	Armenia	7143	28	Australia	1665
9	Croatia	6778	29	Romania	1497
10	Sweden	5362	30	Netherlands	1253
11	Portugal	4544	31	Brazil	1170
12	Poland	4515	32	Finland	1084
13	Switzerland	4230	33	Germany	1050
14	South Korea	4103	34	Russian Federation	1037
15	Bosnia-Herzegovina	3797	35	United States	945
16	Bangladesh	3553	36	Venezuela	892
17	Norway	3316	37	Greece	869
18	Bulgaria	3185	38	Cuba	698
19	Kyrgyz Republic	3155	39	Lithuania	662
20	Canada	2961	40	Belarus	660

The above list is long and it makes you feel good but there are many countries whose citizens have never submitted to Valladolid Site. Some countries who have never submitted to Valladolid Site but has more than 5 million population are: Haiti (8.1), Cambodia, Kingdom of (13.1), Rwanda (8.4), Papua New Guinea (5.7), Mali (13.4), Niger (12.4), Guinea (9.2), Congo, The Democratic Republic of the (58), Syria (18.0), Tadjikistan (6.6), Madagascar (17.5), Malawi (11.9), Laos (5.8), Cameroon (16.1), Ethiopia (72.4), Sierra Leone (5.2), Angola 13.3. And it is obvious that the Subcontinent Africa is mostly out of touch of programming contests. We are making a world map that will indicate the growth of programming contest in different parts of the world. But it is still not finished. The almost finished map is below.



The darker the red the more is the popularity. They grey countries have either less than 10 members in UVa or they have never submitted to UVa.

What type of errors do contestants make most frequently?:

Suppose one submits a program in a contest and gets accepted, another contestant submits a program he gets wrong answer and then he submits again he gets accepted, another contestant submits a program six times and every time he gets wrong answer. Which one of these three events is more likely to happen in a programming contest? To find it out we analyzed all the submissions of UVa site and found out which are the most common response sequence for a contest. We actually took a method like digraph, trigraph analysis of a text. First we tried to analyze which submission response is most common for a problem. And the most frequent response sequences are given in the tables below:

Table 6: A table for most popular response sequence

Monograph	AC	WA	CE	TL	PE	RE
Frequency	541518	334946	109977	78434	74882	65046
Di-graph	WA WA	WA AC	CE CE	TL TL	RE RE	CE WA
Frequency	167499	74206	40860	31178	27800	20089
Tri-graph	WA WA WA	WA WA AC	CE CE CE	TL TL TL	RE RE RE	CE WA WA
Frequency	93753	33686	20508	14551	14349	10015
Tetra-graph	WA WA WA WA	WA WA WA AC	CE CE CE CE	RE RE RE RE	TL TL TL TL	CE WA WA WA
Frequency	56065	16828	11774	8040	7521	5686
Penta-graph	WA WA WA WA WA	WA WA WA WA AC	CE CE CE CE CE	RE RE RE RE RE	TL TL TL TL TL	CE WA WA WA WA
Frequency	35025	9241	7368	4810	4187	3416
Hexa-graph	WA WA WA WA WA WA	WA WA WA WA WA AC	CE CE CE CE CE CE	RE RE RE RE RE RE	TL TL TL TL TL TL	CE WA WA WA WA WA
Frequency	22475	5330	4650	2962	2433	2092

Table 7: A table for most popular response sequence ending with an AC or having length 6

Popularity	1 st	2 nd	3 rd	4 th	5 th	6 th
Monograph	AC					
Frequency	465516					

Di-graph	WA AC	CE AC	TL AC	PE AC	RE AC	
Frequency	71018	18099	10612	9213	8205	
Tri-graph	WA WA AC	CE CE AC	TL TL AC	CE WA AC	RE RE AC	TL WA AC
Frequency	32765	4685	3540	3511	2620	2423
Tetra-graph	WA WA WA AC	CE CE CE AC	CE WA WA AC	TL TL TL AC	RE RE RE AC	TL WA WA AC
Frequency	16518	1750	1636	1340	1158	1114
Penta-graph	WA WA WA WA AC	CE WA WA WA AC	CE CE CE CE AC	TL TL TL TL AC	TL WA WA WA AC	RE RE RE RE AC
Frequency	9115	842	827	618	573	563
Hexa-graph	WA WA WA WA WA WA	WA WA WA WA WA WA AC	CE CE CE CE CE CE	RE RE RE RE RE RE	TL TL TL TL TL TL	CE WA WA WA WA WA WA
Frequency	22475	5330	4650	2962	2433	2092

Due to a small error in mapping of our program the results of table 7 has little errors. For example the number of AC monograph should be same in both tables but the first table shows more. It is because our program considers sequence AC and AC|AC the same. This will be fixed in the final version. For this reason the response sequence WA|WA|WA|WA|WA|AC has same frequency on both tables.

Many comments can be made based on these tables. But some things are obvious

- When a contestant make mistakes for a problem he tends to make the same mistake again.
- We can say that if someone gets five consecutive wrong answers then in the next submission he is four times more likely to get a wrong answer than an accepted verdict.

All these results can help coaches to identify the mistake pattern of his team, one can judge whether a programming contest took place ideally (Must have similarity with these results), in future we may be able to predict the result of a contest based on this.

Things yet to be done in this paper:

I had plans to find many results for this paper. But each topic became longer than what I expected. So the results that are half complete or yet to be implemented are:

- Age specific contestants in different country. Does it have to do anything with IOI result, I mean do the countries with more school college level contestants do well in IOI. (We are least hopeful to complete it because only UVa data is not enough for this)
- Tendency of solving math/geometry/NT problems, anything to do with IMO. What if when solving difficult problems of math and geometry and number theory.
- Coding in holidays. More or less, 25th December, 1st May and Saturdays, Sundays (In Christian countries), Friday in muslim countries.
- Countries which has much interest in university level but in poor shape in school/college level
- Does TL (Time limit exceeded) has anything to do with age and experience.
- Error rates in different types of problems. Contestants and coaches often say that geometric problems are more risky and hard to get accepted, we will try to find out if it is true with respect to UVa. Does numerical problems have less acceptance rate?
- At what time do most people use online judge (Can we assume that this is the time when people like to code as well? I am not sure)
- We also hope to add some references when we add more text with this paper. The main reason for absence of reference is that we have not read something similar to this paper before.
- Suggestions are welcome if someone wants to see new types of results.