Competing in Computing

Fredrik Manne[*]

Abstract

This paper describes a study performed in conjunction with organizing a local qualification contest for the ACM International Collegiate Programming Contest. This is an annual contest where teams of students compete against each other in writing computer programs.

The objective of the study was to see if the effort that was put into organizing this event gave sufficient results that it warrants a continued participation. It was also a chance to evaluate how a competition can be used as a learning environment.

The data for the study was gathered from having the participating students fill in a questionnaire after the competition.

1 Introduction

The Association for Computing Machinery (ACM) [1] is an interest group for professionals in computer science which for the last 24 years has arranged "The Annual ACM International Collegiate Programming Contest" (ACM-ICPC) [2]. This is a competition where teams of students compete against each other in writing computer programs.

In this paper we describe a study that was performed in connection with the local qualifying competition for the ACM-ICPC held at the Department of Informatics, University of Bergen, Norway in 1999. This was the first time that the University of Bergen organized a local qualifying competition and sent a team to the regional final (in 's-Hertogenbosch, The Netherlands [14]).

Organizing a local competition requires some effort from the faculty in terms of spreading information, designing problems, organizing the needed hardware and software, and in terms of administering the competition itself. In addition, there is a financial cost in sending the best team to the regional final. In 1999 this amounted to approximately 20.000 Norwegian kroners for The University of Bergen. Since both time and money is spent on participating in the ACM-ICPC it is of interest to evaluate what the benefits are to the university and if these are substantial enough to defend a continued participation. This is the purpose of the present paper.

Our main motivation for participating in the ACM-ICPC was to create a positive social and academic event for the students. In the Norwegian university system students can select courses fairly freely and are responsible themselves for combining courses that make up a bachelors degree. Due to this freedom there is seldom a feeling among the first and second year students of belonging to a certain class or department. It has been reported, and is also the author's own experience, that it takes time, often several

[*] Department of Informatics, University of Bergen, N-5020 Bergen, Norway. Fredrik.Manne@ii.uib.no
.

semesters, before a student starts to feel like an “informatics student”. Thus we wanted the competition to be an event that could help students identify more closely with the department.

Since the competition is organized as a team competition we also had the chance to see if the competition itself could be used as an environment for learning. This concerns both learning pure computer science and also seeing the importance of team-work. One might hope that both during the preparation for the event and during the event itself students would learn things that are hard to convey in an ordinary classroom setting. In addition, the competition gave us a possibility to see to what extent university education prepares students for such a competition.

Currently in Norway, there has been a strong increase in the number of students wishing to study informatics. Of the students starting their studies in 1999 at the Faculty of Mathematics and Natural Sciences, University of Bergen, 24% reported that they wished to pursue a masters degree in informatics [17]. However, this number has varied strongly in the last 10 years mainly depending on the job market. In the coming years the number of students eligible to enter higher education will decrease due to lower birthrates. It is therefore expected that there will be stronger competition between schools and colleges to attract students. This is reinforced by the numerous private schools offering education in various aspects of computer science. Thus the competition was also viewed as an opportunity to obtain publicity in the media for informatics at the University of Bergen.

The rest of the paper is organized as follows: In Section 2 we give a short description of how the contest is organized and also of the relevant education at the University of Bergen. In Section 3 we give a description of how the local competition was organized. In Section 4 we report on the evaluation of the contest both as perceived by the organizers and from student feedback. Finally, in Section 5 we conclude.

2 Background

In this section we first give a description of the ACM-ICPC. We also give a description of the relevant courses given at the Department of Informatics, University of Bergen.

2.1 The ACM International Collegiate Programming Contest

The ACM-ICPC is organized as a two-tiered competition. Teams first compete in regional contests around the world with the best teams advancing to the world finals. It is also possible for each participating university to arrange an initial local competition in order to select the teams for the regional contest. In 1999 there were more than 2400 teams world-wide participating in the contest.

A team may have no more than three members of which no more than one can be a graduate student. Moreover, the graduate student cannot have completed two or more years of graduate studies or hold a graduate degree.

During a contest each team is assigned one computer that must be shared between the members of the team. The teams are allowed to bring any written material while the use of material stored in an electronic format is prohibited. At the start of the contest the teams are given a set of between 6 and 8 problems to solve in 5 hours. Along with each problem there is also an example of sample input and sample output. A solution consists of a computer program that reads a valid input file and outputs the correct answer. In addition, the program must output the answer within a certain time limit in order to be accepted.

During the contest the teams submit solutions electronically to the judges. The judges then compile and run the program on a set of secret test data. Depending on the outcome, the team is told if the solution is accepted or not. If it is not accepted the team is only told what type of error occurred. This is done by returning one of the following messages: “run-time error”, “wrong answer”, “presentation error”, or “time-limit exceeded”.

The teams are scored by counting the number of accepted solutions. If two teams have solved the same number of problems they are ranked by the total accumulated time spent on the accepted problems. For each solved problem there is also a 20 minute penalty for each submitted solution that was not accepted.

The problems in the ACM-ICPC only requires knowledge of basic algorithms in addition to programming skills. This includes various forms of search: breadth-first, depth-first, and brute force. Other subjects that are assumed known include dynamic programming, simple computational geometry, basic combinatorial knowledge, and tree structures.

The problems differ from text-book problems in that they are rich in text and that they try to make novel use of standard solution techniques. The problems are always given in a real-world setting. Thus part of the problem is to extract the essence of the problem and reduce it to a problem in computer science. The students must then decide which solution technique to apply and implement it.

2.2 Informatics at the University of Bergen

In this section we give a description of the education at the Department of Informatics, University of Bergen, that is relevant for students wishing to compete in the ACM-ICPC.

The Department of Informatics offers a bachelor, a master, and a doctoral program. There are approximately 450 undergraduate students and approximately 90 master students. One cannot start on the doctoral program until one has completed a masters degree. Doctoral students are therefore ineligible to compete in the ACM-ICPC.

In order to obtain a bachelors degree in informatics students must initially take a basic programming course [7, 13]. This is then followed by an introductory data structures and algorithms course [6, 8]. Both of these courses have a work-load corresponding to a half semester of full time study and contains several compulsory programming exercises using Java. A student who has completed these two courses should be able to solve most of the problems given in the ACM-ICPC. Before finishing a bachelors degree the student must also take a more advanced theoretical algorithms course [4, 10]. After this course he or she should be able to solve all of the problems in the ACM-ICPC.

Finally, we mention the “Computers and Society” course [9]. This course is also compulsory for undergraduate students in informatics. Here groups of students write a joint term paper on some issue related to the impact of computers on society. In addition to learning the subject itself this course is intended to teach the students how to work in groups. The grading for each of these courses is based on a written final exam.

3 The Local Contest

The local competition at the University of Bergen was organized by the author with the additional help of his colleague Jan Arne Telle on the day of the competition. Information about the contest was spread among the students by the means of posters, announcements in lectures, and e-mail. The main sales pitch was that this would be a fun event with the possibility of winning a trip to the Netherlands to compete in the regional final. In addition it was pointed out that a good placement would most likely be an advantage later on when applying for a job.

Detailed information such as the specific rules and sample problems were made available on a web site [12]. Two weeks before the contest an information meeting was held where sample problems were presented and rules were explained. The students were told that they could either enter the competition as a whole team or they could enter individually and we would help them form teams. At the deadline for registration 8 teams with a total of 23 students had registered for the contest. Prior to the contest a final problem set was developed and 8 computers in different offices were made available. New accounts were created on the departmental network for each team and the computers where checked to see that they all ran the same required software.

During the contest the teams communicated with the judges by use of e-mail. In addition to being notified about the status of their submitted solutions the teams also received temporary rankings during the competition. To ensure that the rules were being followed the computers were set up to log the commands that were entered. In addition one of the judges occasionally went around and checked on the teams.

Apart from one error in the secret test data and two clarifications about the problem set the competition went smoothly. Teams solved between one and six problems with an average of 2.5 accepted answers. After the competition the teams were gathered in a lecture room where pizza and soft drinks were served before the announcement of the final standings. The day after the competition all participants were contacted by means of e-mail and asked to fill out a questionnaire about how they had perceived the competition.

The winning team went on to compete in the North European regional final finishing in 20th place out of 41 teams.

4 Evaluation

In this section we present an evaluation of the contest. Initially, we give a short summary of how the organizers perceived the event. We then give a more thorough presentation of the questionnaire feedback from the participating students.

4.1 Self Evaluation

The general feeling among the organizers was that this had been a positive and successful event. The immediate feedback from the students after the competition supported this view. Although most of them were exhausted after the competition the immediate feedback was that this had been fun and was well worth repeating. Comments from other faculty members who witnessed the contest was also very favorable.

The subsequent participation in the regional finals generated four news articles. Two in the local daily newspaper [3], one in the biweekly newsletter from the University of Bergen [15], and one in an online computer newsletter [5].

As mentioned in Section 3 we discovered an error in one of the data sets used for testing and also needed to clarify some issues related to the test problems. Although some teams lost time for this reason, it is highly unlikely that it influenced which team won the competition.

After the competition had ended we realized that we had misunderstood the rule for treating incorrect answers. We had informed the students that all incorrect answers would give a 20 minute penalty independently of whether the problem was finally solved or not. However, as this would not have changed the placement of the first three teams we choose to uphold our initial interpretation of the rule.

4.2 Student Feedback

As mentioned in Section 3 a questionnaire was sent to all the participating students the day after the competition. We got 19 answers from the 23 participants.

The questions were divided into four categories. Initially, the background of the students was established.

- Of the students 16 were undergraduate while 3 were graduate students.
- The students ranged from first-year students to students in their 5th year. The average number of semesters studied was 5.4, with an average work-load of 3.0 semesters in computer science.

We were reasonably satisfied with the attendance especially among the undergraduate students. We had hoped that more graduate students would have entered. In a similar contest held at the Department of Informatics, University of Oslo, 21 students participated [11]. This is a department of approximately double the size of the Department of Informatics at the University of Bergen.

4.2.1 Before the Competition

The second group of questions were related to events prior to the competition.

- The students were asked if they had received sufficient information prior to the contest. To this 16 students answered yes. There where two comments about ambiguities in the information, one student wanted the rules to be more specific and one wanted the way I/O should be handled to be more precise. There was also one student who wanted the information earlier.
- When asked why they had entered the competition 13 students answered that they had joined to “have fun”, four to compete/win, one was curious, and one had joined a friend.
- Most students (14) stated that they would have joined even if the trip to the regional final had not been offered as first prize. Only two answered that they would not have joined.

- The amount of time spent on preparations varied from none (5 students) to more than 10 hours (3). The median student spent between 3 and 4 hours in preparations. In order to prepare most students (11) solved test problems from previous competitions.

Judging from the student feedback the amount of information given to the students prior to the contest was sufficient. It seems also that the “fun” part of the contest was well perceived by the students. The two comments about ambiguities in the information stems from the fact that the organizers had no previous experience in arranging events of this kind. Hopefully this might improve if we were to arrange a similar contest in the future.

It seems as if the first prize was not the major incitement for participation. However, by arranging the contest within the ACM-ICPC we got a well-tested framework for the contest that would otherwise have been difficult to achieve. Also, the first prize was used for obtaining media publicity. Moreover, competing in the regional final gave us a chance to evaluate some of our best students compared to those of other universities.

It should be noted that the members of the winning team spent considerable more time on preparations than the other teams. Moreover, the two top teams both contained a graduate student. This shows the importance of both training and experience in performing well in this type of event.

4.2.2 The Competition

The third group of questions were concerned with the competition itself.

- When asked if they were satisfied with what their team accomplished only 5 students answered yes. The rest were dissatisfied.

- There were 11 students who thought the level of difficulty was as expected, while there were 4 who thought the problems were more difficult than expected, and 3 who thought the problems where too easy.

- All students reported that they found the event to be “fun”. 8 students also reported that they found it to be “stressing”.

- 7 students reported that the cooperation within their teams was either good or excellent. Other responses include “not used to work under pressure” and “did not have a clear division of responsibilities”.

  Students report that they gained programming and team-work experience from the competition. Some of the specific comments were:

  “The team-work was much more team-work than the project in the “Computers and Society” class. We were forced to fill in each other’s gaps of knowledge”,

  “I was almost shocked when I realized how long time I needed to get a fairly simple piece of code to work”,

  ”Java is difficult when it comes to I/O”.

- When asked what was the most difficult part of solving the problems an equal number of students (6 each) reported I/O, debugging, and designing algorithms.

- 10 students thought the social gathering/result session was OK. The ones who were missing anything wanted a quick explanation of how to solve the more difficult problems and a more official price ceremony.

- On what to change in order to improve the contest 10 students did not want to change anything. Other suggestions included bigger offices to work in, clearer rules, and more thorough review of the problem set and the test data.

- 17 of the students reported that they would participate if a similar event was held next year. Only one did not want to participate. His reason was that all his friends would then no longer be eligible to compete.

When constructing the problem set we tried to make it somewhat easier than those that had been used in previous regional and world finals. The reason was that we wanted all the teams to be able to solve at least one or two problems.

It seems like several students were surprised at how difficult it was to write correctly working programs. Most participants had expected to do better than they did. Of the students who were satisfied with their own performance 4 out of 5 were all members of the two top teams. One may hope that one lesson learned is that both programming and team-work require practice and planning in order to be efficient.

It was surprising that so many teams had problems with I/O. The I/O was always very structured and should not have represented a problem. It might be that this problem is particular to Java since none of the teams using C++ reported this problem.

The high number of students who found the event fun and who would like to participate again was very encouraging. However, it should be noted that since the questions were answered by email the students were not anonymous. It is thus possible that some of the answers were tainted by a wish to please the organizers.

4.2.3 The Education

Finally the students were asked questions concerning how they perceive the education at the department related to this kind of an event.

- Only one student reported that the education given at the department was not relevant for the competition. The other students report that the course on data structures and algorithms and the advanced algorithms course were relevant.

- When asked what is lacking in the education in order to take part in this kind of competition, 6 students answered more programming experience, while two students wanted more computational geometry.

- The contest does not seem to have influenced the determination of undergraduate student's of whether or not to pursuit a masters degree, nor does it seem to have influenced what subject they would be more likely to choose in case they did continue.

- The students were evenly divided on whether this kind of a competition is realistic compared to what they will be doing when they graduate and get a job. They were also equally divided when asked if such a competition could be used as a final exam in a course. The main objection was that there would be no credit for partially correct answers and that the result may depend on the other members of the group.

The comments on which courses were useful for the competition were as expected. The focus on programming and on actually implementing the algorithms that are taught is perhaps too low. This applies especially in the theoretical algorithms course which has no compulsory programming exercises.

It it is to much to hope that this event in itself would encourage students to pursue a masters degree. However, it is conceivable that it might be one of several factors that encourages students to continue with graduate studies.

It is interesting to see that many students would accept this kind of competition as the final exam of a course. As of today all the exams are (hand) written and computers are only allowed for students with writing disabilities. The non-use of computers is mainly due to technical reasons and might change in the future.

5 Conclusion

In this paper we have described the organizing of a local qualifying competition for the ACM-ICPC and presented and discussed feedback from the students who participated in it.

Our conclusion is that the event functioned as intended. The students perceived it as fun and were also able to gain experience in programming and in team-work. In addition, the students performed some self study that they would otherwise not have done while preparing for the event. We believe that the time and money that was spent in arranging the event was well worth. It is therefore our recommendation that the University of Bergen should continue to participate in the ACM-ICPC.

We note that it may be objected that encouraging students to train and participate in the ACM-ICPC takes the focus away from training them to be scientists. The objection is that too much emphasis is on writing code quickly and practicing using the same basic techniques over and over again. The setting with three students sharing a single computer might also be perceived as unrealistic.

As long as the importance of participating in the ACM-ICPC is not overdone we believe that these issues do not present a problem. Some level of repetition is always needed in order to learn. Moreover, the problems used in the ACM-ICPC are often novel and challenges the creativity of the students. By only allowing one computer per team it forces the students to work on paper and to think before they start coding. It is often a problem that students start coding before they are adequately prepared thus wasting much work and time.

Although we wish to continue participating in the ACM-ICPC there is still room for improvements in how to organize our local competition. We should have checked both the problem set and the secret test data better. We should also have read the rules better. These are things that we hope will get better as we gain more experience. The social event after the competition was also not very well planned. It should be more structured with a separate session for scoring and prizes and one for discussing the problems.

If we stress the importance of preparation it is conceivable that the amount of time spent on this can be increased and that problems such as those that were reported on getting I/O to work can be avoided. However, this must be balanced against the “fun” part of the event. We do not want to scare students from participating because they fear it will be too time-consuming.

We were satisfied with the response from the media, although it is difficult to estimate what effect it had. If one is to get even more attention the event itself must be made more visual. In the regional final the students competed in a large open room and were given differently colored helium balloons for each correct answer. In addition, scoring was displayed online just outside the competition area using a projector. However, it is not clear if this is an option for the local contest since it would require more people and a bigger organization.

It should be noted that all participants were male. While between 15 and 20 % of the students are female none of them chose to participate. It might be possible that organizing all-female teams might make it easier for female students to enter the contest.

Together with the University of Oslo we are currently working on plans for turning the local qualification competition into a national contest similar to that in Sweden [16]. In this way it should be easier for new schools to enter the competition. If one arranges the contest at the same time on different locations one only need to develop one set of test problems for the whole country. The work carried out at each school would then be to attract students, to set up local hardware and software, and to act as judges during the day of the contest.

References

[1] *ACM: Association for Computing Machinery.* http://www.acm.org.

[2] *The 1999-2000 ACM International Collegiate Programming Contest.* http://acm.baylor.edu/acmicpc/.

[3] *Bergens Tidende*[1], 15/11-1999. http://www.ii.uib.no/nyheter/prog99/bt/.

[4] T. H. Cormen, C. E. Leiserson, and R. L. Rivest, *Introduction to Algorithms*, The MIT Press, 1996.

[5] *digi.no*, 1999.
http://digi.no/digi98.nsf/pub/jobb19991117112800TKW1115892271.

[6] M. T. Goodrich and R. Tamassia, *Data Structures and Algorithms in Java*, John Wiley & Sons, 1998.

[7] *I110 - Grunnkurs i databehandling.* http://www.ii.uib.no/undervisning/kurs/i110/.

[8] *I120 - Algoritmer, Datastrukturer og Programmering.*
http://www.ii.uib.no/undervisning/kurs/i120/.

[9] *I191 - Databehandling og Samfunn.* http://www.ii.uib.no/undervisning/kurs/i191/.

[10] *I234 - Algoritmer.* http://www.ii.uib.no/undervisning/kurs/i234/.

[11] *Ifi-mesterskapet i programmering 1999.* http://www.ifi.uio.no/ db/ifi-m99/.

[12] *Instituttmesterskap i programmering 1999.* http://www.ii.uib.no/nyheter/prog99/.

[1] All Norwegian websites are in Norwegian

[13] J. LEWIS AND W. LOFTUS, *Java, Software Solutions, Foundations of Program Design*, Addison-Wesley, 1998.

[14] *Northwestern Europe Regional Contest of the ACM-PC 1999.* http://212.142.21.29/acm/99/.

[15] *På Høyden, Nr. 18*, November 1999. http://www.uib.no/elin/elpub/ph/1999/18-99.pdf.

[16] *Svenskt Mästerskap inom ACM-ICPC.* http://www.dna.lth.se/contest/.

[17] *Informasjon fra studieveileder, Det Matematiske-naturvitenskapelige fakultet, Universitet i Bergen*, October 1999.