Computational Sciences Learning Project for Pre-university Students

Mustafa Murat Inceoglu[1] and Burak Galip Aslan[2]

[1] Ege University, Computer Education and Instructional Technology,
35100 Bornova, Izmir, Turkey
mustafa.inceoglu@ege.edu.tr
[2] Izmir Institute of Technology, Department of Computer Engineering,
35430 Gulbahce, Urla, Izmir, Turkey
bgaslan@ieee.org

Abstract. It is quite useful for high school students to study on computational sciences, especially for those who are planning to have a higher education in engineering. This paper proposes a pilot study on 9th and 10th grade students for teaching them the basic aspects of computer sciences. This study focuses on a schedule consisting two consecutive courses – first one theoric and the second one is applied – for student volunteers from science high schools.

Keywords: pre-engineering education, computational sciences education, parallel computing, MPI.

1 Introduction and Motivation

Pre-university engineering education is introducing the notion of engineering to high school students who are inclined to engineering education. Such an engineering education is crucial for; graduation of destined engineers, preparing the students for the competitive field of engineering following their graduation, and making them gain experience on critical decisions and creativity. 2-day pre-university design course of Kanazawa University [1], and pre-engineering project design studies of the University of West Florida over selected 4-year high school graduates [2, 3, 4] are interesting examples of pre-engineering education projects.

Computer engineering and electrical engineering are essential areas of engineering in developing countries like Turkey. Parallel computing lays the foundation of computational sciences. Engineering students are expected to solve complex problems via parallelization in this field.

2 Computational Sciences Education for High-School Students

Computer clusters are perfect tools for parallel computing education. Today, clusters are very cost-effective for education units in teaching parallel computing education. The students are learning the concept of *coarse grain parallelism* easily by using clusters [5]. There are also several studies in literature for teaching parallel computing to both undergraduate and graduate students [6-9].

Y. Shi et al. (Eds.): ICCS 2007, Part III, LNCS 4489, pp. 607–610, 2007.

3 Project Setup

The implementation of the project consists of 5 steps; teacher selection, student selection, orientation seminars, courses, and evaluation.

Teacher Selection: Five different teachers required for different stages of project are chosen from the high schools of volunteer students by informing them about the project and discussion. Four of these teachers are assigned to prepare the math, physics, chemistry and biology questions of a selection exam for volunteer students. Same teachers are also assigned to evaluate the exam results. *Blind-evaluation* technique is used for the evaluation of exams. One of the teachers who has a background of mathematics is assigned to teach basic matrix algebra in program.

Student Selection: All of the selected students are studying in science high schools which are education grounds of science and technology. The project has started in 2003-2004 academic year and only 9th grade students are enrolled in the project in order to achieve continuity. The selected 9th grade students are invited for the second stage of the project in the 10th grade. The project is finalized as of November 2006. Some information about the selected high-schools and students are given in Table-1. The students have been asked a total of ten questions from mathematics, physics, chemistry, and biology in the selection exam for better education of pre-university students in the field of computational sciences. The students who are preparing for the Olympics of computer and mathematics have been given higher priority. The willingness of students has been the most important criteria during the student selection process.

Academic year	2003 2004	2004 2005	2005 2006	TOTAL
Selected High-Schools (A)	2	3	3	8
Student volunteers who have taken selection exam (B)	11	17	19	57
Student volunteers who have passed the selection exam (C)	7	14	11	32
Selected students (D)	5	9	7	21
Ratio of selected students (D/B)*100	45%	53%	37%	

Table. 1. Information about the selected high-schools and students in project

Orientation Seminars: Three orientation seminars have been organized so as to introduce the students the concept of computational sciences. The first seminar was based on introducing the basic concepts of engineering. The second seminar followed as giving information about computer engineering, electrical engineering, and professional organizations such as Institute of Electrical and Electronical Engineers (IEEE) and Society for Industrial and Applied Mathematics (SIAM). The last seminar was about parallel computer systems, parallel applications, and some applications from daily life.

Courses: The project is planned to be completed in 3 years. Table. 2. shows the number of selected students, the number of drop-outs, and their corresponding ratios in relation with the academic year. Combining the information from Table.1. and Table.2., it can be seen that there is an increase in both the number and the continuity of students during the three-year project.

Academic Year	2003 2004	2004 2005	2005 2006	End of Project
Selected students (A)	5	9	7	---
The students who did not continue in the second year (B)	---	2	3	1
The ratio of drop-out students (B/A)*100	---	40.0%	33.3%	14.3%
The students who have completed their education plan (C)	---	3	6	6
The ratio of students who completed their education (C/A)*100	---	60.0%	66.7%	85.7%

Table. 2. The attendance information about enrolled students in the project

The project of computational sciences education has been planned as a 2-year program. There will be two semesters in an academic year, 15 weeks in a semester, and 3 hours of lesson for each week so that there will be a total of 90-hour program.

First year covers; Algorithms and C programming (40 hours), and parallel computing (15 hours) courses are given by professors from university, while basic matrix algebra (15 hours) course is given by high-school mathematics teachers. Parallel applications (20 hours) are realized within the supervision of university professors in the applied course. Second year covers; Parallel algorithms (30 hours) course where five algorithms are introduced in this course. Namely; parallel vector addition, parallel sorting, parallel matrix addition, parallel matrix multiplication, and parallel solving of linear systems. Laboratory (60 hours) is about basic knowledge about Message Passing Interface (LAM/MPI) installation, and MPI programming is introduced to students in this course.

Evaluation: *Academic year evaluation* covers the first year evaluation of enrolled 9th grade students in the project has been handled by the university professors and high-school math teachers who have given the courses during the academic year. This stage evaluates the correspondence between project targets and achieved results. The criticisms of students are examined thoroughly by both one-to-one and group discussions. *Semester evaluation* covers the 10th grade students who have completed the first year of the project and continuing the second year are evaluated in two stages. The first stage of evaluation is the evaluation of laboratory applications as an ongoing task. The five parallel algorithm applications are run within the supervision of university professor who gave the course, and the results are discussed. The second stage of evaluation is one-to-one and group discussion with the enrolled students.

4 Results and Discussion

Most of the enrolled students who completed the 2-year project clarified their positive comments on the education project. Considering that science high-schools education is based on science and technology, the students did not show much of resistance; quite the contrary, the students sometimes gave new comments on what could be done in the following years.

Teaching concepts of basic matrix algebra, open-source software such as Linux and MPI, C programming could be very useful for high-school students could be very useful if they continue their higher education in engineering departments. These courses mentioned above lay foundations of several university courses such as differential equations, linear algebra, computer programming which are usually being given in the first years of engineering education. It has been realized that, it is quite advantageous for a future engineering student to know how to program an important parallel library, MPI.

References

[1] Matsuishi, M., Kitamura, T.: Introductory Pre-College Engineering Design Course Based upon Collaborative Learning to High School Students, IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing, Vol. 2 (2006) 138 – 142

[2] Harrell D., Bataineh M., El-Sheikh E., Spolski J.: The Development of A Pre-College Engineering Curriculum for High School Students: Design and Implementation, 34th ASEE/IEEE Frontiers in Education Conference, Savannah, GA (2004)

[3] Hirsch, L., S., et al: High School Student's Attitudes To and Knowledge About Engineering, , 33rd ASEE/IEEE Frontiers in Education Conference, Boulder, CO (2003)

[4] Todd R.: A New Paradigm for Pre-Engineering Education: Design and Technology Education K-12, 1993 Frontiers in Education Conference (1993)

[5] Prins P.: Teaching Parallel Computing Using Beowulf Clusters: A Laboratory Approach, Journal of Computing Sciences in Colleges. 20 (2004) 55-61

[6] Fung Y., F., Ercan M., F., Chong, Y., S., Ho, T., K., Cheung, W., L., Singh, G: Teaching Parallel Computing Concepts with A Desktop Computer, International Journal of Electrical Engineering Education. 41 (2004) 113-125

[7] Bernreuther, M., Brenk, M., Bungartz, H., J., Mundani, R., P., Muntean, I., L.: Teaching High-Performance Computing on A High-performance Cluster, Lecture Notes in Computer Science. 3515 (2005) 1-9

[8] Aloisio, G., Cafaro, M., Epicoco, I., Quarta, G.: Teaching High Performance Computing Parallelizing A Real Computational Science Application. Lecture Notes in Computer Science. 3515 (2005) 10-17

[9] Apon, A., Mache, J., Buyya, R., Jin, H.: Cluster Computing in the Classroom and Integration with Computing Curricula 2001. IEEE Transactions on Education. 47 (2004) 188-195