Implementing a New Computer Science Curriculum for Middle School in Israel

Iris Zur Bargury
Ministry of Education
Tel-Aviv, Israel, and
Babeş-Bolyai University Cluj-Napoca, Romania
iriszu@education.gov.il

Orna Muller
Software Engineering Dept., and
Teaching & General Studies Depts.
Ort Braude College of Engineering
Karmiel, Israel
ornamu@braude.ac.il

Bruria Haberman
Holon Institute of Technology, and
Davidson Institute of Science Education
Weizmann Institute of Science
Rehovot, Israel
bruria.haberman @weizmann.ac.il

Doron Zohar
Open University of Israel
Ministry of Education
Tel-Aviv, Israel
doron.zohar@openu.ac.il

Avi Cohen
Ministry of Education
Inspector-In-Chief, Computer Science
Tel-Aviv, Israel
avico@education.gov.il

Dalit Levy
Zefat Academic College, and
Kibbuzim College of Education
Tel Aviv, Israel
dalit_lev@smkb.ac.il

Reuven Hotoveli
Afeka - Tel-Aviv Academic College of Engineering
Tel-Aviv, Israel
reuvenh@afeka.ac.il

***Abstract*— as part of a national strategic plan recently established by the Ministry of Education in Israel to strengthen science and technology education, an innovative Computer Science (CS) curriculum for middle school was developed. One main goal of the new curriculum is to expose students at an early stage of education to the fundamentals of CS and computational thinking, and to encourage students to study CS in the future. We present the curriculum and its initial implementation, focusing on issues of teachers' professional development.**

Keywords- algorithmic thinking, computational thinking, computer science, curriculum, middle school, education, teachers' professional development

I. Introduction

Aiming to increase the number of high-school students who choose to major in science and technology, the Israeli Ministry of Education launched the *Science and Technology Excellence program* (STEP) in 2011 to increase the number of high-school students who choose to major in science and technology. This unique program is part of the overall strategic plan to strengthen science and technology K-12 education in Israel[1]. Currently, less than ten percent of Israeli high-school graduates major in both science and technology. Raising this low percentage may improve the readiness and preparation of potential candidates for academic studies in science and engineering, and eventually influence their becoming part of the high-tech industry.

The main goals of the STEP are therefore to expose middle-school students to the fundamentals of science and technology in order to encourage them to choose a major in these areas in high school, and to develop a sense of leadership in science and technology during their high-school studies. The new middle school STEP curriculum, which precedes the existing high-school curriculum in science and technology, creates a continual and comprehensive six-year science and technology curriculum for highly capable students. It includes extra hours of Mathematics, Natural Sciences (Chemistry, Biology, and Physics) and Computer Science (CS) combined with Robotics.

The Israeli CS curriculum for high school is well known internationally [7]; however, until recently, in typical Israeli middle schools the students studied computer literacy, except for a few short-term educational initiatives practiced on a local

[1]http://cms.education.gov.il/EducationCMS/Units/MadaTech/

basis for learning CS. The need to rethink computing education on a national level complies with the current national effort to adapt the education system to the twenty-first century.

The education authority's decision to include CS in the middle-school's new STEP was based on the premise that learning CS in middle school would promote early acquisition of computational thinking, which in turn, would enhance students' scientific thinking and technological literacy. The guiding principle for the new CS curriculum has therefore been to focus on developing thinking skills rather than programming skills, and to expose students to various development environments using an inquiry-based approach, and utilizing learning-through-enjoyment pedagogical methods that can increase young students' motivation.

The curriculum consists of both mandatory and elective modules: *a. Introduction to CS*, which provides the basis for the entire program, and emphasizes the fundamentals of algorithmic thinking; b. the *spreadsheet*, with an emphasis on its usage for scientific inquiry; its inclusion in the CS curriculum bridges the gap between the CS middle school curriculum and other parts of the six-year STEP; *c.* Selection between *Introduction to Robotics*, which exposes students to engineering concepts and problems, and *Basic Internet Programming; d.* The last module is devoted to developing a small *programming project* from scratch.

The above modular program has been designed by a professional committee established by the Israeli Ministry of Education. The authors of this paper are all members of that committee. From 2011 to the present, the first two modules have been implemented in 183 pilot schools, concomitantly with the initiation of professional development activities under the supervision of the first author.

The next sections of this paper elaborate on the principles and constraints that guided the committee in designing the curriculum. The first section presents the rationale for teaching computing at the K-12 level, and lists the programs currently taught in Israel. Thereafter, the new CS curriculum for middle school is described, and finally, implementation issues are discussed, focusing on teachers' professional development.

II. K-12 Computing Program In Israel

A. Rational and Motivation

Computer science provides the knowledge and skills foundation for contemporary technological advances: "Maintaining our ability to meet present and future challenges requires us to acknowledge CS as a core element of all STEM (science, technology, engineering, and mathematics) initiatives" ([11], pp. 15). Learning CS enhances computational thinking and may contribute to a better understanding of other subjects as well [12]. Strengthening the status of CS as a full-fledged and self-contained subject in the educational system is most important [11]. However, the increasing complexity of the field led to an unfavorable external image [3] and posed new challenges in motivating students to pursue CS as a career choice or a course of study in which to major [11]. "Despite many years of our trying to broaden our image, computing is still widely perceived as a programmer's field … Many outsiders wonder whether CS departments will eventually disappear as the technology evolves and other fields take over as the main contributors of new computing technology" [3, pp. 336].

To address this problem on a national as well as a global level, [1] suggested that CS be presented as a ***language of technology,*** which describes structures, processes, relationships, and communications. Computer science serves as a platform for problem solving, knowledge representation, and formalization of processes, as well as for understanding technology and for performing technology-related processes. CS should be taught to youngsters as one of five basic languages: a mother-tongue, an elective international foreign language, a language of science (mathematics), a language of art and body, and a language of technology (computer science), each of which is used to express themes and ideas or feelings associated with specific domains and contexts. "Long-term study of these five languages, along with intelligent practice while elaborating on utilizing communication skills, is highly useful for successful functioning on personal, national, and global levels" [2, pp. 54].

B. K-12 Computing Curricula in Israel

During the last few decades, various computing curricula have been taught in Israeli schools as an elective subject, ranging from computer literacy, up to computing fluency, usually learned at elementary and middle school as part of other subjects. This includes CS for high school in the academic track [7] and software engineering for high school in the high-technological track [9]. The curricula evolved over the years according to changes and development of the field. Pre-service and in-service teacher training courses were created to provide teachers with technical and pedagogical knowledge [7,10]. Many research papers were published regarding the implementation of these curricula using various pedagogical approaches and students' conceptualization of CS concepts and their problem-solving performance (for example, see a review in [11], pp. 29-51).

The CS curriculum for high school introduces *CS* concepts and problem-solving methods independently of specific computers and programming languages, along with their practical implementation in actual programming languages [7]. The program consists of five modules (90 hours each): (a) *Fundamentals of CS 1 and 2*, which introduce algorithmic problem-solving (two modules, 180 hours in total); (b) *Software Design*, which focuses on abstract data types and object-oriented programming; (c) *A Second Paradigm* - alternatives to this module are logic programming, functional programming, computer organization and assembly language, computer graphics, information systems, and stateless programming; and (d) *CS Theory*: Computational models.

The programming languages chosen for teaching fundamentals and software design have changed over the years and currently are Object Oriented (JAVA or C#).

The Software Engineering program: During the last two decades, a Software Engineering (SE) program especially designed for high-school level has been in operation in Israel [8]. The program consists of (a) an elective topic in natural sciences/introduction to technology sciences, (b) Computer Science, and (c) an elective and advanced topic in CS.

The program has scientific foundations and can be viewed as an extension of the CS program. The main goals of the program are (1) to introduce students to knowledge representation, system-level perspective and formalization of processes, and (2) to promote students' creativity, and enable them to construct an integrative knowledge of CS topics. The specialized topics that schools can choose from are as follows: Operating Systems, XML Web Services, Computer Graphics, Expert Systems, and Design & Programming of Information Management Systems. The students' final assignment is to develop a comprehensive software project related to the elected specialized topic.

III. The New CS Curriculum For Middle School

The new CS program for middle school is part of the STEP, based on an overall strategic plan to strengthen science and technology K-12 education in Israel [13]. The program (180 hours: 60 hours a year, two hours a week) is intended for students from the seventh to the ninth grades. With regard to the high-school curriculum, a six-year sequence of the curriculum is created. The main goal is to impart knowledge and skills required for a person in the twenty-first century. It is not aimed at training students to be programmers or computer scientists, but rather to introduce students to logical and algorithmic thinking and to expose them to different development environments at an early stage. A somewhat similar approach has been recently suggested by the American Computer Science Teacher Association (CSTA) in their Level I model curriculum for K-12 Computer Science, in the parts that discuss K5-K8 [5].

Lowering CS contents from high school to middle school will enable excellent students from the academic track to complete the high-school curriculum a year earlier, during the eleventh grade. That will allow such a student to be exposed (if interested) during the twelfth grade to one of the specialized areas in the SE program taught in the technological track, and to develop a software project.

The first module constitutes the core of the whole program. Its main goal is to expose the students to the fundamentals of computational thinking and programming. The subjects include serial execution, variables, conditions, loops, counters, accumulators, messaging, and event handling. Since this is the first year the students study CS at school, it was important to choose a suitable programming environment that will:

- Expose the students to algorithmic problems and their solutions and enhance algorithmic thinking.
- Enable students to implement various control structures.
- Make programming enjoyable, interactive, easy to use, and graphically appealing.
- Be translated to different natural languages.
- Be free of charge (if possible).

Scratch (http://scratch.mit.edu) was the chosen environment. Other alternatives that exist worldwide include Logo-based environments, Alice (http://www.alice.org/),Greenfoot (http://www.greenfoot.org/book/), and more recently Bootstrap, "a standards-based curriculum for middle and high-school students, which teaches them to program their own videogames using purely algebraic and geometric concepts" (http://www.bootstrapworld.org).

The second year begins with introducing the students to using a spreadsheet for scientific research (***second module***). Teaching a spreadsheet is required for the mathematics and physics curricula. Hence, its inclusion in the CS curriculum helps to create a bridge between the curriculum in middle school and the general six-year STEP. The tools to be taught include representation of graphs, using mathematical and statistical functions, and wise use of conditionals.

The third module is elective and its guidelines are as follows:

- It is based on the knowledge taught during the first year.
- No new control structures are introduced
- Students are exposed to new kinds of algorithmic problems and new technologies.

It was decided that in the second year the main subject will be *Introduction to Robotics*, focusing on algorithmic problem solving and not (just) on the mechanical and electrical aspects. Students receive a ready-to-program robot and can add to it various sensors and download their software. The goals of adding robotics to CS curricula are to (1) combine logical thinking with engineering thinking, (2) expose students to other technological areas, and (3) stimulate the students to be independent learners. The module contains the following topics: controllers, actuators, sensors, electrical energy and mechanical energy, energy transformations, motors, an open-loop control, and a closed-loop control. Students in the program can compete in FLL competitions (http://www.firstlegoleague.org/).

Since the *Introduction to Robotics* module is budget dependent (which might be a problem) and because those teachers without an engineering or electrical engineering background were reluctant to teach this program, the authors decided to suggest an alternative module that is less engineering oriented and free of charge. In order to create more continuity between the middle-school and the high-school curricula, it was decided to suggest teaching *Basic Internet Programming* by focusing on client-side programming as an introduction to stateless programming (taught in an elective module in high school). The module focuses on HTML5 and JavaScript; the students practice conditionals and loops.

The fourth module, which is taught during the third year of study, is devoted to developing a programming project including writing a project proposal, modeling a problem, designing a solution, and implementing it. The teachers can choose the development environment for their students. The programming project helps students internalize the use of algorithms for solving problems and prepares them for further studies in high school.

IV. Implementing The New Program

The program has been implemented in stages for the past two years. At the first stage 30 schools were selected to participate in a pilot program, most of which are located in the periphery of the country in order to promote a segment of the population that is less accessible to educational resources. Twenty-seven of them (709 students) continued to the next stage the following year.

At the second stage 183 middle-schools (5696 students) participated in the program. In each school, the students that were chosen for the program excelled in their age group. At the third stage about 100 additional schools are planned to join the program the following year (2013). The total number of teachers who teach the CS program this year is 172.

Teachers constitute the cornerstone of any curriculum [4,8,11]. Successful implementation of a new curriculum greatly depends on the pedagogical and content knowledge of the teachers as well as their satisfaction from the ongoing training and the support offered by the curriculum's initiators. Prior to the development of the curriculum presented here, no formal CS program was available for middle schools in Israel. Accordingly, recruiting and retraining teachers for the new program has been challenging but rewarding.

A. *Preliminary criteria for approving teachers to teach the CS program*

Initially, the Ministry of Education in Israel decided that the new STE curricula will only be taught by experienced and professional teachers. The rationale behind this decision is that qualified teachers should exhibit the following general qualities: (a) possess at least a Bachelor's degree in CS or engineering and a teaching certificate, and (b) have experience (of at least 3 years) in teaching the CS program at the highest level and in successfully preparing students for the high-school matriculation exam.

Since the criteria plan was restrictive, its implementation produced a shortage of qualified teachers. Academic retraining courses for prospective teachers with professional hi-tech experience were established to alleviate teacher shortages. Teachers having a background as high-tech professionals were assumed to have an additional advantage of encouraging students to study sciences, especially computer science.

B. *Difficulties in assigning teachers*

Shortly before the school year began, it became clear that assigning qualified teachers is problematic for the following reasons: (a) Qualified CS teachers who previously taught high-school students felt uncomfortable and even refused to teach middle-school students since:

- The emotional needs of the younger students were unclear to them;
- Teaching skills at the middle-school level seemed foreign to these teachers;
- Different physical locations of the middle-school and the high school complicated their work day logistically;
- Middle-school computer labs have a limited number of computers compared with high-school labs.
- The teachers needed to prepare lesson plans and teaching materials for the new program.

(b) The Ministry of Education had to assign tenured teachers to the program, even though they were not qualified for it.
(c) In schools that had not established a computer-related program and thus had no qualified CS teachers, it was difficult to find qualified teachers in the surrounding area, or to find a suitable teacher who would agree to come on-site and teach only two hours a week.
These kinds of difficulties were also encountered with Math teachers, but mostly in CS and Physics, since it was the first time those subjects were taught at middle school.

C. *Reducing the criteria for qualified teachers*

Owing to the difficulties in finding qualified teachers who could teach the program, it was decided to relax the criteria for accepting teachers, and to develop training courses for them. Teachers who did not meet the original requirements were permitted to teach the program provided that they agreed to participate in a suitable course. Relaxing the professional criteria resulted in accepting to the program less qualified teachers, for example, CS teachers who had previously taught only the lower levels in high school, qualified and experienced science/electronics teachers who had some CS education but who had not taught CS so far, students who were in their last year of academic CSED studies, and Computer Literacy teachers.

In implementing the program, several difficulties were encountered:

- Experienced high-school teachers were able to cope with the challenge of teaching excellent students but were not accustomed to teaching younger students.
- New teachers faced typical difficulties of entering the education system.
- New teachers and teachers with no CS background often taught at the technical-applicative level and did not focus on the program's algorithmic requirements.

D. *Training Courses*

Teachers are obligated to participate in a training program that was designed to provide them with pedagogical tools for enhancing their students' algorithmic thinking. The training consists of several courses, each of which is related to a specific module of the program. Additional course in Java and C# extends teachers' knowledge in order to give them an idea of what direction the students are heading to in high school. Each course lasts approximately 3 months. The courses are taught both in a computer lab and in an online environment to ease the teachers' burden. There are three or four lectures per course and coursework is assigned weekly via a website.

E. Teachers' Support System

Reducing criteria implied that teachers constitute a heterogeneous group, with different backgrounds and knowledge. In addition, the fact that the teachers are physically scattered throughout the country made it difficult to support teachers and to arrange face-to-face (F2F) meetings. Therefore, there was an urgent need to create a supporting system that could overcome these constraints. In the beginning, the Ministry of Education program coordinators communicated with the teachers mainly through emails and phone conversations; therefore, a forum designed for the teachers in the program was established. The forum is mostly used for interacting with the program's coordinators. In addition, a blog was established in order to manage the distribution of instructional materials, either those developed by professionals or materials developed by teachers in the training courses that were found appropriate for distribution to other teachers.

F. Preliminary Evaluation

The program will be evaluated at two levels. One is by administering a nationwide exam aimed at assessing the students' understanding of the material taught. The first exam was administered at the end of the first year of the program and a preliminary evaluation of it is described in [13]. The second evaluation is planned to take place at the end of this year. The other level concerns the teachers.

Teachers completed a Likert-type questionnaire assessing their perception of the program (1 (strongly disagree) - 5 (strongly agree)). The questions evaluated the extent of students' internalization of programming structures and algorithmic thinking in their teachers' eyes (Table 1), and teachers' sources of support, and their general satisfaction with the program (Table 2). Sixty teachers completed the questionnaire; fifty of them taught the program to 7th grade classes for the first time, whereas the other ten teachers taught the program for the second year, in both the 7th and 8th grades. In addition to the questionnaire, personal conversations with teachers were conducted.

TABLE I. TEACHERS' ATTITUDES REGARDING STUDENTS' LEARNING

The Statements:	***1st year teachers (N=50)***	***2st year teachers (N=10)***
The program is interesting	4.5	4.6
The program promotes students' algorithmic thinking	4.2	4.1
Students master conditional statements	4.2	4.5
Students master loop statements	4.1	4.3
The Algorithmic module contributed to students' algorithmic thinking	4.3	4.2
The Robotics module contributed to students' algorithmic thinking		3.8

TABLE II. TEACHERS' ATTITUDES REGARDING THE PROGRAM'S SUPPORTING TOOLS AND THE TRAINING COURSE

I was assisted by:	***1st year teachers (N=50)***	***2st year teachers (N=10)***
Colleagues who teach the program	3.2	3.0
Other teachers	2.5	3.3
The program's blog	4.1	4.0
The teachers' forum	3.0	3.6
The program's supervisors	3.1	3.6
The materials developed for the program	4.1	4.2
Comments regarding the training:		
F2F meetings may be given up	1.9	2.9
I've developed additional materials based on examples I've seen in the course.	3.7	3.1
I would recommend my colleagues to join the training course.	4.3	4.1

According to Table 1, the teachers were highly satisfied with their students' learning outcomes. According to Table 2, the blog and the program materials are the teachers' most appreciated support tools. Despite the differences between the original program design and the actual situation, providing virtual support tools in addition to F2F meetings, established during the pilot stages, show that teachers were satisfied with the course plan and that their ability to use remote supporting tools is increasing. Teachers need fewer F2F meetings, and are able to study through consulting and by using virtual support tools. Apparently collaboration tools contributed mainly to retrieving more instructional materials and less to maintaining ongoing communication among the teachers themselves.

Additional information gathered from personal conversations and documentation of the difficulties encountered indicated that there was a great diversity of teachers' content knowledge and pedagogical knowledge. In addition, teachers' perceptions of the goals of the program and its feasibility in teaching middle-school students differed. For example, teachers expressed significantly different perceptions regarding the need for an informal, game-like learning approach, compared with the desire to move CS contents from high school to middle school.

V. DISCUSSION AND CONCLUDING REMARKS

The *Science and Technology Excellence* program for middle school, which includes the new CS program described here, is an educational initiative that aims at motivating and encouraging highly capable young students to choose science and technology studies in high school and academia. It is based on the assumption that early exposure to science and technology is a critical factor in attracting youngsters to these fields and in adequately preparing them as well as sowing the seeds for the development of the next generation of scientists and engineers.

The program is based on expanding the scope of math and science studies, beyond what most middle and high-school students learn today, and to new areas of science and technology that young students are not usually exposed to in the traditional and existing curricula. Computer Science integrated with Robotics was chosen as one of the main scientific-technological areas to be included in the program.

Until recently, no official CS curriculum by the Israeli Ministry of Education has been tailored to the needs and capabilities of seventh to ninth graders, which also takes into consideration the background and expertise of most middle-school teachers.

Lowering CS content levels from high school to middle school will enable excellent students from the academic track to complete the high-school curriculum a year earlier and during the last year of study to get a taste of the SE curriculum in the technological track, and to develop a software project. The aspiration to recreate for outstanding students a sequence of six-year high-level studying of computer science necessitated the construction and operation of a formal curriculum for the lower grades. The new program emphasizes the gradually building of basic concepts and principles in computer science, the development of logical reasoning and computational-algorithmic thinking, coping with the cognitive challenges of problem solving, exposure to the processes of software project development and the development of students' creativity skills. Achieving these goals is fostered by familiarizing the young students with several learning environments where these concepts and principles can be identified and elaborated.

The cornerstone of implementing a new educational idea or a program lies in the teachers; therefore, at this stage of implementing the program, we focused on in-service training and on evaluating the process that the teachers underwent and their feelings and attitudes at the end of one or two years of experience.

It was realized that the original plan of setting high criteria standards for approving teachers to teach the program was unrealistic; this resulted in reducing the professional criteria of acceptance to the program and the training courses; still, the formation of an array of courses and support tools during the first two years of implementation evidently helped those teachers with different backgrounds.

The information gathered in the preliminary assessment indicated that teachers' content knowledge and pedagogical knowledge were very diverse, as were their perceptions of the goals of the program and its feasibility in teaching middle-school students. They reflected on their satisfaction with their students' achievements and the available supporting tools. Noteworthy are the improved attitudes of those teachers who taught the program the second time regarding the program’s potential to teach problem solving and to develop algorithmic thinking among young students.

A main conclusion to be deduced from this preliminary study is that building a properly tailored training courses for a heterogeneous group of teachers, as well as diverse supporting tools and suitable guidance, mostly on the web, contributes to the professional development of teachers and enables bridging the pedagogical and content gap between the desired and the actual availability of qualified teachers.

Future work will further examine the relationship between teachers' backgrounds and how they deal with the program's instruction, and the effect of teaching the entire 3-year middle-school program on teachers' perceptions of the program and its implementation. In addition, students' achievements will be evaluated as well as the percentage of students who choose to study computer science in high school.

References

[1] A. Cohen, and B. Haberman, “Computer science- A language of technology, ” Inroads SIGCSE Bulletin, 2007, vol. 39(4), pp. 65-69.

[2] A. Cohen, and B. Haberman, “CHAMSA: Five languages that citizens of an increasingly technological world should acquire Inroads,” SIGCSE Bulletin, 2010, vol. 1(4), pp. 54-57.

[3] P. J. Denning, “Great principles in computing curricula. Proceedings of SIGCSE'04, Norfolk, Virginia, USA, 2004, pp. 336-341.

[4] B. Ericson, M. Armoni, J. Gal-Ezer, D. Seehorn, C. Stephenson, and F. Trees, “Ensuring Exemplary Teaching in an Essential Discipline: Addressing the Crisis in Computer Science Teacher Certification.” Final Report of the CSTA Teacher Certification Task Force, September 2008, http://csta.acm.org/ComputerScienceTeacherCertification/sub/CertificationResearch.html [Accessed April 2012]

[5] D. Frost, A. Verno, D. Burkhart, M. Hutton and K. North, “A Model Curriculum for K–12 Computer Science Level I: Objectives and Outlines, ” 2009, http://csta.acm.org/Curriculum/sub/CurrFiles/L1-Objectives-and-Outlines.pdf [accessed June 2012]

[6] Futschek, G. “Algorithmic Thinking: The key for Understanding Computer Science, ” ISSEP, 2006, LNCS 4226, pp. 159-168.

[7] J. Gal-Ezer, and D. Harel, “Curriculum and course syllabi for a high school CS program,” Computer Science Education, 1999, vol. 9(2), pp. 114 -147.

[8] J. Gal-Ezer, and C. Stephenson “Computer Science Teacher Preparation is Critical, ” ACM Inroads, 2010, vol. 1(1), pp. 61-66.

[9] B. Haberman, and A. Cohen, “A high-school programme in software engineering, ” Int'l. J. of Engineering Education, 2007, vol. 23(1), pp. 15-23.

[10] O. Hazzan, and T. Lapidot, “The practicum in computer science education: Bridging the gap between theoretical knowledge and actual performance, ” SIGCSE Bulletin, 2004, vol. 36(4), pp. 47–51.

[11] C. Stephenson, J. Gal-Ezer, B. Haberman, and A. Verno, “The new educational imperative: Improving high school computer science education. Final report of the CSTA Curriculum Improvement Task Force February 2005, ” 2006, http://csta.acm.org/Publications/White_Paper07_06.pdf [Accessed April 2012]

[12] J.M. Wing, “Computational thinking. Communication of the ACM,” 2006, vol. 49(3), pp. 33-35.

[13] I. Zur Bargury, “A new Curriculum for Junior-High in Computer Science,” in press.