A Model for High School Computer Science Education: The Four Key Elements that Make It!

Orit Hazzan
Dept. of Education in Technology & Science
Technion – Israel Institute of Technology
oritha@tx.technion.ac.il

Judith Gal-Ezer
Computer Science Department
The Open University of Israel
galezer@cs.openu.ac.il

Lenore Blum
Computer Science Department
Carnegie Mellon University
lblum@cs.cmu.edu

ABSTRACT

This paper presents a model program for high school computer science education. It is based on an analysis of the structure of the Israeli high school computer science curriculum considered to be one of the leading curricula worldwide. The model consists of four key elements as well as interconnections between these elements. It is proposed that such a model be considered and/or adapted when a country wishes to implement a nation-wide program for high school computer science education.

Categories and Subject Descriptors

K.3.2 [*Computers and Education*]: Computers and Information Science Education - CS education

General Terms

Management, Human Factors.

Keywords

Computer science education, curriculum, teacher preparation programs, teaching certificate, computer science education research.

1. INTRODUCTION

In the Final Report of the ACM K–12 Task Force Curriculum Committee ([19]), the Israeli high-school computer science (CS) curriculum is mentioned to illustrate the fact that “the development of K–12 computer science is making more headway internationally than in the United States.” The report continues:

“In Israel, a secondary school computer science curriculum ([9]) was approved by the Ministry of Higher Education and implemented in 1998. It blends conceptual and applied topics, and is offered in grades 10, 11, and 12. ” (p.6). The curriculum comes in two versions, one for those who have only a limited interest in CS (the 3-unit version) and one for those who have much more interest in CS and might even pursue higher education later on (the 5-unit version). The two versions have a particularly heavy emphasis on the foundations of algorithms.

Our paper suggests that this description of the Israeli high school CS curriculum, despite the fact that it points to some of its main characteristics, *does not* capture the entire richness of the Israeli high school CS program. Accordingly, based on an analysis of this program, our paper presents a model for high school CS education. The model consists of interrelationships among four key components:

- A well-defined curriculum (including written course text books and teaching guides).
- A requirement of a mandatory formal CS teaching license.
- Teacher preparation programs (including at least a Bachelors degree in CS and a CS teaching certificate study program).
- Research in CS education.

It is proposed that each of these components, as well as the relationships among them, establishes the solid infrastructure of the Israeli high school CS program, strengthens it and makes it, as is indicated by the ACM K–12 Task Force Curriculum Committee report, one of the leading CS high school curricula in the world. To establish this proposition, this paper elaborates on how the *amalgamation* of these four components is, in fact, the factor that dominates the strength of the Israeli high school CS education system.

That said, we should remember that Israel is a relatively small country (with only about 7 million citizens) and with a central educational system. These circumstances, of course, have benefits as well as pitfalls. One benefit of Israel’s small size as well as of its central education system is expressed by the establishment of a National CS Teacher Center (the center’s website is: http://cse.proj.ac.il/). This national center is considered as the professional home for all Israeli CS teachers; it supports their teaching and ongoing training by forming learning communities and organizing different activities (such as an annual conference and workshops about specific ad-hoc and timely topics). As a pitfall of the Israeli central education system, we mention here that this structure forces the entire system to be oriented towards the matriculation exams that students have to pass in order to graduate from high school. This, unfortunately, can limit teachers’ creativity. However, as outlined in this paper, the central structure of the Israeli educational system has positive influences with respect to high school CS education.

In the following sections, we first provide a brief background about high school CS education in the world, in the US, and in Israel (Section 2). Then, in Section 3 we present our proposed model for high school CS education. In Section 4 we summarize and present relevant questions based on the discussion presented in the previous sections.

2. BACKGROUND

2.1 High school CS education in the world

A growing number of countries (e.g. Canada, Finland, Germany and Lithuania) already claim to require CS education for all high school students. In many cases, however, the programs are too narrowly focused on programming and therefore do not give an appropriately broad view of the discipline.[1] An examination of the experiences of an international panel of CS educators from Canada, Israel, Scotland, South Africa, and the United States ([18]) supports the argument that such curriculum initiatives will be successful only to the extent to which they meet the following criteria:

- There is a link between the outcome required and the strategies used.
- Change is driven by real learning needs and not politically manufactured needs.
- Educational change must be seen in the context of larger social and economic forces.
- All of the stakeholders must agree to the need for change and on the strategies put in place to achieve it.
- Change requires the commitment of adequate resources through all phases of the design, implementation, and testing of the new curriculum.
- Change is a long-term process, not a short-term intervention.

As countries grapple with CS curricular implementation and reform, we believe they will be well served to heed these guiding principles.

The United States is a particular case in point. Stephenson *et al.* ([18]) report that despite the fact that computers have infiltrated all areas of society, and there is now a clear link between technology, innovation, and economic survival, no U.S. national K–12 CS educational program exists. Lack of leadership in high school CS education at the highest legislative and policy levels has resulted in insufficient funding for classroom instruction, resources, and professional development for CS teachers. In addition, complex and contradictory teacher certification requirements, as well as salaries that cannot possibly compete with industry, make it exceedingly difficult to ensure the availability of exemplary CS teachers.

Concern over this state of affairs—as well as concern about the apparent decrease in interest amongst students in the US to study CS—has prompted the development of programs to tackle various facets of the problem. For example, in the professional arena, the Computer Science Teachers Association (CSTA), "the Voice for K-12 computer science education and its educators," was launched in 2005 by the ACM to support and promote the teaching of CS and other computing disciplines (see: http://csta.acm.org/). At the state level, the Institute for Computing Education (ICE), created in 2004 as a partnership between the State of Georgia Department of Education and the College of Computing at Georgia Tech, aims to increase the number and quality of CS teachers and diversity of CS students in the state ([6]). The CS4HS summer workshops, initiated by the School of Computer Science at Carnegie Mellon and now joined by a network of universities nation-wide (including UCLA and the University of Washington), provide high school teachers instruction and resources to help them teach CS principles ("computational thinking") to their students in a fun and relevant way ([4] and see: http://www.cs.cmu.edu/cs4hs/). Recent faculty summits at industrial centers, like Microsoft and Google, sponsored panels addressing CS educational issues. But still, a comprehensive nation-wide initiative in the US is sorely lacking.

2.2 High School CS education in Israel

In 1990 a professional committee was appointed by the Israeli Minister of Education with the goal of putting together a rigorous and coherent high school CS curriculum. This curriculum was implemented in Israel in 1995 and will soon be updated.

The principles that guided the work of the committee, and still stand today, are as follows:

- *Computer science is a full-fledged scientific subject.* It should be taught in high-school on a par with other scientific subjects.
- *The program should concentrate on the key concepts and foundations of the field.* The main notion to be emphasized throughout the program is that of an algorithmic problem and an algorithm as a solution. To some extent, the more general notion of a system, and the accompanying principles of modularization and abstraction, should also be discussed.
- *Two different programs are needed, one for students with only a general interest in CS, and the second, which should be deeper and broader, for those with more special interest in CS.*
- *Each of the two programs should have mandatory units and electives.*
- *Conceptual and experimental issues should be interwoven throughout the program.*
- *Two quite different programming paradigms should ideally be taught.*
- *A well-equipped and well-maintained computer laboratory is mandatory.*
- *New course text books and teaching guides must be written for all parts of the program.*
- *Teachers certified to teach the subject must have adequate formal CS education.* An undergraduate degree in CS is a mandatory requirement, as is a formal teachers' certificate program of studies.

Further details about the curriculum are presented in [7] and [9].

3. PROPOSED MODEL FOR HIGH SCHOOL CS EDUCATION

Figure 1 presents the model which reflects our perspective of the current situation of the Israeli high school CS education program. As has been mentioned earlier, and as can be seen in Figure 1, it consists of more than just the curriculum itself. Further, as the following discussion illustrates, the success of the Israeli high school CS education may be explained, at least partially, by that

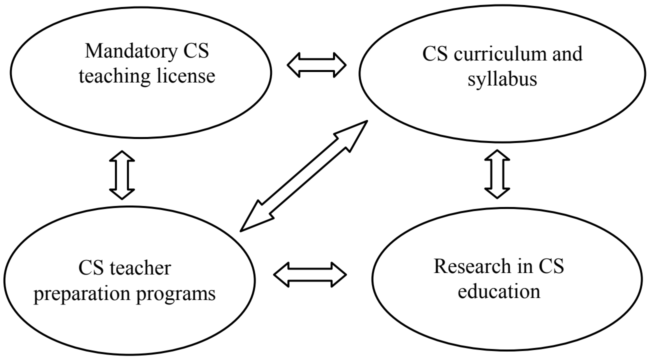
[1] This claim is based on an extensive literature review conducted by Dr. Noa Ragonis as background for the new Israeli CS program committee. An English version of Dr. Ragonis' report is anticipated.

more complex picture evolved in Israel, which consists of the interrelations of several components, elaborated in what follows.

3.1 The model components

The model is based on four components: A well defined high school CS curriculum (including text books and teaching guides), a mandatory CS teaching license, teacher preparation programs (and in-service training), and research in CS education.

Figure 1: A model for a high school CS education program


Mandatory CS teaching license
CS curriculum and syllabus
CS teacher preparation programs
Research in CS education

In what follows, the main characteristics of each component are outlined. Then, connections among these components are explained.

National high school CS curriculum: This national curriculum describes the rationale for and structure of CS education in high school and the contents of each of its (curricular) units.

The current Israeli high school CS curriculum is modular and enables each student to learn and take the matriculation exam at the level that fits his or her skills. The basic level 3-unit track was designed for students with only a general interest in CS; the 5-unit track, designed for those with more interest in CS, is deeper and broader. However, even the basic track exposes the students to two programming paradigms and emphasizes both theoretical and practical aspects of CS.

The topics taught in each unit of the program are specified. Text books and teaching guides are provided for each unit. They specify pedagogical aspects of the given topics, such as, recommended lesson plans, additional problems to offer to the pupils, plausible learners' difficulties, and additional clarifications related to the material.

The 3-unit track consists of 270 hours of study while the 5-unit track consists of 450 hours. The programs are constructed from five modules taken by students over a 2-3 year period, starting in 10th or 11th grade, depending on the school. We present a brief outline of each of the modules and then indicate how they fit into the two tracks.

- *Fundamentals 1 and 2* (2 units): Introduces the central concepts of solving algorithmic problems and teaches how to apply them in a programming language.
- *Software Design* (1 unit): Concentrates on data structures, introduces abstract data structures and discusses the design of complete systems.
- *Second Paradigm* (1 unit): Introduces a second programming paradigm. Logic programming, functional programming and system-level programming are three of the current possibilities.
- *Applications* (1 unit): Focuses on one particular application, emphasizing both theory and practice. Current possibilities are computer graphics, management information systems and Internet programming.
- *Theory* (1 unit): Exposes students to selected topics in theoretical CS. One of the current possibilities is models of computation, mainly finite automata.

The two *Fundamentals* are mandatory for both tracks. Students in the 3-unit track can then choose to take either the *Second Paradigm* module, or the *Applications* module. *Software Design* is mandatory for the 5-unit students, as is the *Theory* module. They too can choose either the *Second Paradigm* or the *Applications* module. More detailed information can be found in [9].

Another nation-wide element of this curriculum is the matriculation exam that sets teaching standards and clarifies explicitly what pupils should learn.

This program has been in place for about a decade. Though it is considered to be a leading program world-wide, a new committee has been appointed recently by the Israeli Ministry of Education to scrutinize the current curriculum and to see how it correlates with the massive changes and current developments of the field of CS itself.

Mandatory CS teachers' license: In Israel, in order to teach CS in high school, a teacher should have both a Bachelors degree in CS[2] and a teaching license in CS. Only then will he/she be authorized by the Ministry of Education to teach CS in high schools. The CS teaching diploma can be achieved in two different frameworks:

a) Programs such as the ones described in the "Teacher preparation programs" component of the proposed model;

b) Specific training programs, offered to CS graduates without a teaching license and to teachers of another scientific topic who wish to switch to CS education. These programs are usually offered after school hours or during vacations.

Teacher preparation programs: In most cases teacher preparation programs are taught in universities or colleges. The prospective CS teachers study for a Bachelors degree in CS while at the same time take teacher preparation program courses (which are equivalent to one academic year) during the four years of

[2] In fact, a Masters degree is required for teaching any subject in the Israeli high school system. It is, however, difficult to meet these requirements and in most cases high school teachers only have a Bachelors degree. Unfortunately, and as it happens in other countries, in the case of CS, some teachers, mainly those who joined the system many years ago, do not have even a Bachelors degree in CS. They usually have a Bachelors degree in another scientific subject and switched to teaching of CS for different, administrative as well as personal, reasons. Today a teacher who joins the "club" must graduate with a CS degree from a university or a college.

study. The contents of these programs correlate with the statement that "beyond the mastery of core CS material, good CS educators should also be familiar with a significant body of material that will expand their perspectives on the field, and consequently, enhance the quality of their teaching." ([8])

A typical teacher preparation program includes general pedagogical courses (such as, psychology and educational philosophy), basic teaching skills, and specific courses about the teaching of CS.

We elaborate on the two main courses which deal specifically with the teaching of CS – *Methods of Teaching CS in the High School* and *practicum* in high school CS classes. (A detailed example of a teacher's preparation program is described in [12]).

The *Methods of Teaching CS* course can be based on different frameworks ([15]), such as the National Council for Accreditation of Teacher Education (NCATE) standards ([19]), the merger of CS with pedagogy, Shulman's model of teachers' knowledge ([17]), and research findings. It is important to note that the suggested frameworks and implementations are not limited to a particular curriculum, programming paradigm, programming language, or level of students. Additional details about the Methods of Teaching CS course can be found in [13, 15].

The *practicum* is carried out in several ways. Some programs require a full year's participation in school activities; others require that the practicum be performed for a specific, shorter period of time. In all these cases, however, the main objective of the practicum is to let the prospective teachers experience what real teaching is before becoming CS teachers. For more details about the CS practicum see [14].

We would add that the CS teacher preparation programs serve also in-service teachers by offering to the teachers ongoing training about the curriculum, teaching methods and the development of CS as a scientific field. Indeed, it is reasonable to assume that it is easier to organize and facilitate trainings for in-service CS teachers within the existing infrastructure of the CS teacher preparation programs.

Research in CS education: Intensive research in CS education is carried out by Israeli researchers who are usually involved in the development of the text books and the teaching guides of the nation-wide curriculum. In many cases, this research is carried out during the development process of the material. The purpose of this research is to guide the development process of the text books so that the final product fits high school CS pupils' level and that it will be possible to teach it during the given period of time.

A typical research project that accompanies the development of new material usually involves both the development team and a group of teachers who agree to be the first to try the new material and to participate in research activities conducted with respect to that material. The research activities mainly include reflective talks about the teaching process, discussions about pupils' conception of the learned topics, interviews with pupils, and observations within the pioneer classes.

Such a research process is iterative; at each step lessons are learnt and are implemented in the next edition of the developed material. Also, during the first stages of the development process, the research focuses on small number of teachers and classes; as the development of the material proceeds, additional teachers join the group of teachers who teach the new developed material. Along such a cyclic process, the teacher and pupil populations who use the developed material gradually increase, and the text books and teaching guides are shaped and converged towards its final form.

As a bonus, this process enhances CS education research in Israel beyond just research directly connected to the evaluation of the course text books and teaching guides. This additional research addresses, for example, undergraduate students' understanding of CS concepts (cf. [1, 3 5). It is plausible to assume that the infrastructure needed for this additional research (expertise in education research as well as in CS itself) has been established and shaped by the need to evaluate the high school material during its development process.

3.2 Connections among the model components

Connections exist among the four elements of the proposed model for high school CS education. We outline several.

Mandatory CS teaching license ⇔ Teacher preparation programs: Clearly, a mandatory CS teacher license requires teacher preparation programs to be established to grant these certificates; on the other hand, when teacher preparation programs exist, it is just natural to establish and demand a CS teaching license and to exploit the experience and knowledge shared by these programs.

Teacher preparation programs ⇔ Research in CS education: Teacher preparation programs should include some research elements, such as reading assignments of papers which deal with CS education research, and mini-research projects, carried out by the prospective CS teachers themselves. Thus, instructors of CS teacher preparation programs should be familiar with research in CS education and thus the community of practitioners who are interested in CS education research enlarges. On the other hand, when the community of CS education researchers is well established, it naturally wishes to deliver its achievements to target audience. One appropriate framework for this purpose is teacher preparation programs.

Research in CS education ⇔ High school CS curriculum and syllabus: The existence of a CS education research community, with its accumulative experience, eases and guides the development process of the text books and the teaching guides. Further, from an organizational nation-wide perspective, the existence of a research infrastructure enables decision makers to promote the development of a curriculum and to allocate the needed resources for this purpose. This, naturally, boosts the curriculum development. On the other hand, it is critical to accompany the development of a high school CS curriculum and syllabus with research that both assesses the implementation of the curriculum and syllabus and evaluates the material developed. A few examples are presented in [2, 10, 11, 16].

One can easily give arguments for other direct mutual connections between the model components. Many indirect interconnections exist as well.

One compelling example is illustrated by interconnections related to Dalit Levy's Ph.D. research ([16]). Levy researched pupils' learning processes and social interactions, from pedagogical and cognitive perspectives, when dealing with a rich activity taught as part of the elective unit of the Israeli high school *CS curriculum* on functional programming. Since Levy participated in the unit development team, this *research* served the unit development process. As a rich activity that had been researched so intensively, it became part of the folklore of the Methods of Teaching CS course (at least the one that is taught at the Technion) that is taught as part of the *CS teacher preparation program*.

4. SUMMARY

The model proposed and presented in this paper is based on the Israeli experience with respect to the establishment of a nation-wide curriculum in CS for high school students.

In the talk we will present specific illustrations of the CS preparation programs that exist in Israel and present challenges still encountered. Naturally, it will be interesting to ascertain the model's applicability to other places on the globe. For this purpose, we present the following questions that will be open for discussion and criticism with the audience of our presentation at the conference.

- What can be learned from the Israeli experience? What components of the model can be applied to other countries? Is the model relevant at all for other countries?
- The model consists of four components which are interrelated to each other. It is not always clear which component is the chicken and which is the egg. If a country wishes to apply the Israeli model, how should it proceed?
- The model has benefits that were explained in the paper. What disadvantages does the model have?
- The model works well for *CS* education in high school. Does the model work well for other subjects? For other scientific subjects?

Clearly, these questions are also open and deserve further research.

5. REFERENCES

[1] Aharoni, D. (2000). Cogito, ergo sum! Cognitive processes of students dealing with data structures. In S. Haller (Ed.), *Proceedings of the Thirty-first SIGCSE Technical Symposium on Computer Science* Education, pp. 26–30.

[2] Armoni, M. and Gal-Ezer, J. (2003). Non-determinism in CS high-school curricula, FIE2003, http://fie.engrng.pitt.edu/fie2003/index.htm.

[3] Armoni, M., Gal-Ezer, J. and Hazzan O. (2006). Reductive thinking in computer science, *Computer Science Education* **16**(4), pp. 281-301.

[4] Blum, L. and Cortina T. J. (2007). CS4HS: An Outreach Program for High School CS Teachers, *Proceedings of SIGCSE 2007*, pp. 19-23.

[5] Dubinsky, Y. and Hazzan, O. (2005). A framework for teaching software development methods, *Computer Science Education* **15**(4), pp. 275-296.

[6] Ericson, B., Guzdial, M. and Biggers, M. (2007). Improving Secondary CS Education: Progress and Problems, *Proceedings of SIGCSE 2007*, pp. 298 – 301.

[7] Gal-Ezer, J., Beeri, C., Harel, D. and Yehudai, A. (1995). A high-school program in Computer Science, *Computer* **28**(10), pp. 73-80.

[8] Gal-Ezer, J. and Harel, D. (1998). What (else) should CS educators know?", Communications *of the ACM* **41**(9), pp. 77-84.

[9] Gal-Ezer, J. and Harel, D. (1999). Curriculum for a high school computer science curriculum. *Computer Science Education* **9**(2), pp. 114-147.

[10] Gal-Ezer, J. and Zeldes, A. (2000). Teaching software designing skills, *Computer Science Education* **10**(1), pp. 25-38.

[11] Gal-Ezer, J. and Zur, E. (2004). The efficiency of algorithms-misconceptions, *Computers and Education* **42** (3), pp. 215-226.

[12] Gal-Ezer, J. and Zur, E. (in press, 2007). Reaching out to CS teachers certification via distance learning, to be published in *Mathematics and Computer Education*.

[13] Hazzan, O. and Lapidot, T. (2004A). Construction of a professional perception in the "Methods of Teaching Computer Science" course, *inroads – the SIGCSE Bulletin* **36**(2), pp. 57-61.

[14] Hazzan, O. and Lapidot, T. (2004B). The practicum in computer science education: Bridging gaps between theoretical knowledge and actual performance, *inroads – the SIGCSE Bulletin* **36**(4), pp. 47-51.

[15] Lapidot, T. and Hazzan, O. (2003). Methods of Teaching Computer Science course for prospective teachers, *inroads – the SIGCSE Bulletin* **35**(4), pp. 29-34.

[16] Levy D. (2000). Classification and discussion of recursive phenomena by computer science teachers. In Robson R. (ed.), *Proceedings of the International Conference on Mathematics/Science Education and Technology* (M/SET 2000), San Diego, California.

[17] Shulman, L. S. (1987). Knowledge and teaching: Foundations of the new reform, *Harvard Educational Review* **57**(1), pp. 1-22.

[18] Stephenson, C., Gal-Ezer, J., Haberman, B. and Verno, A. (2005). *The New Educational Imperative: Improving High School Computer Science Education*, Final Report of the CSTA, Curriculum Improvement Task Force, http://csta.acm.org/Publications/CSTAWhitePaperNC.pdf.

[19] Tucker, A., Deek, F., Jones, J., McCowan. D., Stephenson, C., and Verno, A. (2003). *A Model Curriculum for K-12 Computer Science*. *Final Report of the ACM K-12 Task Force Curriculum Committee*. Retrieved February 20, 2007, from http://csta.acm.org/Curriculum/sub/K-12ModelCurr2ndEd.pdf.