# High school computing curriculum*

LEO BUDIN†

In the past, we can observe many controversies and confusion in the computing curricula planning and implementation, as well as in teacher preparation. Today, many teachers are still beginning their careers with almost no knowledge in computing. Some schools have carefully developed plans for computer use for many years, and others are just beginning to think about such a planning process. Goals for computing education vary widely among different schools and different countries. In recent years, information technology is recognized as the most pervasive generic technology, and the need for a new approach to education in the field emerged.

We will illustrate this new approach by quoting some sections from the paper "ACM Model High School Computer Science Curriculum" published in the May 1993 issue of the *Journal of the ACM*. The paper was signed by the *Task Force of the Pre-College Committee of the Education Board of the ACM*.

> "Computer technology has had a profound effect on our society and world. Every citizen needs some familiarity with this technology and its consequences in the home, school, workplace, and community. Because the details of the technology change from day to day, keeping up with those details is difficult and often unproductive. Therefore, the study of the subject must concentrate on the fundamental scientific principles and concepts of the field.
>
> In 1989 the Pre-college Committee of the Education Board of the ACM formed a task force to consider the college proposals and prepare recommendations for a new high school computer science course. The task force is composed of high school, college and university faculty members and computer administrators.
>
> The need for computer science education is similar to the need for education in the natural sciences. The natural sciences are studied by all high school students in order to understand the physical (tangible) world.
>
> As high school students study the natural sciences in order to understand the natural world. In the same way, they need to study computer science to comprehend the social, economic and cultural environment of the information age.

*The lecture presented at the MATHEMATICAL COLLOQUIUM in Osijek organized by Croatian Mathematical Society – Division Osijek, May 9, 1997.

†Faculty of Electrical Engineering and Computing, University of Zagreb, Unska 3, HR-10 000 Zagreb, e-mail: `leo@ZEMRIS.FER.HR`

> The current study of computers in high school is characterized by either the use of computers as a tool for other disciplines (word processing for English, spreadsheets and databases for business, CAD/CAM for technology education, Mathematica for mathematics and science) or programming. Neither of these is computer science, although both comprise aspects of the discipline. The study of computer science is composed of basic universal concepts that transcend the technology and comprise an essential part of high school education. It is these concepts that enable the student to understand and participate effectively in our modern world.
>
> The charge of the force is to develop a computer science curriculum for secondary schools. This curriculum takes the form of a comprehensive one-year computer science course. The intention is that this course be similar in scope, depth, breath and methodology to typical high school science courses. It should serve all students in the same way that introductory biology, chemistry and physics do. This course presents the background of the field, discusses important issues, studies and solves problems in the field, and applies mathematics to problem solving.
>
> The intended level for such a computer science course is approximately 10th grade. It can serve as a minimal requirement in itself or as a prerequisite for advanced computing courses, such as advanced placement computer science course. Student preparation should include first-year algebra and some computing experience."

The focus on the course is on fundamental concepts of computer science. Computer programming, application and literacy course that have evolved may be expanded to the computer science course by enchancing their existing contents or by combining them to contain the essential core topics.

The task force identified five basic areas that provide a broad introduction to computer science for high school students. These are: algorithms, programming languages, operating systems, computer architecture, social and ethical context. The basic areas have to be expanded by selected topics of computer applications, as shown in the following table:

| Core topics | Recommended topics | Optional topics |
|---|---|---|
| **Algoritms** | | |
| Algorithms in the world.<br>*Techniques used to design and represent the algoritms.*<br>Examples of important algorithms.<br>Basic problem solving concepts. | Methods to test algorithms.<br>*Basic data structures.*<br>Boolean algebra. | Characteristics of an algorithm (correctness, finiteness).<br>*Complexity of algorithms.*<br>Limits of computability. |
| **Programming languages** | | |
| Introduction to a specific computer language.<br>*Concept of sequence, selection and repetition.*<br>Levels of computer languages. | Competence in a high-level language.<br>*Compilers and interpreters.*<br>Program modularization.<br>*Reading and analyzing existing programs.* | Comparison of languages: procedural, structured, functional, object-oriented, parallel.<br>*Modifying existing programs.*<br>Program verification<br>*Theoretical machines and formal languages.* |
| **Operating systems and User Support** | | |
| Command language and its use.<br>*Files and disk management.*<br>Telecommunications, local and wide area networks | Human-computer interaction.<br>*Working with large complex systems.* | Communication network implementation.<br>*Memory management and virtual memory.*<br>Operating systems functions (e.g. task scheduling, interrupts, buffered I/O).<br>*Single and multiuser machines.* |
| **Computer architecture** | | |
| Basic Computer model (e.g. CPU, memory, IO).<br>*Basic data representation: numbers vs. characters, ASCII vs. non-ASCII.* | Logic, gates and circuits.<br>*Data representation (e.g. bits and bytes, binary numbers, real numbers).*<br>Accuracy of numerical computation.<br>*Von Neumann stored program model, opcodes, registers, clock, fetch-execute cycle.* | Physical disk organization.<br>*Sequential and parallel processing.*<br>Special data representation (e.g. graphics, sound).<br>*Data compression.* |
| **Social, Ethical and Professional Context** | | |
| Impact of technology on today's society.<br>Ethics in electronic community.<br>Team solution of problems. | Future of computer technology.<br>*Risk and liability in computing viruses.*<br>Computer support of disabled.<br>*Software, public domain private.*<br>Privacy, reliability and system security.<br>*Uses, misuses, and limits of computer technology.*<br>Electronic crime: stealing and spying.<br>*Intellectual property, infringement and penalties.* | Legal issues. |

Recognizing the diversity of school systems, teachers and students, the task force identified several different approaches for presenting these computer science areas. The topics of the five basic areas are divided into three levels: core topics, recommended topics and optional topics (the most important of them shown italicized). Each approach should cover core topics, with students working on examples, projects and reports. The teacher should introduce as many of the recommended topics as possible, some possible in a survey level. Other recommended and optional topics should be covered as appropriate. This model curriculum seems very appropriate, and we can expect that it will stimulate an increasing trend to thoroughly integrate, in a proper way, the computer science in the high-school education.