A SET OF WORKSHOPS FOR HIGH SCHOOL COMPUTER SCIENCE TEACHERS

Dr. George M. Whitson III
Computer Science Department
The University of Texas at Tyler
Tyler, Texas 75701

ABSTRACT: This paper outlines a set of workshops to provide training for certified high school computer science teachers. Upon the completion of the four core workshops, a high school teacher would have an excellent background to teach high school computer science as detailed in the new proposed ACM Curriculum for high school certification. The workshops should also do a good job upgrading the background of high school computer science and computer math teachers to teach courses currently in the high school curriculum. One workshop, PASCAL with Applications to Data Structures, was specifically designed to prepare current high school teachers to teach a PASCAL course whose goal is to prepare students for the advanced placement test. Each of the six workshops is a three semester hour course and most carry graduate credit. Three of the six courses have already been offered and more should be taught next summer. The reception of the high school teachers to the workshops has been very enthusiastic.

Introduction

The Proposed ACM High School Certification Recommendations (3) should set the standard for future training of high school teachers of computer science. There are many teachers who are currently certified and did not complete some of the courses in the ACM Recommendations. Many of these teachers are very interested in taking courses of a workshop nature that are equivalent to courses of the recommendations. These high school teachers also desire courses taught on microcomputers since this matches the equipment they use in their teaching. In response to this need we have introduced a set of workshops and courses at The University of Texas at Tyler to provide much of the content of the courses in the ACM Recommendations. Two of these were offered in the summer of 1985, one will be offered in the spring of 1986 at night, and we hope to offer one or two more in the summer of 1986. Based on a profile of the teachers attending the summer courses and their evaluation of the courses, I think we have been successful in achieving our goals of:

1. Teaching courses with the content of the ACM Recommendations.
2. Attracting currently certified high school teachers to the program.
3. Teaching these courses as regular courses in our undergraduate and graduate programs.

Many of the high school teachers that wanted to upgrade their computer science background were also teaching mathematics in high school. Since the high school math teachers have considerable interest in using the computer as a tool to teach mathematics or making writing of programs an integral part of teaching mathematics, we designed some of the courses to meet the need of both Computer Science and Mathematics teachers.

The ACM Proposed Curriculum consists of the following required and elective courses:

 ACM-0-89791-178-4/86/0002/0303 $00.75

R1. Computer Science I
R2. Computer Science II
R3. Introduction to Computer Systems
R4. Organization of Computer Languages
R5. Data Structures and Algorithm Analysis
R6. Methods for Teaching Computer Science

E1. Computers in Education
E2. Computer-Assisted Instruction
E3. Introduction to File Processing
E4. Fundamentals of Computer Organization

Many of these are equivalent to courses in the ACM 78 Undergraduate Curriculum and high school teachers could take the corresponding CS 1 to CS 8 courses of this curriculum to learn the appropriate material. In fact, our workshop courses cover most of the material of the corresponding CS 1 to CS 8 courses. But we think that our program is better for training experienced high school teachers than putting them in the appropriate CS 1 to CS 8 courses for the following reasons:

1. All programming assignments are done on microcomputers (Apple 2e and IBM PC).
2. Since most, but not all, of the students in the workshops are teachers, there is a desirable interaction between the teachers from different schools. Also, they are the focus of the course which is not the case in the ACM 78 courses we teach.
3. Again, since most of the students are experienced teachers, some time can be spent in each course showing the relevance of the material in the course to the courses in the ACM Proposed High School Curriculum. We think this is very important and this opinion was shared by those students who have completed courses in the program.
4. Finally, we spend a small amount of time in each course discussing pedagogical problems associated with teaching computer science in general and high school computer science in particular.

The courses in our workshop series are:

(C1) PASCAL with Applications to Data Structures
(C2) Microcomputer Organization and Assembly Language
(C3) Microcomputer Software
(C4) LISP, Symbolic Computation, and the Theory of Programming Languages
(O1) Computer-Aided Instruction
(O2) Microcomputer Computer Graphics

The next section gives a more detailed outline of the courses with actual texts that have been or are proposed to be used.

Workshop Descriptions

In planning a set of courses and workshops, we operated with certain goals and constraints.

(a) Since many of the teachers taking these workshops were already certified, we needed to make the workshops timely and attractive so that teachers would want to take them.
(b) All courses would be taught on microcomputers. When possible, we would have software available for both IBM PCs and Apple 2es.
(c) We would teach four workshops as the main program. Certain courses that we regularly teach would also be recommended to teachers taking courses to upgrade their training.
(d) Some assembly language and computer hardware concepts would be presented to the students. We think this background is still important and place a higher value on this than the ACM proposed curriculum for certification appears to.
(e) We propose LISP and a study of symbolic computation as an excellent vehicle for presenting the concepts of a programming languages course. The new Common LISP, when contrasted to LISP 1.5, shows many of the ideas like data abstraction in a way that improves one's understanding of basic concepts. The development of a simple symbolic manipulation package in LISP is an excellent way to illustrate the problems of global variables, argument/parameter passing, and systems development.
(f) Most schools do not teach an advanced undergraduate course on software packages. We think that such a course should be taught and think all students of computer science should be encouraged to take this course as an elective. It is, of course, an excellent course for prospective and current high school teachers.
(g) We think that one can learn enough PASCAL and Data Structures in one semester to teach CS 1 and CS 2 of the new ACM proposed high school curriculum.

After considering the constraints and some of the solution ideas, we came up with the six courses in our retraining program. Four of these were central to the program, two would be helpful but not central to the program. We now look at course descriptions of these six courses:

(C1) PASCAL with Applications to Data Structures

Description This course introduces the student to the PASCAL programming language. Special attention is given to data types, subprograms, and pointer variables. The final third of the

course studies some topics in data structures selected from sorting (heap and quick), searching (hashing), and binary trees (searching).

Text(s) PASCAL (Micro Edition) by Koffman, Addison & Wesley 1982. Data Structures Using Pascal by Augenstein and Tenenbaum, Prentice-Hall 1981.

(C2) Microcomputer Organization and Assembly Language

Description This course introduces the student to the architecture of microcomputers. A detailed study of the IBM PC is given along with a short look at the Apple 2e. A number of programs will be written in a microcomputer assembly language: either 8088 or 6502. The hardware topics covered include computer arithmetic, CPU structure, bus concepts, memory, and I/O devices.

Text Microcomputers for Engineers and Scientists by Gibson and Liu, Prentice-Hall 1980.

(C3) Microcomputer Software

Description This course will introduce the student to the software that is available for use on microcomputers. The types of software will include integrated packages (spreadsheet, wordprocessor, database, graphics), statistical packages, decision support packages, and scientific packages (numerical analysis and symbolic manipulation). A number of projects will be given to illustrate the use of the software packages. The course is designed to give the student a balanced view of the package software available for microcomputers today.

Text Using Symphony by Ewing and LeBlond, Que Corporation 1984; plus manuals for the software packages studied.

(C4) LISP, Symbolic Computation, and The Theory of Programming Languages

Description This course will introduce the student to the LISP programming language. Both LISP 1.5 and Common LISP will be studied. The course will look at data types, data abstraction, subprograms, procedure abstraction, scope rules, control structures, and the programming environment. All of these concepts will be illustrated with LISP 1.5 and Common LISP. As an illustration of the power of LISP, a study of symbolic computation will be given as a part of the course. Students will write a number of programs including a small package to do some symbolic computation.

Text(s) LISP 1.5 Programmers Manual by McCarthy, et al, MIT Press 1976. LISP - A Gentle Introduction to Symbolic Computation by Touretzky, Harper & Row 1984; plus Manuals for MuMATH/MuSIMP, MuLISP, and Golden Common LISP.

(01) Computer-Aided Instruction

Description This course will introduce the student to the general theory of CAI as well as demonstrating a number of working CAI systems. Students will develop a CAI project in BASIC as a part of the course. Several specialized CAI languages will also be introduced.

Text CAI Sourcebook by Burke, Prentice-Hall 1982.

(02) Microcomputer Computer Graphics

Description This course will give the student an introduction to the hardware and software used to do computer graphics on microcomputers. A detailed discussion of the graphics capability of either the IBM PC or the Apple 2e will be given. A complete coverage of 2-dimensional graphics will be given as well as an introduction to 2½-dimensional and 3-dimensional graphics. Hidden line and surface algorithms will also be covered. A number of graphics packages will be demonstrated, including a CAD system, a 2-dimensional drawer, a 3-dimensional drawer, and a business-presentation package. Students will develop a small business package on a microcomputer using a team concept as well as write a number of short programs to illustrate graphics concepts. Both standard BASIC graphics and GKS graphics will be used.

Text(s) Microcomputer Graphics (PC & Apple editions) by Roy Myers, Addison & Wesley; or Interactive Microcomputer Graphics by Park, Addison & Wesley 1985; plus appropriate computer and software manuals.

Evaluation

It is always difficult to evaluate how well a set of workshops like ours have succeeded. The student evaluations and comments for the workshops given thus far have been very positive. It appears that a large number of high school teachers plan to take our future workshops. We have a list of those teachers who attended our workshops and plan some follow-up to see if the comments are as positive a year from now as they were immediately after the workshops ended. At present, I believe that our workshops have been successful at achieving their goals.

A survey was taken of the high school teachers in the PASCAL workshop. Among other items, we asked them to indicate which other workshops they most wanted to take and found that the most desired workshops after PASCAL were:

1. Microcomputer Software
2. LISP, Symbolic Computation, and The Theory of Programming Languages

I expected 1. but was surprised by 2. It turned out that most of the high school teachers wanted to study LISP because of its Artifical Intelligence connections. There was also considerable interest in symbolic computation (MuMATH/MuSIMP) since many of the teachers in our program had previously been (or still were) high school mathematics teachers. Depending on faculty availability and budget, I hope we can offer both of these workshops next summer. The LISP workshop would be offered in a three-week format and the microcomputer software in a five-week format. As teachers finish a larger number of the six workshops, we hope to get a better measure of exactly how well the workshops have achieved their goals.

References:

1. ACM Curriculum Committee on Computer Science. "Curriculum '78--Recommendations for the undergraduate program in computer science", Commun. ACM 22,3 (Mar. 1979) 147-166.

2. ACM Curriculum Committee Task Force for CSI, "Recommended Curriculum for CSI, 1984", Commun. ACM 27,10 (Oct. 1984) 998-1001.

3. ACM Task Force on Secondary School Curriculum and ACM Task Force on Teacher Certification, Curricula Recommendations for Secondary School and Teach Certification, ACM Publication, 1985.

4. Hartog, Curt, "Of Commerce and Academe", Datamation, Vol. 31-Number 17, pp 68-78.