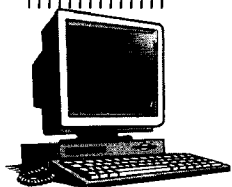


History in the Computer Science Curriculum

John A. N. Lee

Department of Computer Science
Virginia Tech
Blacksburg Virginia 24061-0106
janlee@cs.vt.edu
<http://ei.cs.vt.edu/~janlee>



When John Impagliazzo asked me to contribute to the *SIGCSE Bulletin* by writing an occasional column on computer history - one of the loves of my life - I began to wonder where to start. I am proud of the fact that during 1984-1986 when I served on the joint ACM/IEEE Computer Society committee that established the Computer Science Accreditation Board that I was instrumental in ensuring that every accredited degree program pays attention to the study of the social impact of the computer on society and includes a scrutiny of computer ethics. At that time the need for concern was predicated on the recent rash of computer break-ins, the emergence of viruses, and the bad press that our young users were getting.

In 1997 we celebrate the 50th Anniversary of ACM, following only a year behind the celebration in 1996 of the 50th Anniversary of the unveiling of the ENIAC at the University of Pennsylvania, an event that most of us recognize as the true dawn of the computer era. Thus it is appropriate, in this beginning of the second half-century of computing, that we think about how the history of computing can and should be integrated into the computer science curriculum. A joint task group between the IFIP Technical Committee 3 (Education) and the IFIP Working Group 9.7 (History of Computing) is currently developing guidelines for computer history syllabi and hopefully will be able to present its report to their sponsors within the next year. But will it be useful or effective?

Let me start one step further back than assuming that we all agree that computer history should be part of the computer science curriculum. Can I respond to those who ask the question "why"?

ACM has been extremely supportive of the notion of preserving our history since it was one of the sponsors through with AFIPS of the 1960s Smithsonian History project led by Hank Tropp and Walter Carlson. It took several years and several fumbles before a full exhibit relating to "The Information Age" was opened in Washington, and in the meantime several other activities bolstered the notion that we had an interest in our own history. Through AFIPS and the good offices of the ACM Past President Bernard Galler, the journal *The Annals of the History*

of Computing was founded in 1978, and shortly thereafter the future ACM President Gwen Bell, together with her husband Gordon Bell and support from Ken Olsen (President of DEC) established the Computer Museum firstly in Marlborough, Mass, and later in Boston and Silicon Valley. Other museums throughout

the world took up the challenge to the point where we now have a multiplicity of displays.

But is that merely a sop to our more senior members who want their heritage to be preserved or is there something of educational value for our rising generation of programmers, systems analysts, computer engineers, and scientists? Is there anything to be gained by including the history of computing in our curriculum at the college and university level, or should this be relegated to the high school social studies program? Allow me to resurrect some thoughts that have been expressed elsewhere but not necessarily in this forum.

When we were preparing for the Second History of Programming Languages Conference in 1993, the program committee decided that it would be beneficial to organize a pre-conference forum which would look at the status of the history of computing in general and which would provide some guidance for teachers. Well in advance of the conference historian, Michael Mahoney, prepared a paper on the subject "What Makes History?" which was used for several years as part of the guidelines for authors for the *Annals*. That paper led me to write a paper for the 1994 IFIP World Congress entitled "Those who forget the lessons of history are doomed to repeat it or, Why We Study the History of Computing". Here are some thoughts from that paper.



History is not truly a stranger to the computer science curriculum. For the first time the 1991 curriculum for computer science, developed by the ACM/IEEE Computer Society Joint Task Force, included explicit educational modules related to history in four specific areas:

- Artificial Intelligence,
- Operating Systems,
- Programming Languages, and
- Social, Ethical and Professional Issues

However the curriculum is not accompanied by guidelines and support materials related to the history of these subjects, and few teachers have the background, training or resources to effectively provide this instruction. Common knowledge of the field is primarily populated with anecdotes and myths. One of the difficulties of teaching the subject has been the lack of a textbook on the subject. Michael Williams' book, *A History of Computing Technology*, originally published by Prentice-Hall in 1985, went out of print and only recently has been republished by the IEEE Computer Society Press. That is not to say that there are not other textbooks, but the majority of others are more restrictive in their scope and do not lend themselves easily to classroom use.

The very first conference on the history of computing, held at the Los Alamos National Laboratory in 1977 was keynoted by the master of computer-related adages - Richard Hamming - who entitled his talk "We Would Know What They Thought When They Did It" in which he pleaded for a history of computing that pursued the contextual development of ideas, rather than merely listing names, dates, and places and "firsts". Hamming suggested that historians go beyond the published documentation and to speculate about those elements of the history of projects and events that were still unrecorded. He pointed out that what people actually did and what they thought they were doing did not always find its way into the literature or records, and yet this is perhaps the most useful part of history. But can we pass that understanding on to our students? I always make the point of showing a 12 minute video on the development of the first FORTRAN compiler which was produced by IBM on the occasion of the 25th anniversary of the language and initially shown at the NCC Pioneer

Day in 1982. That video (though mostly head and shoulders I will admit) has the developers of the language giving some of their thoughts on how it happened, and I delight in pausing the show to introduce the students to Harlan Herrick who put "GO TO" into the language. Through this means I hope that the students of Java have an understanding that there was a time when life was not so easy as they have it, even though they believe life is difficult. I guess it is my technological way of telling them about my school days when I had to walk 6 miles to school without shoes! It is also useful to point out that the one woman on the project, Lois Haibt, was responsible for the first syntactic analyzer of arithmetic expressions. I hope my students have an awareness that many of the contributions to the field were made by women in our business including Lois and Grace Murray Hopper.

It used to be the case that Grace would appear in our institution at least once every couple of years, and I believe that it was impossible for a student to go through our curriculum without meeting her. Those who met her never forgot the message she brought and went away impressed by her enthusiasm and liveliness. Her message still needs to be transmitted to our students; it is through history and identifying her place in that story that we can keep reinforcing the message since I do not believe that the world has changed that much that the message is no longer valid.

I have not answered the question "why?" properly, but perhaps over the next few months we can come to an understanding of both the question and potential answers. Let me leave you with these thoughts for the next column:

All teaching is, in effect, a teaching of history; we do not invent concepts on the spot as we teach each day, they come from history, but perhaps not from ancient history.

Can history be effectively used as a tool in teaching "modern" computer science?

References

- Mahoney, Michael S. March 1993. "What Makes History?", The Second History of Programming Languages Conference (HOPL-II), *ACM SIGPLAN Notices*, Vol. 28, No. 3, pp. x-xii.
- Williams, Michael R. 1985. *A History of Computing Technology*, First edition, Prentice-Hall, Englewood Cliffs, N.J., 431 pp., Second Edition, IEEE Computer Society Press, Los Alamitos, CA, 1997, 426 pp.