JOURNAL OF OBJECT TECHNOLOGY

Online at http://www.jot.fm. Published by ETH Zurich, Chair of Software Engineering ©**JOT, 2006**

Vol. 5, No. 1, January-February 2006

On Addressing the Computer Science Crisis

Won Kim

During the past several years, there has been a rather precipitous decline in both the number and quality (relative to students choosing other majors) of students choosing to major in computer science (or computer engineering) in many countries. As a result, computer science faces a crisis today. I will review many of the reasons for this situation, and explore some measures to re-vitalize the field.

1 REASONS FOR THE CRISIS

During the past several years, in countries such as the United States and many Asian countries, the number of students majoring in computer science (or computer engineering) in colleges has fallen significantly as compared with the late 1990s. India and China are notable exceptions to this trend. Moreover, in many countries, the quality of the students who choose engineering in general, and computer science in particular, has significantly fallen, as compared with students who choose to major in other disciplines, such as management, medical, law, Oriental medicine (in the case of Korea), natural sciences, etc. The shortage of undergraduate computer science students naturally has led to a comparable shortage of graduate students, which in turn has meant trouble for computer science academic research and innovation.

If the need for a workforce trained in computer science and the need for innovation in computer science have proportionately decreased, and it is unlikely that such needs will increase in the foreseeable future, the current situation of course does not merit any concern or consideration. However, the IT (information technology) sector of the world's economy badly needs a workforce trained in computer science, both at the undergraduate and graduate levels. This is the case even for India and China. I have talked with university professors in various countries, and senior executives of IT corporations in a few countries to try to understand the reasons for the current crisis in computer science. I summarize them below. Some of the reasons are what they have offered me, and others are my own observations.

One major reason is the bursting of the Internet bubble in the early 2000 in many countries. The mass layoffs from the IT sector of the economy scared off many students potentially

Cite this column as follows: Won Kim: "On Addressing the Computer Science Crisis", in *Journal of Object Technology*, vol. 5, no. 1, January - February 2005, pp. 45-49 http://www.jot.fm/issues/issue_2006_01/column05

interested in majoring in computer science. Although the IT sector of the economy has been on a steady and robust rebound, the effects of the contraction of the IT sector appear to linger.

A second reason is the fairly massive outsourcing of the private sector software development to India and China. This is a significant reason for the computer science crisis in the United States. Of course, this does not apply to India and China. It also is not a key reason for countries such as Korea and Japan. The level of cost savings resulting from outsourcing software development, although still significant, has been much less than originally and naively thought possible. As the cost of labor increases in India and China, and as difficulties that cut into cost savings become clearer, outsourcing is likely to become more selective and more software development will be done in the United States.

A third reason is the perception of a less-than-attractive career prospect for computer science majors. This is perhaps the most serious reason in Korea and Japan, where students majoring in computer science fear that after graduation they have to "program 16 hours a day until they are burned out and have to retire at the age of 35", while their non-computer-science major friends get to work until 55 to 65. On top of that, they feel, rightly or wrongly, that annual and career total earnings for computer science majors are much lower than those for their friends in management, stock brokerage, Oriental medicine, etc.

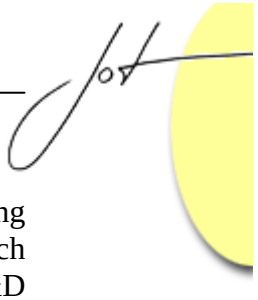
A fourth reason is the failure of many of the computer science departments to adapt computer science education to the changed circumstances – the needs of the IT industry, the mindset of today's youths with respect to careers and jobs, the compensation structures in different areas of human endeavors, etc. Computer science curriculum still includes courses that may best be dropped; offers courses whose contents are needlessly challenging, and therefore discourage potentially worthy students; and does not well prepare students to enter certain IT industries that are under extreme competitive pressures.

2 ADDRESSING THE CRISIS

The best and fundamental solution to the current computer science crisis is simply a sustained and robust IT economy. If several major industry segments, along with the public sector (government-related) can sustain a significantly higher level of employment for skilled software workers, the current crisis will subside to a good extent. However, with or without help from the economy, I believe the computer science crisis has revealed some fundamental problems for both the colleges and the IT industry, and computer science departments must take the initiative to address them at this time. Specifically, computer science departments should take two types of strategic action.

First is the restructuring of the curriculum. The restructuring should include three key elements.

1. "Industry tracks" should be created to support the needs of key industries that are able to hire a continuous stream of a large number of graduates. For example, 13 universities in Korea, working with Samsung Electronics, are about to launch a 2-year "Samsung Electronics embedded systems track" in their computer science curriculum, with financial support from Samsung Electronics. One Korean university now offers a "Samsung Electronics semiconductor track" with financial support from Samsung Electronics. It is understood that students who complete



these tracks will be given special consideration for employment with Samsung Electronics. Although Samsung Electronics has initiated the creation of such "industry tracks" in order for them be able to assign the graduates on R&D projects shortly after hiring them, without costly and lengthy on-the-job retraining, I believe these Samsung Electronics initiatives can serve as useful models for universities elsewhere in adding industry tracks to computer science curricula. Computer science departments in each country should work with major employers to identify such "industry tracks", and seek sponsorship arrangements with them. Candidate industry tracks include the Web industry, the entertainment industry, the space and defense industry, etc. I believe that an industry track should include, besides all essential technical courses, introductory courses on management, leadership, and technical communications. After all, when students graduate, they must survive in the real world, and they really should enter the real world with a modicum of understanding about how it works, and how they should live in it.

2. All courses should be structured into three categories: industry track, required core, and elective. The required core courses are those that are fundamental to computer science, and prerequisites for the industry track courses. As technology has advanced, it has become difficult to justify keeping certain courses or elements of courses that were deemed fundamental 20 years ago. Courses that are no longer fundamental should be either dropped or moved to the "elective" category. Courses such as compilers, automata theory, and artificial intelligence come to mind. In programming language courses, such languages as FORTRAN, COBOL, LISP and PASCAL should really be dropped or moved to elective courses.
3. The emphasis on programming should be shifted to concepts and applications during the first two years, and to software development, as part of the industry tracks during the final two years. For example, in most computer science departments in Korea, freshmen are required to take the C programming language. Although the cell phone, the Internet, and electronic games are integral parts of Korean youths, the C programming language is a harrowing experience to a lot of them. Although some US computer science departments still have the luxury of being able to inject a "drop out" course in their curriculum to weed out less talented or less determined students, many other countries have to work with pretty thin pools of students interested in majoring in computer science. In such computer science departments, C should be taught either in its minimal form first or such a simple language as BASIC should be taught first to give students a feel for what programming is and to whet their appetite for a more elaborate language. During the "tender" first two years, extra efforts must be made on the part of the computer science faculty to introduce basic computer science concepts and tools in ways that are much more comfortable and tangible to students, for example, by using computer games, hand-held wireless devices, robots, etc. During the final two years, training on programming should be intensified both as an important part of software development and as a key tool for delving into details behind

> many of the concepts in industry track courses. Computer science majors should learn software development by heart, not just programming. Today, most students merely learn to program, that is, figuring out an approach to solve a very small problem, coding (alone), and making sure the code generates the desired result. Software development in the real world often involves a large set of detailed requirements, an design and coding by a number of developers, often interfacing with third-party software, and a rigorous sequence of testing aided by debugging and testing tools, and documentation of the design and source code. Colleges have done a very poor job of teaching students software development.

The second type of strategic action computer science departments must take is to cause changes in the IT industry that will result in fundamental changes in how computer science careers are perceived by potential students, parents, and secondary-school teachers. Computer science departments and the IT industry have common interests and must work together on this very difficult action item. Governments should also be involved to help where it can. The changes in the IT industry include more generous profit sharing with employees; and in certain Asian countries, the lengthening of technical careers, from the current unofficial 45-55 to 55-60. In many Asian countries, the salaries for new employees are the same whether they are computer science majors or electronics engineering majors, even if they both will be assigned to the same software development project. Where there is a much greater demand for computer science majors than electronics engineering majors, but there is much smaller supply of computer science majors, as is the case in Korea today, computer science graduates should be paid higher salaries. If corporations can be persuaded to make such adjustments in compensation based on the law of supply and demand, it will help buoy the high-demand field.

Further, colleges should try to bring their students and professors closer to the industry. Colleges should run co-op programs, if they do not already do. Further, colleges should appeal to the IT industry to create student internship programs, and encourage students to join such programs for one or two summers. Samsung Electronics and LG Electronics in Korea already run substantial student internship programs. Colleges should encourage professors to take sabbatical leaves in corporations rather than other universities, so that the professors will become more intimately familiar with some of the current technologies in the real-world and will teach and research more effectively after the sabbatical. In certain countries, professors do not have adequate number of teaching assistants. Colleges, the government, and perhaps even the IT industry should take care of this problem, so that better homework exercises can be developed and graded.

As an integral part of this strategic action, computer science departments should create and run a “professional” marketing program to better inform potential students, parents, and secondary-school teachers about college computer science education and computer science careers. To be sure, many computer science departments hold occasional informational events, invite their alumni or IT industry executives to talk to students, etc. Such efforts are of course important and must continue. However, rather than solely relaying on amateurish “marketing by engineers”, computer science

departments should work with the IT industry and the government to create "professional" marketing program. Basically, the marketing program should be aimed at projecting the numerous types of jobs for a workforce trained in computer science and the numerous types of exciting or surprising applications that require substantial software, thereby dispelling the unfortunate impression prevailing in some countries that a computer science career simply means never-ending programming in the "scary" C language.

About the Author

Won Kim is Senior Advisor at Samsung Electronics, Korea. He is Editor-in-Chief of ACM Transactions on Internet Technology (www.acm.org/toit). He is Global General Chair of the Human.Society@Internet International Conference. He is the recipient of the ACM 2001 Distinguished Services Award, and is an ACM Fellow.