Update on Professional Development

For many years, the only way to attain an education in software engineering has been through the school of hard knocks. A common lament among experienced software developers is that colleges don’t teach students the skills they need to perform effectively on the job. What is needed to give software developers a better start?

Formal professional education has traditionally been divided into three parts. Initial education consists of attaining an undergraduate degree in a particular field from an accredited university. Skills development gives nascent professionals real-world experience in their chosen occupation. It can consist of an internship before graduation or an apprenticeship afterwards. Then, after working in the field for awhile, ongoing professional development helps professionals keep their skills up to date.

Software engineering is seeing encouraging signs in each of these three areas.

University Programs

Graduate-level programs in software engineering have existed for 20 years or more. Seattle University awarded the world’s first software engineering master’s degree in 1982. At present, about 25 master’s programs in software engineering are offered in the US; Canada, the UK, Australia, and other countries offer a handful more.

However, undergraduate software engineering programs are still in their infancy, especially in North America. The Department of Computer Science at the University of Sheffield in the UK introduced the first such program in 1988. The Rochester Institute of Technology initiated one in the US, admitting freshmen in 1996. At least 13 UK and six Australian universities offer undergraduate programs. As of fall 1999, in the US, new bachelor’s programs will begin at Auburn University, the Milwaukee School of Engineering, Monmouth University, and Montana Tech. In Canada, bachelor’s programs are offered by Concordia University, McMaster University, Memorial University of Newfoundland, and the University of Ottawa. Several other North American universities are actively considering adding programs.

One encouraging element of these programs is the skills-development component. RIT’s program, for example, requires five quarters of work experience in addition to the academic studies. I hope that many more undergraduate programs emerge, but compared to just five years ago, numerous opportunities are now available for the undergraduate who wants to study software engineering.

SWEBOK

Supporting all levels of professional development is the joint IEEE Computer Society/ACM initiative to define the software engineering body of knowledge (SWEBOK). This initiative, spearheaded by researchers at the University of Quebec at Montreal, focuses on identifying the generally accepted elements of software engineering. The effort involves both academic and industrial participants and is currently looking for reviewers. If you would
like to help, please see the call for reviewers on the SWEBOK Web site at http://www.swebok.org.

**Professional Development**

“These developments in undergraduate education sound great,” you say. “But what about those of us who have already been in the field for 10 years? Do we have to go back to school?” The answer to this question is, “Yes and no.” A lot of knowledge has been developed in the software engineering field in the past 10 to 20 years, and professionals who haven’t kept up with these developments probably have become out of date. If you’re out of date, you do need to study to catch up. But you don’t have to go to school. One of the hallmarks of any field in which knowledge changes rapidly is that professionals are expected to engage in an ongoing program to keep their skills current. In well-developed fields such as medicine, law, and accounting, professionals must attain some number of continuing-education credits to renew their license or certification.

Software engineering doesn’t yet have any widespread licensing or certification, so the requirements for continuing education are left to individual initiative. Many software developers work by themselves or in small groups and don’t have any idea where to start on their continuing education. In support of that need, my company has made its professional-development ladder publicly available at http://www.construx.com/ladder. The ladder contains the following levels:

- Levels 8–10. Acquire proficiency in fundamental skills of excellent software development. These levels are appropriate for students just out of college and other workers who haven’t engaged
in any systematic study of software engineering.

- Level 11. Acquire professional-level skills in software engineering. This level is designed to train a person to become a professional software engineer.

- Level 12. Perform as a professional software engineer. This is the plateau level. At my company, most software engineers are expected to attain this level five to 10 years into their career and then maintain the education and training necessary to perform at this level.

For each level, we have mapped out a program of self-study that is loosely based on the SWEBOK knowledge areas. Emphasis in Levels 8 through 10 is on attaining “introductory” or “competency” knowledge. The emphasis at Level 11 is on attaining “competence” in all knowledge areas and “mastery” in a few. The Web site contains detailed listings of the books and articles a concerned software professional should know in order to perform effectively in the field.

IEEE Software’s Role

IEEE Software’s mission is to “build the community of leading software practitioners.” As part of that mission, our articles and columns should be both thought-provoking and educational. How are we doing? Are we helping you keep your education and training current? Are we hitting the mark? Please let us know by sending e-mail to me at software@construx.com or to the magazine staff at software@computer.org.

By the way, the next issue of IEEE Software will focus on professional issues. Look for it in November.