Welcome to the first column of SEEd, whose purpose is to provide current and useful information and opinions to SEN readers regarding education in the discipline of software engineering. Occasionally, I plan to entice other software engineers (educators, practitioners, researchers, etc.) to write a column to introduce you to other views and opinions (which I may not condone nor have to defend). If you have read this far, congratulations and thanks – please keep reading. By the way, I am neither British nor red headed – the ‘other’ Peter Henderson.

So let’s get started:

- **Fact 1** – “The first software engineering programs were graduate masters level.”
- **Fact 2** – “There are currently about 20 undergraduate software engineering programs in the U.S. and 3 to 4 times that many internationally.”
- **Fact 3** – “There are only a handful of Ph.D. programs in software engineering world wide.”
- **Claim 1** – “Software engineering is very similar to traditional engineering.”
- **Claim 2** – “Software engineering is very different than traditional engineering.”
- **Prediction** – “Within 50 years, software engineering will supplant computer science as the educational discipline for professional software developers.”
- **Challenge** – “To make software engineering an acknowledged professional ‘engineering’ discipline.”

**What is software engineering?** A hard question to answer, but clearly it is a sub-discipline of computer science since many computer science programs, both undergraduate and graduate, require at least one software engineering course. Wait, Henderson, this reasoning is not logical and I don’t even agree with you. Good!! Think about the relationship between science and engineering, say chemistry and chemical engineering. To paraphrase others, engineering applies the knowledge of science to develop ‘things’ useful to mankind. This is a potential model for understanding the relationship between computer science and software engineering. A peek at the undergraduate software engineering curricula (CCSE) page [cse] provides additional insight.

**What is CCSE?** Computing Curricula Software Engineering. It is part of a joint IEEE Computer Society and ACM effort to develop undergraduate curricula for four key computing disciplines: Computer Science, Computer Engineering, Information Systems, and Software Engineering (CCSE) - see [1], the Computing Curricula home page for links to each. It is important to recognize that these four curricula attempt to capture the current state of ‘computing’ as a discipline. It is anticipated that as the discipline of computing expands and becomes better defined that corresponding new curricula will be created and existing curricula will be updated.

Computer science is one sub-discipline of computing, and the Computer Science Volume (CC 2001) curricula recommendations was the first to be published in December 2001. The first draft of SEEK, Software Engineering Education Knowledge, can be viewed at [6]. It specifies the proposed core knowledge for an undergraduate software engineering curriculum.

**What is the SE core knowledge in SEEK?** It includes: Fundamentals (or foundations) [260], Professional Practice [35], Software Requirement [43], Software Design [78], Software Construction [46], Software Verification and Validation [46], Software Evolution [9], Software Process [16], Software Quality [17], and Software Management [20]. Many of these areas overlap, so the proposed hours can be misleading. The largest, Fundamentals (foundations) is comprised of mathematical foundations [60], computing foundations [140], engineering foundations [25] and modeling [25]. Bored yet?

**What are the computing foundations?** It is the core computer science knowledge upon which software engineering builds (see page 9 of [6]). Now we have come full circle with engineering using/applying the knowledge of science (just as it should be).

Selected and public evaluations of SEEK were done in the summer and fall of 2002 respectively. The final SEEK chapter of the Computing Curricula Software Engineering Volume should be completed during the spring of 2003. The pedagogy focus groups are using SEEK as a guide for proposing model curricula, courses, etc. As with SEEK, this effort will build upon CC 2001 to ensure some level of consistency (e.g., topics, courses, etc.) with the latter.

When complete, CCSE will be a guide for the development, evolution, assessment and accreditation of undergraduate software engineering programs - each potential topics for future SEEEd columns. However, not everyone is patiently waiting for the CCSE volume to begin creating undergraduate software engineering programs. Some existing programs are based upon traditional engineering programs. These require basic engineering courses and/or extensive continuous mathematics. Others are building on foundations of discrete mathematics, problem solving and/or computer science. Lessons learned from existing and evolving software engineering

---

1 Not necessarily mutually exclusive groups.

2 The second draft should be available by the time this column is published – please check [2]

3 The numbers in […] correspond roughly to recommended classroom contact hours for each category.
programs will significantly influence the final draft of the CCSE volume.

Perhaps the phrase ‘final draft’ does not compute in your mind. This signifies that this is an ongoing, long-term effort. The frequency of new ‘final drafts’ is currently not specified; however, it is anticipated that they will be more frequent than the 10+ years of past computer science curriculum publications.

The acronym, SEEd, for this column was recommended by Jorge Díaz-Herrera (Rochester Institute of Technology) to capture the essence of Software Engineering Ed-ucation and to plant an important message in the readers mind as these are exciting and pivotal times for software engineering education worldwide. Accordingly, please help to make my prediction "Within 50 years, software engineering will supplant computer science as the educational discipline for professional software developers" come true, and to rise to the challenge "To make software engineering an acknowledged professional 'engineering' discipline."

If you are interested in further information regarding software engineering education please check out: the Software Engineering Institute (SEI) [7], the Forum for Advancing Software Engineering Education (FASE) [4]. Several conferences devoted to software engineering education include: the International Conference on Software Engineering Education and Training (CSEE&T) [3], and the ICSE Education track [5].

Acknowledgements
I would like to thank Drs. Jorge Díaz-Herrera, Tom Hilburn, and Tim Lethbridge for their comments and suggestions on a preliminary draft of this column which have greatly improved its readability and accuracy.

References

---

4 ‘plant’ a ‘seed’ in case you missed the play on words!
5 This is a ‘prediction’ not a ‘wager’!