PROGRAMMING LANGUAGES

The author of “Computing Trends Lead to New Programming Languages” (S. Ortiz, Technology News, July 2012, pp. 17-20) presents an interesting point of view. Having programmed in approximately 15 languages over more than 25 years, my gut feeling is that while we can understand the emergence of new languages, this might not have any correlation with their eventual survival and adoption in the general programming community.

Actually, a key to understanding this phenomenon might be the fact that there are two distinct and perhaps contradictory perspectives in terms of language adoption: the company’s and the programmer’s. Understandably, in the face of a complex evolving market and economic landscape, any small- to medium-sized software company is likely to have a continuing interest in the diversification of its programming portfolio. However, from the programmer’s perspective, the effort required in terms of the heavy time commitment and learning curve might not always warrant making the switch to a new language.

In terms of language adoption, this might lead to the proverbial “chicken and egg” problem: developers create new programming languages with the hope that they will become a mainstream option, but doing so would entail attracting a large following of programmers, something that might never occur. Thus, while many new languages become available, language adoption actually resembles a scale-free network, resulting in a power-law distribution of connections between languages and programmers. In other words, while a few languages might be adopted by most programmers, a larger number of languages attract only a few followers.

This phenomenon would explain the outcome noted at the start of this article: in general, even though numerous languages ranging from the old ones to those in their relative infancy are available, the programming community tends to have a significantly unbalanced focus on a few key languages.

Muaz A. Niazi
muaz.niazi@bahria.edu.pk

THE FOUNDATIONS OF HIGHER EDUCATION

In a letter referring to the function of colleges and universities (Letters, July 2012, pp. 6-7), Victor Skowronski stated that “universities were originally founded to train clergy,” and only dealt with the intellectual aspects of learning.

But this claim is not accurate in two respects. Universities did not have a medieval origin—they are much older. For example, in ancient times, Plato’s Academy focused on higher learning and research and also provided athletic training. In addition, Gregory Thaumaturgu, a third century Christian bishop, described the work of the celebrated scholar Origen, head of the Catechetical School of Alexandria.

And even medieval clergy schools offered something more that teaching how to deal with heretics’ ideas. The battle with heretics focused not only on the theoretical plane but also on the style of living. Thus, most of the aspects of developing the whole person that the Forward Slash column mentioned (May 2012, p. 116) were considered essential, including the spiritual, moral, social, and intellectual aspects of character.

Piotr Karocki
pkar@ieee.org

Victor Skowronski responds:

The situation when universities came into existence in the Middle Ages was more complicated than what Piotr Karocki describes. The schools of Greek philosophy had died out at about the same time that Rome fell. They only survived via their writings stored away in monastic libraries.

By the way, during the Middle Ages, the institution that dealt with the spiritual, moral, social, and intellectual aspects of individuals was the medieval monastery. Monasteries existed before universities, and the rule of St. Benedict specifically addressed the social, moral, spiritual, and intellectual lives of the monks.

Even so, the record with respect to monasteries is ambiguous. Toward the end of the Middle Ages, they became a source of literate clerks for the expanding monarchies. At the same time, however, new orders of monks were established. One reason for this was that existing monasteries were losing their spiritual focus, presumably because they were spending too much time training individuals to be clerks.

Victor Skowronski
victor31@ieee.org

RAISING CAPITAL

As president of a software start-up (now 13 years old), I’d like to offer a few comments about “Raising Capital: Where to Find It, How to Secure It, and Tips on What to Avoid” (B.M. Gaff, R.N. Kimball, and J.M. Hanson, Computing and the Law, July 2012, pp. 11-13).

We welcome your letters. Send them to letters@computer.org. Letters are subject to editing for style, clarity, and length.
The authors respond:
Mr. Freedman makes a good point. In some situations, a start-up’s first customers might be willing and able to provide “up front” payments for products or services to be delivered later. Ideally, this funding should be nondilutive. However, it’s important to keep in mind that customer funding is usually available only in certain limited situations where the risk of nonperformance from the customer’s perspective is relatively low, the time for performance is relatively short, or the customer has limited alternative options for the products or services.

If there are unexpected problems with the products or services, or delays in delivering them, there’s a risk of alienating those first customers and getting some unfortunate attention that could make future sales more difficult. Indeed, many customers are concerned with doing business with smaller companies due to their concerns about the company’s ability to stand behind its products or services, particularly if the company is thinly capitalized.

Brian M. Gaff, Richard N. Kimball, and Jill M. Hanson

ERRATUM
In “Simware: A Holistic Warehouse-Scale Computer Simulator” (S. Yeo and H.-H.S. Lee, Sept. 2012, pp. 48-55), there is an error in the labeling of the y-axis in Figure 5a. The correctly labeled figure is provided here.

Computer regrets this error.

Figure 5. SimWare simulation of a Standard Workflow Format (SWF) workload from SHARCNET: (a) power consumption, utilization, and latency; (b) effect of air travel time; and (c) energy usage breakdown, power usage effectiveness (PUE), and total PUE (tPUE).