Robert A. Hovis (r-hovis@onu.edu)
Department of Computer Science
Ohio Northern University
Ada, OH 45810

Position Statement
FYI, Duke University
July 15-16, 2000

Although I am a mathematician by training, I have been involved with computer science education for more than twenty years. I have two observations to make based on my experience with CS1-CS2 during that time frame.

1. The programming environment has become more complex (and less comprehensible). In the early days, we followed a simple edit, compile, link, and execute sequence that was essentially the same whether the language was FORTRAN, PL/1, Pascal, COBOL, or C. I am talking about interactive, time-sharing environments, not batch processing. Although the graphical user interface of today may be more appealing, it also may be more confusing and complex. I give one example: During the course of these two decades, I have taught an Intro COBOL course maybe five or six times. It seems that every time I taught it, we used a different compiler under a different operating system. This past fall, for the first time, we used a graphical IDE for COBOL. Because of the problems in dealing with the environment at the front of the course, we actually covered less COBOL programming than at any time in the past.

2. In CS1-CS2, as we have progressed from FORTRAN to PL/1 (SP/k) to Pascal to C++ (to Java), we need to recognize that we have pushed more and more abstraction into freshman level computer science. The FORTRAN FORMAT statement may have been ugly, messy, and confusing, but it was not abstract. Such things as classes and operator overloading in C++ are very abstract indeed. Mathematics departments are constantly being berated for making first year calculus too abstract. I believe that we are now at the point where the abstraction present in CS1-CS2 exceeds anything done by the mathematics faculty in calculus.

I am not suggesting a return to earlier programming paradigms and I am not suggesting that we abandon the current integrated development environments. I am only suggesting that we need to remind ourselves that we are subjecting our students to a level of complexity and abstraction that goes far beyond what a previous generation of students had to contend with.

Sunday, July 9, 2000