

CS-480—Senior Seminar
Study questions for *Software Certification Debate*

Fall '02

James Rogers

x1671

jrogers@cs.earlham.edu

www.cs.earlham.edu/~jrogers

1. What is Software Engineering? How is it different from Computer Programming? How is it different from Civil Engineering?
2. What does this Software Certification initiative seek to certify?
3. What happens if a Certified Software Development Professional produces bad software?
4. What are the constituencies that Tripp believes will benefit from Software Certification? How does he expect them to benefit?
5. According to Tripp, “The CSDP . . . encourages individuals to draw from a broad base of software knowledge.” Kolawa, in contrast, does not believe the knowledge base of the software professional is the problem. What does he believe the problem to be? Are these two points of view necessarily incompatible?
6. Kolawa claims that CDSP will stifle innovation. How?
7. Kolawa characterizes the CDSP as “an unnecessary proof that developers have not forgotten the fundamentals they have already learned in an accredited computer science program.” From this perspective, if the knowledge base of software professionals is deficient, who is to blame? But he also claims that, in his experience, the “most brilliant” programmers usually “did not develop their talent by completing whatever set of courses a university committee deemed important.” In that case, if the knowledge base of software professionals is deficient, who is to blame? How is one to tell?
8. What, in Kolawa’s view, distinguishes computer science from “true sciences”? Does he have it right? As Kolawa notes, Math and Physics do not, as disciplines, have certification programs. Where do certification programs exist?

9. Given the short bio at the end of each article (and the add from the back of another issue of *Computer*) how do you think their professional roles have influenced Tripp and Kolawa's perspectives on this issue?