

## Panel proposal for CCSCNE 2002, Worcester State College, 19-20 April 2002

### Outcomes and Assessment of the Introductory Sequence of Computer Science Courses

Panelists: Richard T. Close, U. S. Coast Guard Academy  
Danny Kopec, Brooklyn College  
XXX  
XXX

#### Description of Topic:

Curriculum 2001 provides extensive recommendations for the content and delivery of the introductory sequence: the first two (or three) computer science courses. Building on these guidelines, a delivery paradigm can be selected and specific courses developed. Part of this implementation phase involves the construction of measurable outcomes. This panel will examine this effort and report on efforts that have been initiated at several institutions with particular emphasis on the assessment techniques that are in place or under development. Although it is too early for specific assessment procedures and instruments to have been developed and validated, considerable progress has been made. A call for wider involvement in the process also seems to be indicated.

#### Position statements:

Richard T. Close  
Professor, Department of Science, U. S. Coast Guard Academy

Dr. Close is currently a faculty member at the U. S. Coast Guard Academy. He was formerly the Head, Department of Computer Science

and also has served as Associate Dean of Academics at the Academy. He is a member of ACM, IEEE-CS, a former chair of SIGCUE and one of the founding Board members of CCSCNE.

Accrediting agencies are insisting that institutions articulate the outcomes of instruction and demonstrate that these outcomes are being achieved. At the institution, department and course level, considerable effort is being expended to develop specific outcomes and methods for their assessment. For the introductory sequence of computer science courses, Curriculum 2001 provides a basis for this work. However, it is still substantial task to transform these recommendations into measurable outcomes. One method of assessing outcomes, undoubtedly part of a larger effort, is the administration of suitable pre and post tests. The process of development of valid assessment instruments is not a trivial task but several notable models from other disciplines, e.g. physics, are available. The goal here is to produce an inventory of specific questions that can be used to demonstrate that particular content items have been grasped. Although it is too early to provide a finished, quality instrument, some progress has been made and the process for further development seems clear.

Dr. Danny Kopec  
Department of Computer and Information Science  
Brooklyn College, Brooklyn, New York

Dr. Kopec is Associate Professor of Computer and Information Science and for many years has been interested in problem solving, knowledge representation, and the study of errors made in various knowledge content areas necessary for correct problem solution. Testing and evaluation of knowledge is his specialty.

He is Graduate Deputy Chair of a rapidly growing Master's Degree programs at Brooklyn College (over 400 students) and an International

## Chess Master.

Curriculum 2001 presents a series of *knowledge areas* which students should emerge with from the introductory course sequence. The challenge to computer science educators is the demonstration that the essential knowledge in critical computer science content areas (e.g. loops, conditionals, functions, and modular design) has been mastered.

I have had some experience teaching in the British higher educational system, where continuous assessment, as we do it here is not the standard. Examinations are more subjective rather than objective, although there is still concern with *learning outcomes*. One remedy is for each course instructor to identify desired learning outcomes for the courses he/she teaches and then try to evaluate at the end of the course, based on the course measurables (e.g. exams, quizzes, programs, papers, presentations, etc.), whether the course learning outcomes have been met.

I am involved with several efforts to help quantify what it is that students must know to be able to demonstrate mastery of the introductory course material, as well as to understand what kinds of errors non-beginner programmers will typically make. These will be described briefly.