

Second Draft
3/79

MSE79: SECOND DRAFT OF A
MASTERS CURRICULUM IN SOFTWARE ENGINEERING

Richard E. Fairley*
Computer Science Department
Colorado State University
Fort Collins, CO 80523
(303) 491-7026

Introduction

In the January, 1979 issue of Software Engineering Notes, we presented the first draft of a proposed Masters curriculum in software engineering. The draft and the detailed course outlines were mailed to fifty-five well known individuals in software engineering. Twenty-six responded with detailed criticisms and suggested revisions; their reviews form the basis for this second draft of the proposed curriculum.

The reviewers were nearly unanimous on several issues, and somewhat divided on others. A summary of the major comments and criticisms is presented here, and the next section of the paper describes the revisions made in response to those comments.

The reviewers were in close agreement on the following issues:

1. The proposed curriculum is timely and appropriate. All reviewers agreed that software engineering should be taught at the Masters level. Only one reviewer was opposed to development of the curriculum. His reasons are summarized below.
2. The first draft of the curriculum included too many "soft" topics, and not enough technical software engineering and computer science courses.
3. Some of the course outlines in the first draft were too weak to support a full semester of study, and others did not allow enough time to cover the topics in depth.
4. The first draft specified too many undergraduate prerequisite courses, particularly in the hardware area.
5. The first draft did not clarify how the courses fit together to satisfy the purposes and objectives of the curriculum.
6. The first draft did not discuss implementation issues in enough detail.

There were scattered comments on some issues and disagreement on others:

1. One reviewer felt it too ambitious to teach methodology, project management, and applications in one program.
2. Some reviewers felt there should be an abbreviated program for schools with limited resources.

*Chairman
IEEE Computer Society
Subcommittee on Software Engineering Education

Second Draft

3/79

3. One reviewer felt the subject matter in software engineering to be too volatile to justify preparation of detailed course outlines.
4. The first draft included required courses in organizational behavior, security and privacy, human-machine interaction, database systems, and distributed systems. There was considerable disagreement as to the appropriateness of requiring these courses in the curriculum. Some reviewers felt them to be essential, and others had suggestions for alternative topics felt to be of equal or greater importance.
5. One reviewer was opposed to development of a Masters program in software engineering for the following reasons:
 - "Basically I feel that the idea of a separate program in such a narrowly defined field is undesirable since it implies that software engineering can stand alone from other areas of computer science and engineering."
 - "Better management techniques will not be the major factor which reduces cost.....More paperwork is not needed."
 - "The proposed curriculum lacks any depth in design methods and advanced analytical tools to study the properties of software systems."
 - "The discussion of how this program could be implemented particularly disturbs me.....The only place that software engineering can be taught is in a school of engineering."

The first draft of the proposed curriculum was revised in response to the criticisms received from the reviewers. In addition to the major criticisms mentioned here, many reviewers provided course outlines and proposed detailed modifications. The subcommittee is grateful for the thoughtful, in-depth comments received from the reviewers. The present draft is based on their suggestions.

Revisions

An overview of the second draft of MSE79 is presented in Figure 1. The following revisions were made to produce the present version:

1. The prerequisites stated in the first draft were split into core prerequisites (Table I) and recommended prerequisites (Table II).
2. The nature of the introductory course (MSE-1) was changed to emphasize problems areas and current issues in software engineering.
3. The original three courses in software methodology (old MSE-2, MSE-5, and MSE-9) were expanded to four courses (new MSE-4, MSE-5, MSE-6, and MSE-7). This permits deeper coverage of some topics, such as system analysis, design methods, and performance evaluation, and allows addition of other topics such as analytical tools, formal verification, and concurrency.

Second Draft

3/79

4. First draft courses MSE-6, Security and Privacy; MSE-10 Human Factors; MSE-11, Database Systems; and MSE-12, Distributed Systems became recommended computer science electives CSE-1, CSE-2, CSE-3, and CSE-4. These courses are now computer science electives to permit greater flexibility.
5. First draft courses MSE-4, Organizational Behavior; and MSE-7, Software Project Management were merged into new course MSE-9, Software Management. A management elective was created.
6. The Technical Communication course (old MSE-3) was strengthened with more material relevant to software engineering and renumbered (new MSE-8).
7. The software laboratories were renumbered MSE-2 and MSE-3.
8. A great deal of narrative has been added to the report to explain the purposes and objectives of the curriculum, to discuss prerequisites and preparation, to provide an overview of the curriculum, to provide a profile of a typical graduate, and to discuss implementation considerations. Under implementation considerations, a prioritized list of courses has been provided as an indication of the order in which new courses should be developed, and to indicate which courses should be offered if a full curriculum cannot be implemented.

Summary

The second draft of the proposed curriculum has been mailed to the reviewers for comment. A final draft will be prepared and forwarded to the Computer Society for publication in late spring. In the meantime, we welcome your comments and criticisms.

Table I. The Undergraduate Prerequisite Core

<u>CORE COURSE</u>	<u>COMPUTER SOCIETY REFERENCE COURSE(S)</u>	<u>ACM 79 REFERENCE COURSE(S)</u>
Introduction to Computer Programming	SE-1	CS 1; CS 2
Computer Organization	CO-1; CO-2; CO-3	CS 3; CS 4
Data Structures and Design of Algorithms	SE-2; SE-3	CS 7
Operating Systems and Computer Architecture	SE-6; SE-7	CS 6
Language Implementation	SE-8	CS 15
Discrete Structures	TC-1	MA 4

Second Draft

3/79

Table II. Recommended Undergraduate Courses

<u>RECOMMENDED COURSE</u>	<u>COMPUTER SOCIETY REFERENCE COURSE(S)</u>	<u>ACM 79 REFERENCE COURSE(S)</u>
Digital Logic	DL-1; DL-2	CS 4
Digital Lab	L-1	
Microprocessors	DL-3	
Microprocessor Lab	L-3	
Database Systems	SE-4	CS 11
Survey of Language Concepts	SE-5	CS 8
Design and Analysis of Algorithms	TC-2	CS 13
Software Development Lab		

	Introduction/ Laboratory	Software Methodology	Computer Science	Management/ Communication
FIRST SEMESTER	MSE-1 Introduction to Software Engineering	MSE-4 Software Methodology I	Computer Science Elective	MSE-8 Technical Communication
SECOND SEMESTER		MSE-5 Software Methodology II	Computer Science Elective	MSE-9 Software Management
THIRD SEMESTER	MSE-2 Software Laboratory I	MSE-6 Software Methodology III		Management Elective
FOURTH SEMESTER	MSE-3 Software Laboratory II	MSE-7 Software Methodology IV	Computer Science Elective	

Figure 1. The Masters Program in Software Engineering