

SOFTWARE ENGINEERING (S E)

Courses primarily for undergraduates:

S E 101: Software Engineering Orientation

Cr. R.

Introduction to the procedures, policies, and resources of Iowa State University and the department of Computer Science and Electrical and Computer Engineering. Information on engineering and computer-based professions.

S E 166: Careers in Software Engineering

Cr. R.

Overview of the nature and scope of the software engineering profession. Relationship of coursework to careers. Departmental rules, student services operations, degree requirements, program of study planning, career options, and student organizations.

S E 185: Problem Solving in Software Engineering

(2-2) Cr. 3.

Prereq: MATH 143 or satisfactory scores on mathematics placement examinations; credit or enrollment in MATH 165

Introduction to software engineering and computer programming. Systematic thinking process for problem solving in the context of software engineering. Group problem solving. Solving software engineering problems and presenting solutions through computer programs, written documents and oral presentations. Introduction to principles of programming, software design, and extensive practice in design, writing, running, debugging, and reasoning about programs.

S E 319: Software Construction and User Interfaces

(Cross-listed with COM S). (3-0) Cr. 3.

Prereq: COM S 228

Basic theory of grammars, parsing. Language paradigms. State transition and table-based software design. Review of principles of object orientation, object oriented analysis using UML. Frameworks and APIs. User interface architecture, evaluation of user interface. Design of windows, menus, and commands. Introduction to formal specification and model-based software design. Introduction to domain-specific software engineering.

S E 329: Software Project Management

(Cross-listed with CPR E). (3-0) Cr. 3.

Prereq: COM S 309

Process-based software development. Capability Maturity Model (CMM). Project planning, cost estimation, and scheduling. Project management tools. Factors influencing productivity and success. Productivity metrics. Analysis of options and risks. Version control and configuration management. Inspections and reviews. Managing the testing process. Software quality metrics. Modern software engineering techniques and practices.

S E 339: Software Architecture and Design

(Cross-listed with CPR E). (3-0) Cr. 3.

Prereq: S E 319

Modeling and design of software at the architectural level. Architectural styles. Basics of model-driven architecture. Object-oriented design and analysis. Iterative development and unified process. Design patterns. Design by contract. Component based design. Product families. Measurement theory and appropriate use of metrics in design. Designing for qualities such as performance, safety, security, reliability, reusability, etc. Analysis and evaluation of software architectures. Introduction to architecture definition languages. Basics of software evolution, reengineering, and reverse engineering. Case studies. Introduction to distributed system software.

S E 342: Principles of Programming Languages

(Cross-listed with COM S). (3-1) Cr. 3.

Prereq: Minimum of C- in COM S 228 and MATH 165; COM S 230 or CPR E 310

Study of concepts in programming languages and major programming paradigms, especially functional programming. Special emphasis on design tradeoffs that enable students to make sound choices of programming languages for a given software development task. Programming projects.

S E 396: Summer Internship

Cr. R. Repeatable. SS.

Prereq: Permission of department and Engineering Career Services

Professional work period of at least 10 weeks during the summer. Students must register for this course prior to commencing work. Offered on a satisfactory-fail basis only.

S E 398: Cooperative Education (Co-op)

Cr. R. Repeatable. F.S.

Prereq: Permission of department and Engineering Career Services

Professional work period. One semester per academic or calendar year. Students must register for this course before commencing work. Offered on a satisfactory-fail basis only.

S E 409: Software Requirements Engineering

(3-0) Cr. 3.

Prereq: COM S 309; for graduate credit: graduate standing or permission of instructor

The requirements engineering process including identification of stakeholders requirements elicitation techniques such as interviews and prototyping, analysis fundamentals, requirements specification, and validation. Use of Models: State-oriented, Function-oriented, and Object-oriented. Documentation for Software Requirements. Informal, semi-formal, and formal representations. Structural, informational, and behavioral requirements. Non-functional requirements. Use of requirements repositories to manage and track requirements through the life cycle. Case studies and software projects will be required.

S E 412: Formal Methods in Software Engineering

(Cross-listed with COM S, CPR E). (3-0) Cr. 3.

Prereq: COM S 230 or CPR E 310; COM S 311, STAT 330; for graduate credit: graduate standing or permission of instructor

A study of formal techniques for model-based specification and verification of software systems. Topics include logics, formalisms, graph theory, numerical computations, algorithms and tools for automatic analysis of systems. Graduate credit requires in-depth study of concepts.

S E 416: Software Evolution and Maintenance

(Cross-listed with CPR E). (3-0) Cr. 3.

Prereq: COM S 309

Practical importance of software evolution and maintenance, systematic defect analysis and debugging techniques, tracing and understanding large software, impact analysis, program migration and transformation, refactoring, tools for software evolution and maintenance, experimental studies and quantitative measurements of software evolution. Written reports and oral presentation.

S E 417: Software Testing

(Cross-listed with COM S). (3-0) Cr. 3.

Prereq: COM S 309; COM S 230 or CPR E 310; ENGL 250, SP CM 212

Comprehensive study of software testing, principles, methodologies, management strategies and techniques. Test models, test design techniques (black box and white box testing techniques), test adequacy criteria, integration, regression, system testing methods, and software testing tools.

S E 419: Software Tools for Large Scale Data Analysis

(Cross-listed with CPR E). (3-3) Cr. 4.

Prereq: CPR E 308 or COM S 352, COM S 309

Software tools for managing and manipulating large volumes of data, external memory processing, large scale parallelism, and stream processing, data interchange formats. Weekly programming labs that involve the use of a parallel computing cluster.

S E 421: Software Analysis and Verification for Safety and Security

Cr. 3. F.S.

Prereq: COM S 309; CPR E 310 or Com S 230

Significance of software safety and security; various facets of security in cyber-physical and computer systems; threat modeling for software safety and security; and categorization of software vulnerabilities. Software analysis and verification: mathematical foundations, data structures and algorithms, program comprehension, analysis, and verification tools; automated vs. human-on-the-loop approach to analysis and verification; and practical considerations of efficiency, accuracy, robustness, and scalability of analysis and verification. Cases studies with application and systems software; evolving landscape of software security threats and mitigation techniques. Understanding large software, implementing software analysis and verification algorithms.

S E 490: Independent Study

Cr. arr. Repeatable.

Prereq: Senior classification in software engineering

Investigation of an approved topic.

S E 491: Senior Design Project I and Professionalism

(2-3) Cr. 3.

Prereq: S E 329 and S E 339, CPR E 308 or COM S 352, ENGL 309 or ENGL 314

Preparing for entry to the workplace. Selected professional topics. Use of technical writing skills in developing project plan and design report; project poster. First of two-semester team-oriented, project design and implementation experience.

S E 492: Senior Design Project II

(1-3) Cr. 2.

Prereq: S E 491

Second semester of a team design project experience. Emphasis on the successful implementation and demonstration of the design completed in S E 491 and the evaluation of project results. Technical writing of final project report; oral presentation of project achievements.

S E 494: Software Engineering Portfolio Development

Cr. R. F.S.

Prereq: Credit or enrollment in S E 491

Portfolio assessment for Software Engineers. Guidelines and Advice to improve software engineering portfolios and to better use portfolios as a tool to enhance career opportunities.