Harvey Mudd College
Claremont, California

Computer Science Curriculum

16 March 1992

This document defines the newly-instituted Computer Science major at Harvey Mudd College. Please address any questions or comments to:

Professor Robert M. Keller, Chair
Computer Science Department
110 Jacobs Science Center
Harvey Mudd College
Claremont, CA 91711

714-621-8483 or -8225
email: keller@jarthur

Table of Contents

Background and Goals

This document describes the Computer Science (CS) major at Harvey Mudd College (HMC) as approved by the faculty in December 1991. The purpose of the CS major is to provide a clear academic focus for the discipline of Computer Science. The following is a statement of goals and characteristics of the program:

a. To provide a solid education in the fundamental principles and concepts of computer science.

b. To include in the curriculum a blend of experiment, theory, and design.

c. To produce computer scientists who can communicate well and who integrate their broad background derived from HMC's extensive Common Core* curriculum of science, mathematics, and engineering as well as the strong humanities and social sciences program.

d. To prepare students for either graduate study in computer science or immediate employment in computer science and related fields.

We intend our curriculum to both addresses the needs of students and the professional community in providing graduates having unique, and academically- or professionally-attractive, profiles in combining fundamentals, design techniques, and applications†. The HMC CS student will be well-trained in core CS areas and qualified with "bridge" characteristics in two or more of the following dimensions:

Systems $\leftrightarrow$ Applications

Software $\leftrightarrow$ Hardware

Symbolic computing $\leftrightarrow$ Numeric computing

As students are necessarily exposed to a wide variety of application areas within their studies, we expect new cross-area fertilization to occur within the computational medium as well as elsewhere. This seems especially appropriate as enthusiasm mounts, within major research units nation-wide, toward "computational science", that is science done by computer modeling and simulation, as a third alternative to experimental and theoretical science.

The CS Department cooperates with all HMC departments in serving the needs of their students through appropriate core and optional elective courses. It cooperates with the relevant departments in the other Claremont Colleges in providing cross staffing of and access to courses of mutual interest.

---

* The common core includes structured programming and problem solving, system engineering, four semesters of mathematics, three semesters of physics, two semesters of chemistry, one semester of biology, plus two core electives chosen from: Discrete Mathematics; Principles of Computer Science; Modern Physics; Physical Chemistry or Carbon Compounds; any additional course in Biology, Engineering, or Humanities and Social Sciences.

† Notions regarding the content of CS curricula have become refined over the last two decades (in publications in years '68, '78, '83, '89, '91) by the predominant professional organizations concerned with CS as a discipline, namely the Association for Computing Machinery and the IEEE Computer Society.

Computer Science Major Requirements

The HMC CS major will produce CS professionals and future academics well-rounded and well-grounded in CS fundamentals. We plan to offer specializations to the extent allowed by the approved electives, in combination with senior research and clinic projects. The Computer Science major therefore consists of the following requirements:

CS 60 (Principles of Computer Science)	(elect in the Common Core)
CS kernel	6 courses
CS electives	3 courses
Math requirement	1 course
Senior project or clinic	2 courses
Total (above the HMC Common Core including CS60)	12 courses

When taken in the context of HMC graduation requirements, the CS major requirements leave approximately three courses which may be freely chosen.

Each of the components of the major is described below.

CS kernel requirements:

Programming Languages	(CS 131)
Computer Architecture	(CS 110)
Algorithms	(CS 140)
Operating Systems	(CS 124)
Software Development	(CS 121)
Scientific Computing	(CS 144)

Each of the above courses includes an essential laboratory component, in the form of computer-based development exercises and projects.

CS elective requirement (3 courses from the following):

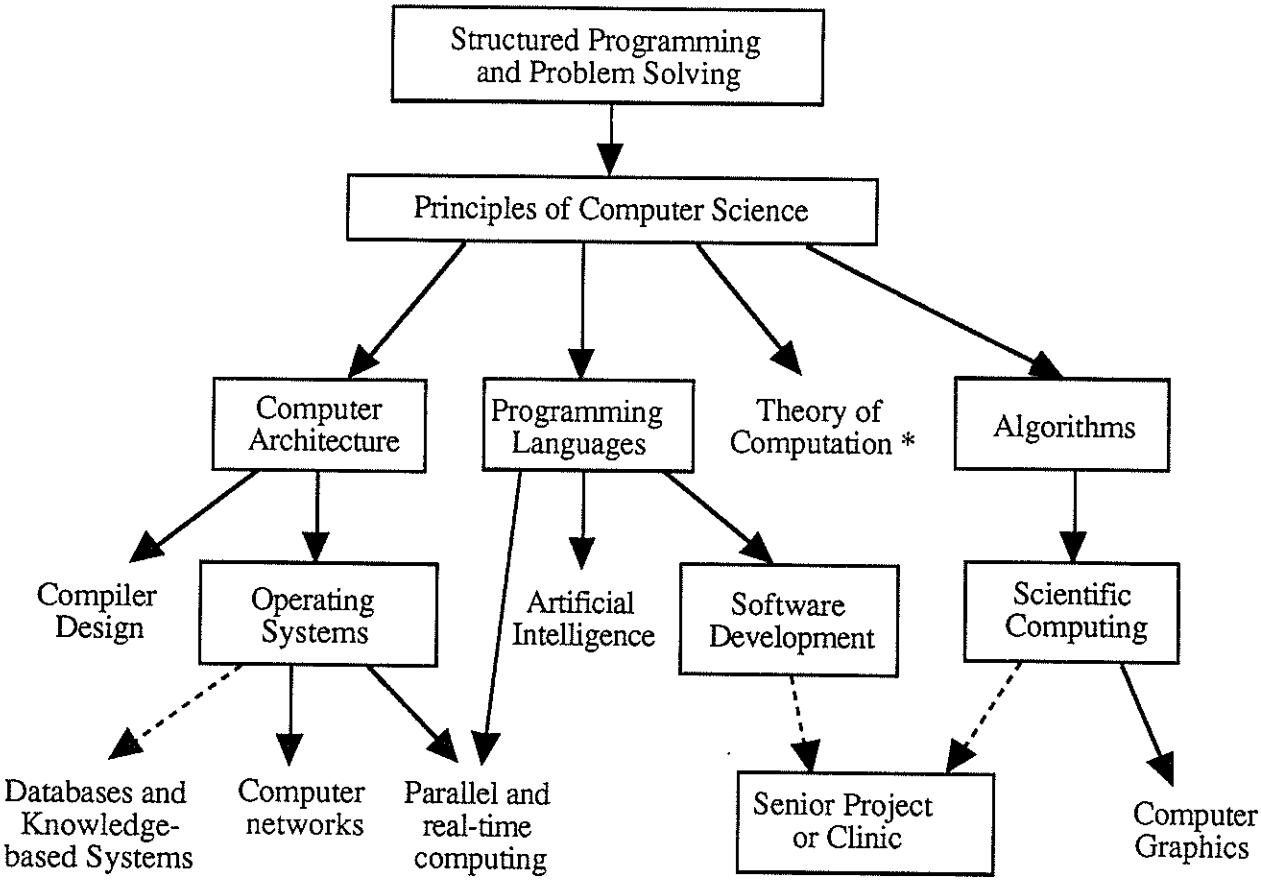
Theory of Computation	(CS 142)
Compiler Design	(CS 132)
Artificial Intelligence	(CS 151)
Computer graphics	(CS 155)
Databases and knowledge-based Systems	(CS 133)
Parallel and real-time computing	(CS 156)
Computer networks	(CS 125)
Seminar in Computer Science	(CS 181, 182)

Math elective requirement (1 course required):

Fundamental Concepts in Mathematics	(Math 121)
Logic	(Math 123)
Probability and its Applications	(Math 151)
Numerical Analysis	(Math 165)

Consistent with the tradition and requirements established in Engineering and Mathematics at HMC, each CS student is required to enroll in a year-long senior activity, either a thesis project or a clinic project. Continued cooperation with the Engineering and Mathematics clinics is planned, especially in this developmental phase.

Precedence Diagram for Computer Science Courses



denotes part of CS kernel; other courses are elective

denotes concurrent enrollment possible

* denotes recommended if graduate study is planned

Typical Major Course Sequence

Year	Fall	Spring
Freshman	Structured Programming	
Sophomore	Principles of Computer Science	Programming Languages Computer Architecture
Junior	Software Development Algorithms	Operating Systems Scientific Computing
Senior	Math Elective CS Elective Project/Clinic	CS Elective CS Elective Project/Clinic

A table indicating possible 4-year course selection sequences appears in the appendix.

Computer Science Department Faculty

Robert M. Keller, *Professor of Computer Science and Chair, Department of Computer Science, 1991.* B.S., M.S., Washington University; Ph.D., University of California, Berkeley; Assistant Professor, Princeton University; Visiting Assistant Professor, Stanford University; Associate and Full Professor, University of Utah; Director of Research and Vice President, Research and Development, Quintus Corporation; Professor and Chair, Division of Computer Science and Chair of the Graduate Group in Computer Science, University of California, Davis; Visiting Scientist, Lawrence Livermore National Laboratory. Professor Keller's interests are in languages and systems for parallel processing and in declarative programming languages.

Michael A. Erlinger, *Associate Professor of Computer Science, 1981.* B.S., University of San Francisco; M.S., Ph.D., University of California, Los Angeles; Lecturer, University of California, Los Angeles; Technical Staff Member, Bell Telephone Laboratories; Senior Project Engineer, Hughes Aircraft Company. Professor Erlinger's interests are in computer networking and network management.

Carol Moon, *Assistant Professor of Computer Science, 1990.* B.S., Ph.D., University of Michigan. Professor Moon's interests are in artificial intelligence, particularly natural language processing.

Wing Cheung Tam, *Professor of Engineering and Computer Science, 1974.* B.S., M.S., Ph.D., University of California, Los Angeles; Post Graduate Research Engineer, University of California, Los Angeles; Assistant Professor, Wayne State University; Senior System Programmer, McDonnell Douglas Corporation. Professor Tam's interests are in software engineering and real-time systems.

Affiliated Faculty

David L. Bosley, *Assistant Professor of Mathematics, 1991.* B.S., Stanford University; Ph.D., University of Washington; Assistant Professor, University of Washington. Professor Bosley's interests are in scientific computing.

Michael Townsend, *Assistant Professor of Mathematics, 1989.* B.A., M.A., Ph.D., University of Michigan; J.D., Yale University; Research Scientist, Bell Laboratories; Assistant Professor, Columbia University. Professor Townsend's interests are in algorithms and logic.

Ruye Wang. *Assistant Professor of Engineering, 1990.* M.S., Tianjin University, China; M.S., Ph.D., Rutgers University; Lecturer, Peking University, China. Professor Wang's interests are in image processing.

Everett L. Bull, *Associate Professor of Mathematics*, Pomona College.

James Pinter-Lucke, *Associate Professor of Mathematics*, Claremont McKenna College. Professor Pinter-Lucke's interests are in user interfaces and educational use of computers, especially the use of graphics to illustrate algorithms.

Paul Gray, *Professor of Information Sciences*, Claremont Graduate School. Professor Gray's interests are in the management of information systems and decision support systems, particularly group decision support systems.

Descriptions of Computer Science Courses

CS 5. Structured Programming and Problem Solving

Tam. Introduction to problem solving using the computer. Algorithms, data representation and structuring. The use of the computer in solving problems in science, engineering, and mathematics. Introduction to use of a programming language, operating system, and software tools. Specification, testing, debugging, and documentation. 2 credit hours. (First semester).

CS 50. Structured Programming

Moon. An introduction to problem solving using the computer. Fundamental concepts of programming languages. Top-down programming, program testing, debugging, and documentation. Emphasis is on development of good programming style using a structured programming language. Prerequisite: mathematical background equivalent to that required for admission to beginning college calculus. No programming experience required. 3 credit hours. (Students may not receive credit for both Computer Science 5 and 50.) (Second semester.)

CS 60. Principles of Computer Science

Keller. Introduction to principles of computer science. Algorithms, complexity analysis, data structuring, data and procedural abstraction, grammars, correctness, logic principles, processor organization, operating system concepts, programming languages, basic automata theory, and theoretical limitations. Prerequisites: Computer Science 5 or 50, and Mathematics 3 or equivalent. 3 credit hours. (Both semesters.)

CS 110. Computer Architecture

Wang. Principles of computer architecture, from logic design through processor and memory structure. Introduction to parallel computers and computer networks. Prerequisite: Computer Science 60. 3 credit hours. (Second semester.)

CS 121. Software Development

Tam. Rigorous introduction to the technological and managerial discipline concerned with the design and implementation of large software systems. Techniques for software specification, design, verification, and validation. Formal methods for proving the correctness of programs. Students working in teams are required to design, implement, and present a substantial software project. Prerequisite: Computer Science 60. 3 credit hours. (First semester.)

CS 124. Operating Systems

Erlinger. Principles and practice of operating systems. Concurrency and process structure. Memory-management, file-management, resource allocation, scheduling, information sharing and protection. Distributed computation and networking issues. Prerequisite: Computer Science 110. 3 credit hours. (Second semester.)

CS 125. Computer networks

Erlinger. Principles and analysis techniques for internetworking. Analysis of networking models and protocols. Presentation of computer communication with emphasis on protocol architecture. Prerequisite: Computer Science 124. 3 credit hours. (First semester.)

CS 131. Programming Languages
Keller. A thorough examination of issues and features in language design and implementation. Issues to be discussed include language-provided data structuring and data-typing, modularity, scoping, inheritance, and concurrency. Compilation and run-time issues. Introduction to formal semantics. Prerequisite: Computer Science 60. 3 credit hours. (Second semester.)

Comment: In addition to providing the elements of study, it is intended that knowledge of languages in this course can be relied upon to provide a vehicle for accelerated discourse in later courses.

CS 132. Compiler Design
Tam. The theory, design, and implementation of compilers and interpreters. The interaction between compiler design and run-time organization. Logistics of porting to new hardware. Prerequisite: Computer Science 110. Three credit hours. (Second semester.)

CS 133. Databases and Knowledge-based Systems
Keller. Fundamental models of databases: entity-relationship, relational, network. Knowledge bases using logic programming languages and logical data languages. Large file organization and distributed database systems. Prerequisites: Computer Science 124 and 131. 3 credit hours. (First semester.)

CS 140. Algorithms
Townsend. Algorithm design, computer implementation, and analysis of efficiency. Discrete structures, sorting and searching, parsing, pattern-matching, and data management. Reducibility and theoretical limitations. Prerequisite: Computer Science 60 and Mathematics 73. 3 credit hours. (Second semester.)

CS 142. Theory of Computation (joint-listed as Mathematics 167)
Townsend. Automata-theoretic concepts. Finite automata. Languages. Computability and undecidability. Applications to logic, mathematics, and system analysis. Prerequisite: Computer Science 60. 3 credit hours. (First semester.)

Comment: Although not required, CS 142 is strongly recommended for students planning on graduate study.

CS 144. Scientific Computing (joint-listed as Mathematics 164)
Bosley. Computational techniques applied to problems in the sciences and engineering. Modeling of physical problems, computer implementation, analysis of results; use of mathematical software; numerical methods chosen from: solutions of linear and nonlinear algebraic equations, solutions of ordinary and partial differential equations, finite elements, linear programming, optimization algorithms, and fast-Fourier transforms. Prerequisites: Math 73 and 82, Computer Science 60. 3 credit hours. (Second semester.)

CS 151. Artificial Intelligence
Moon. Knowledge representation, including rule-based systems and neural networks, learning paradigms, and philosophical challenges to artificial intelligence. Discussion of areas of current research: natural language processing, robotics, vision, cognitive modelling, case-based-reasoning. Prerequisite: Computer Science 131. 3 credit hours. (Second semester.)

CS 152. Neural Networks
Busenberg and Mueller. Modeling, simulation and analysis of artificial neural networks and their relation to biological networks. Relation between dynamic properties and computational properties of large, interconnected parallel processors. Introduction to nervous systems. Design and optimization of discrete and continuous neural networks. Back propagation, simulated annealing and internal learning rules. Simulation of single neurons and of large interconnected networks. Prerequisites: Biology 52 and Mathematics 73 and 82, or permission of the instructor. 3 credit hours. (Second semester.)

CS 153. Organism and Robot Sensory Systems.
Mueller. Sensors and sensory processing in animal and machine. Sensory neurobiology and methods for pattern recognition and scene analysis. Robot sensory systems with emphasis on vision. Prerequisite: sophomore standing. 3 credit hours. (First semester.)

CS 155. Computer graphics
Wang. Geometric models for visual output. Rastering. Three-dimensional volume and surface modeling. Reflectance and illumination models. Texturing and shading. Color and animation. Prerequisite: Computer Science 144. 3 credit hours. (First semester.)

CS 156. Parallel and real-time computing
Keller and Tam. Characteristics and applications for parallel and real-time systems. Specification techniques, architectures, languages, design, and implementation. Prerequisites: Computer Science 124 and 131. 3 credit hours. (Second semester.)

CS 181, 182. Computer Science Seminars
Staff. Advanced topics of current interest in computer science. Prerequisite: permission of instructor. 3 credit hours. (Both semesters.)

CS 183, 184. Computer Science Clinic I, II
Staff. Team project in computer science, with corporate affiliation. Prerequisites: Computer Science 121 and 144. 3 credit hours.

CS 185, 186. Senior Research Project I, II
Staff. Individual research project as required for the Computer Science major (as an alternate to CS 183, 184). Prerequisite: Senior standing. 3 credit hours.

CS 191, 192. Computer Science Project I, II.
Staff. Participation in projects of substantial interest to computer scientists. Emphasis is on the design and implementation of computer software or hardware for real problems. Students typically work in small teams with faculty supervision. Prerequisite: permission of the department chair. 1-3 credit hours per semester.

CS 197, 198. Advanced Problems in Computer Science.
Staff. Independent study in a field agreed upon by student and instructor. 1-3 credit hours per semester.

Sample Four-Year Programs

Year Sem.	Course Purpose	Course Title	Dept.	Number	Program Orientation and Emphasis (numbers represent credit units) Graduate School Theory	Graduate School Systems	Industry Hardware	Industry Software
1 Fall	Core Requirement	**Structured Programming**	CS	5	2	2	2	2
	Core Requirement	Calculus	Math	3	4	4	4	4
	Core Requirement	Mechanics and Wave Motion	Physics	23	2	2	2	2
	Core Requirement	General Chemistry, Lab	Chem	21;25	4	4	4	4
	Core Requirement	Rhetoric	HumSoc	1	4	4	4	4
	Core Requirement	Physical Education	PhysEd		0	0	0	0
	Total Hours				16	16	16	16
1 Spring	Core Requirement	Intro. to Biology	Biology	52	3	3	3	3
	Core Requirement	Multivariable Calculus	Math	4	3	3	3	3
	Core Requirement	Mechanics and Wave Motion; Lab	Physics	24;28	4	4	4	4
	Core Requirement	General Chemistry, Lab	Chem	22;28	4	4	4	4
	Core Requirement	Intro. to Humanities and Social Sciences	HumSoc	2	3	3	3	3
	Core Requirement	Physical Education	PhysEd		0	0	0	0
	Total Hours				17	17	17	17
2 Fall	Major & Core Elective	**Principles of Computer Science**	CS	60	3	3	3	3
	Core Requirement	Linear Algebra	Math	73	3	3	3	3
	Core Requirement	Electromagnetic Theory and Optics; Lab	Physics	51;53	4	4	4	4
	Core Requirement	Intro. to Systems Engineering	Engr	53	3	3	3	3
	HumSoc Requirement	Humanities and Social Sciences	HumSoc		3	3	3	3
	Core Requirement	Physical Education	PhysEd		0	0	0	0
	Total Hours				16	16	16	16
2 Spring	Major Requirement	**Programming Languages**	CS	131	3	3	3	3
	Major Requirement	**Computer Architecture**	CS	110	3	3	3	3
	Core Requirement	Differential Equations	Math	82	3	3	3	3
	Core Elective	Discrete Mathematics	Math	55	3			3
	Elective	Intro. to Electrical Engineering	Engr	96			3	
	HumSoc Requirement	Humanities and Social Sciences	HumSoc		3	3	3	3
	HumSoc Requirement	Humanities and Social Sciences	HumSoc			3		
	Total Hours				15	15	15	15
3 Fall	Major Requirement	**Software Development**	CS	121	3	3	3	3
	Major Requirement	**Algorithms**	CS	140	3	3	3	3
	Math Elective	Logic	Math	123	3			
	Math Elective	Numerical Analysis	Math	165		3		3
	Elective	Electronics; Lab	Engr	163;139			4	
	HumSoc Requirement	Humanities and Social Sciences	HumSoc		3	3	3	3
	HumSoc Requirement	Humanities and Social Sciences	HumSoc		3	3	3	3
	Total Hours				15	15	16	15
3 Spring	Major Requirement	**Operating Systems**	CS	124	3	3	3	3
	Major Requirement	**Scientific Computing**	CS	144	3	3	3	3
	HumSoc Requirement	Humanities and Social Sciences	HumSoc		3	3	3	3
	HumSoc Requirement	Humanities and Social Sciences	HumSoc		3	3	3	3
	Total Hours				12	12	12	12
4	Major Requirement	**Senior Project / Clinic**	CS	183/185	3	3	3	3
	Major Electives	**Databases & Knowledge-Based Systems**	CS	133	3			3
	(elect 2)	**Computer Graphics**	CS	155		3		3
		Theory of Computation	CS	142	3	3		
		Computer Networks	CS	125			3	
		Microprocessor System Design	Engr	157			3	
	Math Elective	One of the following			3	3	3	3
		Fundamental Concepts in Mathematics	Math	121				
		Numerical Analysis	Math	165				
		Probability and its Applications	Math	151				
	HumSoc Requirement	Humanities and Social Sciences	HumSoc		3	3	3	3
	HumSoc Requirement	Humanities and Social Sciences	HumSoc		3	3	3	3
	Total Hours				18	18	18	18
4	Major Requirement	**Senior Project / Clinic**	CS	184/186	3	3	3	3
	Major Elective	**Compiler Design**	CS	132				3
	(elect 1)	**Artificial Intelligence**	CS	151	3			
		Parallel and Real-Time Computation	CS	156		3	3	
	HumSoc Requirement	Humanities and Social Sciences	HumSoc		3	3	3	3
	HumSoc Requirement	Humanities and Social Sciences	HumSoc		3	3	3	3
	Total Hours				12	12	12	12
	Additional units to be taken elected				7	7	6	7
	Total credit hours				128	128	128	128