COMPUTER SCIENCE

HANDBOOK for

UNDERGRADUATES

1986 - 1987

Department of Computer Science

University of Washington

114 Sieg Hall, FR-35

Seattle, Washington 98195

(206) 543-1695
# Table of Contents

- Introduction  
- The B.S. Degree  
- The Honors Program  
- Courses for Non-Majors  
- Further Information and Advice  
- Admissions Procedures  
  - Prerequisites for Application  
  - Application Information  
  - Selection Process  
- Information for Majors  
  - Requirements for a Degree of Bachelor of Science  
  - Approved Senior Electives  
  - Scheduling  
- Computer Science Courses  
- Computer Science Faculty
Introduction

Computer Science is the study of information and algorithms within the context of real and abstract computing devices. Computer Scientists are interested in topics including: the representation and storage of information, algorithms to access, display, edit, and transform information; programming and mathematical languages to express algorithms; and hardware and software processors to execute algorithms. These concerns lead both to theoretical investigations of computers, algorithms, and data and also to practical developments in computer technology and applications.

The Department of Computer Science includes approximately 20 faculty members, 160 graduate students, and 160 undergraduate majors. The emphasis in the Department is divided equally among undergraduate instruction, graduate instruction, and research. In a recent study of graduate departments of Computer Science, our Department was ranked among the top ten in the nation. While no similar study has been performed concerning undergraduate education, this study is a good indication of the high overall quality of our Department. In addition, our student/faculty ratio is probably the lowest among highly rated Computer Science departments with undergraduate programs.

The purpose of this handbook is to acquaint prospective majors with our undergraduate program, to provide information about the application procedure, to list the requirements for our majors, and to give some basic background on our Department.

The B.S. Degree

Since 1975, the Department has offered an undergraduate degree program in which students can pursue a Bachelor of Science degree in the College of Arts and Sciences. The Department of Computer Science is actively seeking qualified students with an interest in Computer Science. Our emphasis is on providing a strong background in the fundamental issues involved in computing. Our courses can be divided into three categories: theory, systems, and applications.

- The theory courses are concerned with abstract notions of computing that are for the most part independent of any specific computing technology. Topics of this sort include the design of efficient algorithms and the analysis of the basic computational requirements of any algorithm solving a particular problem. The purpose of dealing with abstract rather than real computers in these courses is to facilitate focusing on the key issues involved in computing, rather than becoming sidetracked by exploiting the peculiarities of a particular implementation.

- Courses in the systems area are concerned with the logical structure
"architecture") of computer systems, and with the design and implementation of computer system software, such as operating systems and compilers. In contrast to theory courses, where you may do no actual programming, an important component of learning in the systems area is implementation. Courses in this area give the student better understanding of the technological implementations of computers.

- Finally, courses in the applications area deal with computers as tools to solve problems, as contrasted with the previously described courses where computers themselves are the object of study. Of course, this distinction is not clear-cut. The application areas considered require a thorough understanding of computer theory and systems, since a high regard for computational efficiency typically is required. Examples of topics in this area are artificial intelligence and computer graphics.

The purpose of the education provided by our Department is to produce Computer Scientists, as contrasted with computer users. The undergraduate program is designed to prepare students for professional careers or graduate study. We are especially concerned with providing our students with a foundation that will not become obsolete as technology advances and changes.

Upon graduation, our students are well prepared for either of two possible futures. One is graduate study. Our students regularly are accepted by the best graduate departments in the country. Another is a career requiring persons highly skilled in designing, analyzing, implementing, and maintaining hardware and software systems. Typical employers for such people include computer hardware and software manufacturers (Digital Equipment Corporation, Intel, Microsoft, Apple, IBM, and Hewlett-Packard, to name but a few), companies relying heavily on the development of computerized systems (e.g., Tektronix, Weyerhaeuser, John Fluke, Bell Northern Research, Boeing, and Lucas Films), and myriad other companies that are involved with computer consulting or that own substantial computing facilities. In short, the student is prepared for jobs in which the skilled element is a thorough understanding of computing.

On the other hand, an undergraduate degree in Computer Science is inappropriate if you are looking for employment of either of two kinds. The first is as a routine programmer, where you implement and debug software that is specified and designed by some third party. This kind of routine programming does not usually require university training. The second is a job in which the primary skill is in some other field, and the computer is used as a tool. An example of this might be the work done by an economist. The primary skill in this case would be in understanding economic models and forecasting.

Computers might be used to aid in this modelling, but no deep understanding of them would be required. In this case, training in economics would provide a more appropriate background.

The Honors Program
An Honors Program for Computer Science majors in Arts and Sciences was established in 1977. Students may graduate "with College Honors" by completing both the college honors requirements AND the computer science honors requirements or they may graduate "with Distinction" by completing only the Departmental honors requirements. A handout describing the Computer Science Honors Program is available from the Undergraduate Curriculum Advisor.

Courses for Non-Majors
The Department of Computer Science offers several computer science courses for non-computer science majors. Currently, these include CSci 210, 211, 373, 410, and 413. Non-Computer Science majors may petition CSci courses. Forms must be submitted by the week before the quarter begins. Decisions are made during the first week of the quarter. Students should plan to attend classes being petitioned during that week.

Further Information and Advice
Handouts concerning the undergraduate program, as well as application materials are available in the Computer Science main office. The Department also holds regular orientation meetings for prospective applicants. Further information about these and a time schedule for the meetings can be obtained from the receptionist in the Computer Science office. Additionally, the receptionist can make appointments for you to see the Undergraduate Curriculum Advisor, who is responsible for day-to-day advice and any information pertinent to curriculum, and the Undergraduate Faculty Advisor, a faculty member in the Department, who is responsible for the overall direction of the undergraduate program.
Admissions Procedures

Prerequisites for Application
Any student applying to the Department of Computer Science Undergraduate Program must have completed the following requirements prior to the deadline date.

1. Earn a minimum of 45 Quarter credits, applicable towards a Bachelor's Degree.

2. Complete the following courses: Math 124, 125, and 126; Physics 121, 122, and 123; and CSci 210/211 (if you have taken CSci 201 or 241, make an appointment to see the Undergraduate Curriculum Advisor).

3. Have at least a 3.0 as your overall GPA.

4. Be student of or applicant to the University of Washington (applicants must apply to the University through the Office of Admissions).

Application Information
Each applicant must be sure the Undergraduate Curriculum Advisor has the following items prior to the deadline date.

1. A completed application form (available in 114 Sieg Hall).

2. A statement of purpose, approximately 200 words (neatly presented), that describes motivation, goals, related work experience, special circumstances not evident from transcripts, explanation of non-UW transcripts, etc.

3. Copies of complete transcripts from ALL universities and colleges attended (official transcripts or legible copies accepted).

The deadline date for completed applications depends on the quarter in which you wish to commence as a major. For students wishing to enter the program starting in Autumn Quarter, the deadline is the preceding July 1st. For students wishing to enter the program starting in Spring Quarter, the deadline is the preceding January 15th.

Selection Process
The Computer Science Undergraduate Admissions Committee meets shortly after application deadlines to select students for admission. Enrollment is limited due to space, faculty, and budgetary limitations. Those applicants who, in the judgement of the admissions committee, are most likely to succeed in the program and contribute to the field will be admitted. Selection is made primarily on the basis of scholastic achievement and potential. Other selection criteria, such as related work experience, is also considered. Applicants are notified of their admission or denial by letter 4-6 weeks after the above deadlines. Students who are denied admission may discuss their application by meeting with either the Undergraduate Curriculum Advisor or the Undergraduate Faculty Advisor.

Information for Majors

Requirements for a Degree of Bachelor of Science
There are three components (180 credits total):

1. GENERAL EDUCATION COMPONENT (90 credits): This consists of the distribution requirements defined by the College and the University. Some courses specifically required in other components may apply to these distribution requirements. The Department strongly recommends that students take 10 credits of Natural Science, Business, or Engineering in addition to the natural science distribution requirement.

2. PREPARATORY COMPONENT (39 credits):
   - Physics (12 credits): Physics 121, 122, 123.
   - Math (24 credits): Math 124, 125, 126; choose 3 courses from the following: Math 238, 239, 301, 302, 303, 305, any of the Math or Statistics courses in the senior elective component, or Stat 311. Econ 261 does not meet this requirement. Stat 301.
   - Electronics (3 credits): One of Physics 334, EE 306, 310, or 355.

3. COMPUTER SCIENCE MAJOR COMPONENT (51 credits): To graduate, a grade of at least 2.0 must be received in each course in the major component.
• Inner Core Component (31 credits): Majors must take ALL of these classes: CSci 210, 211, 321, 322, 326, 341, 378.

• Outer Core Component (12 credits): Majors must select at least 12 credits of these classes: CSci 401, 421, 431, 451, 470, 473.

• Senior Elective Component (8 credits): Majors must select at least 8 credits from courses in the following groups: 400-level CSci courses (not including those used to satisfy the Outer Core), up to 6 credits of CSci 498 Senior Project (Honors students must take exactly 9 credits), and courses chosen from the approved senior elective course list (included below).

It is recommended, but not required, that students augment their Computer Science major with concentrated work in Physics, Mathematics, Economics, or Electrical Engineering. Although computers are worth studying in themselves, they are also used as tools by other disciplines. Therefore, knowledge of another discipline will allow computer scientists to apply their expertise in computers to that discipline. For students planning to go to graduate school in Computer Science, it is suggested that they take additional courses in Mathematics, emphasizing senior level courses in abstract algebra, analysis, logic, probability, and statistics.

**Approved Senior Electives**

<table>
<thead>
<tr>
<th>EE</th>
<th>417, 418</th>
<th>Introductory Communication Theory - (4,3)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>471</td>
<td>Computer Architecture and Structure - (3)</td>
</tr>
<tr>
<td></td>
<td>475</td>
<td>Digital Electronics and Microprocessors - (4)</td>
</tr>
<tr>
<td></td>
<td>476</td>
<td>Logical Design of Digital Devices - (3)</td>
</tr>
<tr>
<td></td>
<td>478</td>
<td>Design of Computer Subsystems - (4)</td>
</tr>
<tr>
<td></td>
<td>479</td>
<td>Microcomputer System Design - (5)</td>
</tr>
<tr>
<td>Geog:</td>
<td>365</td>
<td>Introduction to Computer Cartography - (5)</td>
</tr>
<tr>
<td></td>
<td>463</td>
<td>Microcomputer Proc. of Geog. Databases - (5)</td>
</tr>
<tr>
<td></td>
<td>465</td>
<td>Computer Cartographics - (5)</td>
</tr>
<tr>
<td>Ling:</td>
<td>461, 462</td>
<td>Syntax - (3,3)</td>
</tr>
<tr>
<td>Math:</td>
<td>402, 403, 404</td>
<td>Introduction to Modern Algebra - (3,3)</td>
</tr>
<tr>
<td></td>
<td>405</td>
<td>Introduction to Metamathematics - (3)</td>
</tr>
<tr>
<td></td>
<td>407, 408</td>
<td>Mathematical Optimization Theory - (3,3)</td>
</tr>
<tr>
<td></td>
<td>414, 415</td>
<td>Number Theory - (3,3)</td>
</tr>
<tr>
<td></td>
<td>461, 462</td>
<td>Combinatorial Theory - (3)</td>
</tr>
<tr>
<td></td>
<td>464, 465, 466</td>
<td>Numerical Analysis I, II, III - (3,3)</td>
</tr>
<tr>
<td>Mech E:</td>
<td>477</td>
<td>Real-Time Microcomputer Appl. in Mech E - (4)</td>
</tr>
<tr>
<td>Physics:</td>
<td>434</td>
<td>Appl. of Computers to Physical Meas. - (3)</td>
</tr>
<tr>
<td>Quan Meth:</td>
<td>470, 472</td>
<td>System Design and Development I, II - (4,4)</td>
</tr>
<tr>
<td>Quan Sci:</td>
<td>381</td>
<td>Introduction to Probability &amp; Statistics - (5)</td>
</tr>
<tr>
<td></td>
<td>390</td>
<td>Prob. and Stat. in Engineering and Science - (4)</td>
</tr>
<tr>
<td></td>
<td>421</td>
<td>Intro. to Applied Stat. and Experimental Design - (4)</td>
</tr>
<tr>
<td></td>
<td>no credit if credit received for Stat 341 or 390</td>
<td></td>
</tr>
<tr>
<td>Stat/Math:</td>
<td>394,395,396</td>
<td>Probability I, II, III - (3,3)</td>
</tr>
<tr>
<td></td>
<td>491,492</td>
<td>Introduction to Stochastic Processes - (3,3)</td>
</tr>
</tbody>
</table>

**Scheduling**

The schedule for majors has been carefully planned; hence, the courses, especially the 300-level courses, must be taken at specific times. Majors who do not take the designated 300-level courses according to the schedule are not guaranteed entry into the courses in later quarters. (If you have unavoidable conflicts with this schedule, promptly see the Undergraduate Curriculum Advisor.) For the 400-level courses, entry is not guaranteed. However, preference will be given to majors interested in taking courses in the designated quarters.

The following table shows the schedules for those majors entering in Autumn and in Spring quarters. Required courses are shown in boldface. Parentheses indicate that the specified course may be taken at that time, although it would not be a regular occurrence. CSci 440 is marked with an asterisk because it may only be taught every other year.
## Computer Science Courses

CSci 210/211 Computer Science I, II (5.5) AWSpS
Integrated two-quarter introduction to Computer Science. Emphasis on four areas: (1) introductory programming as a serious discipline; (2) elementary data structures and algorithms; (3) reasoning about the correctness and efficiency of programs; and (4) the structure of computer systems. A modern programming language, such as Modula-2, is introduced and used. CSci 210 must be completed before CSci 211 is taken.

CSci 321 Discrete Structures (3) ASp
Fundamentals of set theory, graph theory, enumeration, and algebraic structures, with applications in computing. Prerequisite: Math 126 and CSci 241. CSci majors only.

CSci 322 Introduction to Formal Models in Computer Science (3) AW
Finite automata and regular expressions; context free grammars and pushdown automata; nondeterminism; Turing machines and the halting problem. Emphasis on understanding models and their applications and on rigorous use of basic techniques of analysis. Induction proofs, simulation, diagonalization and reduction arguments. Prerequisites: CSci 321. CSci majors only.

CSci 326 Data Structures (5) AW
Data types, abstract data types, and data structures. Efficiency of algorithms. Sequential and linked implementation of lists. Binary tree representations and traversals. Searching: dictionaries, priority queues, hashing. Directed graphs, depth-first algorithms. Garbage collection. Dynamic storage allocation. Internal and external sorting. Prerequisites: CSci 321. No credit if CSci 373 or EE 374 have been taken. CSci majors only.

CSci 341 Programming Languages (5) WSp
Designed to make the student reasonably fluent in several radically different languages, such as LISP, ICON, APL, and SIMULA. Prerequisite: CSci 211. CSci majors only.

CSci 373 Data Structures and Algorithms (3) ASp
Fundamental algorithms and data structures for their implementation. Techniques for solving problems by programming. Linked lists, stacks, queues, directed graphs, trees: representations and traversals. Searching (hashing, binary search trees, multiway trees). Garbage collection, memory management. Internal and external sorting. Prerequisite: CSci 211. For non-CSci majors only. No credit if CSci 326 or EE 374 have been taken.

CSci 378 Machine Organization and Assembly Language (5) ASp
Difference and similarities in machine organization; central processors, fundamentals of machine language and addressing; assembly language programming, including macros; operating system interfaces. Prerequisite: CSci 211. No credit if EE 371 has been taken. CSci majors only.

CSci 401 Introduction to Compiler Construction (3) ASp
Fundamentals of compilers and interpreters; symbol tables, lexical analysis, syntax analysis, semantic analysis, code generation, and optimization for general purpose programming languages. Offered jointly with EE 401. Prerequisites: CSci 326 and CSci 378. CSci majors only.

CSci 410 Computer Systems (3) A
Structure and components of hardware and software systems. Machine organization, including central processor and input-output architectures; assembly language programming; operating systems, including process, storage, and file management. Prerequisite CSci 373. Non-CSci majors. No credit to students who have taken CSci 378 or CSci 451.
CSci 413 Programming Languages and Their Implementation (3) W
Concepts and implementation strategies for Algol-class languages, including PASCAL, MODULA, ALGOL 60, ADA. Compilers for Algol-class languages. Languages with late binding times, including LISP, APL, SMALLTALK. Prerequisite: CSci 373. Non-CSci majors. No credit to students who have taken CSci 341 or CSci 401.

CSci 421 Introduction to the Analysis of Algorithms (3) W
Techniques for design of efficient algorithms. Methods for showing lower bounds on computational complexity. Particular algorithms for sorting, searching, set-manipulation, arithmetic, graph problems, pattern-matching, etc. Prerequisites: CSci 322 and CSci 326. CSci majors only.

CSci 431 Introduction to Theory of Computation (3) Sp
Models of computation, computable and non-computable functions, space and time complexity, tractable and intractable functions. Prerequisite: CSci 322. CSci majors only.

CSci 440 Computer Based Simulation (4) A
Computer based simulation is a valuable tool with wide varying application. In this course, Monte Carlo, continuous time, and discrete event simulations are presented. The design of appropriate simulation experiments and the interpretation of their results is considered. Students will implement simulations using PASCAL, DYNAMO, and GPSS. Prerequisite: CSci 211 and CSci 326. Majors only course. Some familiarity with probability and/or statistics may be helpful.

CSci 444 Introduction to Database Systems (3) Sp
Fundamental concepts, system organization and implementation of database systems will be covered. Topics include the relational, hierarchical and network data models, file organizations and data structures, query languages, query optimization, database design, concurrency control, security, and issues involving distributed database systems. Prerequisite: CSci 326. CSci majors only.

CSci 451 Introduction to Operating Systems (3) WSp
Principles of operating systems. Process management, memory management, auxiliary storage management, resource allocation. Prerequisites: CSci 326 and CSci 376. No credit if EE 474 has been taken. CSci majors only.

CSci 457 Computer Graphics (3) Sp
Overview of the techniques of computer image synthesis, including both hardware and software. Line drawing and color raster graphics. Homogeneous coordinates, hidden surface and smooth shading algorithms. Prerequisite: CSci 326. CSci majors only.

CSci 470 Computer Design (4) AW
Fundamental gating circuits are developed into large logic gating structures. The use of these structures in the design of central processing units, memories, and peripheral equipment is illustrated. Prerequisite: CSci 378. CSci majors only.

CSci 473 Introduction to Artificial Intelligence (3) W
Principal ideas and developments in artificial intelligence: theorem proving, problem-solving methods; representation of knowledge, natural language analysis and synthesis, programming languages for artificial intelligence. Prerequisites: CSci 326, CSci 431 recommended. CSci majors only.

CSci 498-498H Senior Project (1-6; Honors 9) AWSpS
Consists of a report (and perhaps demonstration) describing a development, survey, or small research project completed by the student in an area of specialization. Objectives are: (1) applying and integrating the classroom material from several courses, (2) introducing the professional literature, (3) gaining experience in writing a technical document, and (4) enhancing employability through the evidence of independent work. The project may cover an area in computer science or an application to another field. The work normally extends over more than one quarter. Prerequisite: senior standing. For CSci majors only. Entry cards required.*

CSci 499 Reading and Research (1-24) AWSpS
Available in special situations for advanced computer science majors to do reading and research in the field. Subject to approval of the undergraduate advisor and a Computer Science faculty member. Usable as a free elective, but not in place of a core course or Computer science elective. Prerequisite: Senior standing and permission of instructor. Credit/no credit. Entry cards required.*

* CSci 498 - 498H and CSci 499 are the only undergraduate CSci courses requiring entry cards. Students must complete and have faculty advisor sign appropriate form, available in 114 Sieg, before picking up entry card from the Undergraduate Curriculum Advisor.

The Department of Computer Science offers a wide range of advanced, graduate level courses. Computer Science undergraduate majors may petition for entry to these courses, which are listed in the Computer Science Graduate Handbook and the University General Catalog.
Computer Science Faculty
Jean-Loup Baer, Professor of Computer Science and Adjunct Professor of Electrical Engineering (1969). Diplome d’Ingenieur, 1960, Doctorat 3e Cycle, 1963, Grenoble; Ph.D., 1968, UCLA. Parallel processing, system architecture, data structures.


Hellmut Golde, Professor of Computer Science and Adjunct Professor of Electrical Engineering (1959). Dipl.-Ing., 1953, Technische Hochschule Munich; M.S., 1955, Ph.D., 1959, Stanford. Programming languages, programming systems, compilers.


Jerre D. Noe, Professor of Computer Science and Adjunct Professor of Electrical Engineering (1968). B.S., 1943, Berkeley; Ph.D., 1948, Stanford. Distributed computer systems, operating systems, simulation, performance evaluation.


The University of Washington, as a standing policy, does not discriminate against individuals because of their race, color, religion, age, sex, national origin, handicap, or status as a disabled veteran or Vietnam era veteran. Any discriminatory action can be a cause for disciplinary action. This policy applies to all University programs and facilities including, but not limited to, admissions, educational programs, and employment. Such discrimination is prohibited by Titles VI and VII of the Civil Rights Act of 1964, Title IX of the Education Amendments of 1972, Sections 503 and 504 of the Rehabilitation Act of 1973, Age Discrimination in Employment Act Amendments of 1978, Vietnam Era Veterans' Readjustment Assistance Act of 1974, and other federal and state statutes and regulations. Coordination of the compliance efforts of the University of Washington with respect to all of these laws and regulations is under the direction of the Equal Employment Officer, Dr. Philip W. Cartwright, 140 Administration, AF-16, Seattle, Washington 98195, (206) 543-7630.