UNDERGRADUATE HANDBOOK
FOR COMPUTER SCIENCE

Department of Computer Science
University of Washington
114 Sieg Hall, FR-35
Seattle, Washington 98195
(206) 543-1695
Introduction

The Department of Computer Science evolved from the Computer Science Group, a graduate interdisciplinary group that has offered M.S. and Ph.D. degrees since 1969. Since 1975, the department has offered an undergraduate program in which students can pursue a Bachelor of Science degree under the College of Arts and Sciences.

Computer science is the study of information and algorithms within the context of real and abstract computing devices. Computer scientists are interested in: 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 to practical developments in computer technology and applications.

The department currently offers undergraduate courses in programming, discrete structures, data structures, machine organization, computer design, operating systems, programming languages, compilers, artificial intelligence, design and analysis of algorithms, and theory of computation. The department also requires its majors to take courses in mathematics, physics and electronics.

The objective of Computer Science education is to develop professionally competent and broadly educated computer scientists. Undergraduate education is designed to prepare students for professional careers or graduate study; especially important is a foundation that will not become obsolete as technology advances and changes.

The computer field has a broad base of industrial and governmental users, providing many of the jobs suitable for the B.S. graduate. Typical jobs are systems analyst, systems programmer, scientific programmer analyst, technical salesperson, hardware or software specialist. Above this base is a pyramid of producers and developers of computer systems, as well as teachers and researchers. Graduate education is appropriate for many of the jobs at these higher levels.

ADMISSIONS PROCEDURE

Pre-Selection Requirements

A student may apply to enter the Computer Science program provided:

1. The following courses or equivalent have been completed:
   - Math 124, 125, 126
   - Physics 121, 122, 123
   - Computer Science 201

2. The student has completed at least 45 credit hours applicable to University graduation requirements.

3. The student has an overall University grade point average of at least 3.0

4. The student has been admitted to the University or has completed an application to the University through the Office of Admissions.

Application Information

Any student having completed the pre-selection requirements may apply. APPLICATIONS THAT ARE MISSING COMPLETE TRANSCRIPTS AND/OR ANY PRE-SELECTION REQUIREMENTS WILL NOT BE CONSIDERED.

The following must be submitted by the deadline dates:

1. An application form (available in the Computer Science Department 114 Sieg Hall).
2. A typed, 200-word statement of purpose describing career goals.
3. Copies of complete transcripts from ALL universities and colleges you have attended. (Official transcripts or legible copies accepted.)

Application Deadline Dates

Autumn Quarter - April 15
Winter Quarter - October 15
Spring Quarter - January 15

Selection Process

The Computer Science Undergraduate Admissions Committee will meet each quarter during the academic year to select students for admission for the following quarter. Selection of applicants will be made primarily on the basis of scholastic achievement and potential. Other selection criteria, such as related work experience and grades in Computer Science preparatory courses and in all mathematics, science and engineering courses, will also be considered. Applicants will be notified of their admission or denial by letter 2-3 weeks after the above deadlines. Students who are denied admission may request reconsideration of their application by meeting with the chairman of the admissions committee (make appointments in the main office).
Admission Policy

Objectives and Justification for an Admission Policy:

Enrollment is limited due to space, faculty, and budgetary limitations. Those applicants who, in the judgment of the faculty, are most likely to succeed in the program and contribute to the field will be admitted.

Requirements for a Degree of Bachelor of Science

There are four components: (180 credits total)

1. General Education Component
   Distribution requirements are determined by the quarter and year you entered the University of Washington. These distribution lists are available in any advising office.
   
   Your natural science distribution requirement can be satisfied with your Math and Physics classes.
   
   Free electives: 39 credits

2. Preparatory Component
   Math 124, 125, 126, 205, 238; Physics 121, 122, 123
   Electronics: Choose one from Physics 334, EE 306, EE 310, or EE 355
   (only CSci & EE majors may ask for 310 and 395 entry cards)
   10 credits of Natural Science, Business, or Engineering.

3. Computer Science Core Component (42 credits) *
   CSci 201, 241, 321, 322, 326, 341, 378, 470, Math 464 and 6 credits of CSci 498

4. Computer Science Senior Elective Component (13 credits)
   A student may satisfy this requirement by taking courses on the Approved Computer Science Senior Electives List (page 12 of this brochure) or by taking graduate courses in Computer Science.

To graduate, a Computer Science major must receive a grade of at least 2.0 in each course in (3) and (4) above.

* The Computer Science Core component is currently under consideration for revision. Changes are likely to be implemented in the later part of 1984. It is the student's responsibility to keep informed of changes.

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 in 114 Sieg.

Sample Curricula

Entering Computer Science at Sophomore Level:

<table>
<thead>
<tr>
<th></th>
<th>Autumn Qtr</th>
<th>Winter Qtr</th>
<th>Spring Qtr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Freshman</td>
<td>Math 124 (5)</td>
<td>Math 125 (5)</td>
<td>Math 126 (5)</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Sophomore</td>
<td>Math 205 (3)</td>
<td>Math 238 (3)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Phys 123 (4)</td>
<td>*Electronics (3)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>CSci 241 (3)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Junior</td>
<td>CSci 321 (3)</td>
<td>CSci 322 (3)</td>
<td>CSci 326 (5)</td>
</tr>
<tr>
<td></td>
<td>CSci 378 (5)</td>
<td>CSci 341 (5)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Math 464 (3)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Senior</td>
<td>CSci 470 (6)</td>
<td>CSci 498 (6)</td>
<td>CSci 498 (6)</td>
</tr>
</tbody>
</table>

COMMUNITY COLLEGE TRANSFERS - Entering CSci at Junior Level:

At a community college take: (90 credits)

Math 124, 125, 126, 205, 238 & Physics 121, 122, 123 = 41 credits plus general education credits = 90 credits

At University of Washington take:

Computer Science 201 (summer before sophomore year)
Computer Science 241 (summer before junior year)

<table>
<thead>
<tr>
<th></th>
<th>Autumn Qtr</th>
<th>Winter Qtr</th>
<th>Spring Qtr</th>
</tr>
</thead>
<tbody>
<tr>
<td>Junior</td>
<td>CSci 321 (3)</td>
<td>CSci 322 (3)</td>
<td>CSci 326 (5)</td>
</tr>
<tr>
<td></td>
<td>CSci 378 (5)</td>
<td>*Electronics (3)</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Senior</td>
<td>CSci 470 (6)</td>
<td>CSci 498 (6)</td>
<td>CSci 498 (6)</td>
</tr>
</tbody>
</table>

Fill in blanks with distribution requirements, proficiency (if necessary) and 13 credits of senior electives = 180 credits required for one degree.

It is recommended, but not required, that students augment their computer science major with concentrated work in some application area like Physics, Mathematics, Economics, Accounting, or Electrical Engineering. The reason for this recommendation is that, although computers are worth studying in themselves, they are also used as tools by other disciplines. 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 or probability and statistics.
Undergraduate Courses

CSci 201 Introduction to Computer Science (5) AMSpSu
A rigorous introduction to the theoretical and practical components of computer science: algorithms, programs, data structures, machines, computability, applications, social aspects. Prerequisite: MATH 124. Freshmen & sophs only for preregistration AMSp, all others by permission of instructor after quarter starts.

CSci 241 Programming (3) AMSpSu
Basic algorithms, programming techniques, and basic concepts of the structured high-level language Pascal. Prerequisite: 201 or permission of instructor or dept. advisor. Undergrads only during preregistration in AMSp, all others by permission of instructor after quarter starts.

CSci 321 Discrete Structures (3) A
Fundamentals of set theory, graph theory, Boolean algebra, and algebraic structures with applications in computing. Prerequisite: MATH 126. CSci majors only.

CSci 322 Introduction to Formal Models in Computer Science (3) W
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 on rigorous use of basic techniques of analysis. Induction proofs, simulation, diagonalization and reduction arguments. Prerequisites: 241 and 321. CSci majors only.

CSci 362 Data Structures (5) Sp
Sequential and linked allocation of linear structures; tables, arrays, stacks, queues; in-core searching and sorting; circular and doubly linked lists; trees and threaded lists; dynamic memory allocation and garbage collection. Prerequisites: 321 and 378. CSci majors only.

CSci 341 Programming Languages (5) W
Designed to make the student reasonably fluent in several radically different languages, such as LISP, SNOBOL, APL, ALGOL 60, Pascal, SIMULA 67, and others. Prerequisite: 241. CSci majors only.

CSci 373 Data Structures and Algorithms (3) AMSp
Fundamental algorithms and data structures for their implementation. Techniques for solving problems by programming. Sorting, searching, linked lists, binary search trees, balanced trees, hashing. Offered jointly with EE 373. Prerequisite: 241 or 445 or equivalent knowledge of Pascal. For non-CSci majors, no credit if 326 has been taken.

CSci 378 Machine Organization and Assembly Language (5) A
Differences and similarities in machine organization, central processors, fundamentals of machine language and addressing, assembly language programming, including macros; operating system interfaces. Prerequisite: 241. CSci majors only. (EE 371 accepted as a substitute.)

CSci 401 Introduction to Compiler Construction (3) W
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: 326 plus 378, or EE 371 plus CSci/EE 373, or instructor’s permission.

CSci 421 Introduction to the Analysis of Algorithms (3)
Analysis of behavior of algorithms. Techniques for design of efficient algorithms. Methods for showing lower bounds on computational complexity. Discussion of particular algorithms for sorting, searching, set-manipulation, arithmetic, graph problems, pattern-matching, and their implementations. Prerequisite: 322 and 326.

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: 322. (Offered alternate years.)

CSci 451 Introduction to Operating Systems (3) W
Principles of multiprogramming systems. Process management, resource management, and file systems. Prerequisite: 326 or CSci/EE 373 or instructor’s permission.

CSci 470 Computer Design (4) AM
Fundamental gating circuits are developed into large logic gate structures. The use of these structures in the design of central processing units, memories, and peripheral equipment is illustrated. For computer science majors. Prerequisite: 378 or instructor’s permission. Entry card required for registration, first priority to CSci Seniors - all others must get instructor’s permission before requesting the entry card.

CSci 473 Introduction to Artificial Intelligence (3) Sp
Principal ideas and developments in Artificial Intelligence; LISP as the basis of precise descriptions of AI processes; theorem proving, problem-solving methods; representation of knowledge in procedures, and in frames; natural language analysis and synthesis; inference and generation from conceptual representations. Prerequisites: 326 and 341, or CSci/EE 373.

CSci 498-499H- Senior Project (1-6, max. 6; Honors 1-9, max. 9) AMSp
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 Computer Science majors only. Entry cards are required.
CSci 520 Computer Science Seminar (1, max. 9) AWP
Weekly discussion by students and faculty or visitors on current topics of interest. Offered on credit/no credit basis only.

CSci 521 Design and Analysis of Algorithms I (3) W

CSci 522 Design and Analysis of Algorithms II (3) Sp
Analysis of algorithms more sophisticated than those treated in CSci 521. Design of efficient algorithms for special computing environments such as logical networks and formulas, Turing machines, list processing, machines, and on-line computation. Techniques for proving lower bounds on complexity. Prerequisite: 521.

CSci 531 Formal Languages and Automata (3) A
Formal models in Computer Science: including context-free grammars, finite automata, regular expressions, Turing machines, and pushdown automata. The fundamental concepts of nondeterminism, undecidability, and syntax and semantics are covered.

CSci 532 Complexity Theory (3) W
Space and Time complexity of various models of computation such as Turing machines, random access machines and list processing machines. Hierarchies based on complexity, time and space bounded reducibility, NP-completeness and other complexity classes, and provably difficult problems.

CSci 533 Computability and Logic (3) Sp
Formal systems that characterize the notion of computation and model the notion of logical reasoning. Computability of recursively enumerable sets, the recursion theorem, and diagonalization methods. Logic includes first order predicate logic, nonstandard models, proof systems, the completeness theorem, and undecidable theories. (Offered alternate years.)

CSci 540 Discrete System Simulation (3) A
Principles of simulation of discrete, event-oriented systems. Model construction, simulation and validation, and relationship to other techniques for system analysis and design. Use of special-purpose simulation languages such as S2000A and study of functional components and data structures. Prerequisite: programming experience with ALCOL.

CSci 542 Central Processor Architecture (3) Sp
Several central processing units are examined at the gate level. Included are the logic structures of: I/O bus, memory bus, ALU, address modification, control logic, combinational and multipath instructions, access priority, cycle stealing, etc. Prerequisite: 540. (Offered alternate years.)
CSci 543 Analytic Models of Computer Systems (3) W
Emphasizes the use of queueing network models as tools for analyzing computer systems. Topics include useful results from basic queueing theory, selection of performance measures, modeling methodology, data acquisition, computational algorithms for queueing network models, approximation techniques, and decomposability and hierarchical modeling. A realistic case study will be undertaken.

CSci 568 Computer Systems Architecture (3) W
Notations for describing computer systems. Powerful CPUs. Memory organization. Channels and I/O processors. Micro programming. Stack computers. Array and pipeline processors. 470 to be taken concurrently or instructor's permission.

CSci 551 Operating Systems (3) Sp
Operating systems design and construction techniques. Systems programming languages, concurrent programming, design methodologies, protection, deadlock problems, virtual memory allocation, and other topics. Study of the structure of different kinds of operating systems. Prerequisite: 451 or instructor's permission.

CSci 557 Computer Graphics (3) A

CSci 561 Computer Communications and Networks (3) A
Fundamentals of data transmission: coding, message formats, and protocols; job and data management problems; organization of computer networks. A number of networks are studied, and students are expected to prepare a class presentation of a network. Offered on credit/no credit basis only. (Offered alternate years.)

CSci 573 Artificial Intelligence I (3) A
Introduction to the use of the computer in nonnumerical problem solving. Survey of theorem proving, symbolic manipulation, pattern recognition, and inductive problem-solving techniques. Computer models of human thought. Prerequisite: 508 and knowledge of LISP or permission.

CSci 574 Artificial Intelligence II (3) W
Continuation of studies of artificial intelligence systems, emphasizing theorem proving, symbolic problem solving, pattern recognition, and natural language data processing. Students are required to do projects. Prerequisite: 573 or permission. (Offered alternate years.)

CSci 590 Special Topics in Computer Science (*) AWSp
Lectures and discussions of topics of current interest in computer science. May not be offered every quarter; content may vary from one offering to another. Prerequisite: instructor's permission.

CSci 600 Independent Study or Research (*) AWSpSu Cr/no cr only.
CSci 700 Master's Thesis (*) AWSpSu Cr/no cr only.
CSci 800 Doctoral Dissertation (*) AWSpSu Cr/no cr only.

Petitioning Courses Restricted to Computer Science Majors:
We are forced by limited resources to restrict the enrollment in many of our courses.
Courses 321, 322, 326, 341, and 378 at the junior level are for Computer Science undergraduate majors only. Other students who meet the following requirements can petition for entry cards:
- all prerequisites have been satisfied for entry into our undergraduate program,
- all prerequisites have been satisfied for the specific course, and
- the student will be applying to our undergraduate major program during the quarter for which the petition is being filed.

Effective Winter Quarter 1984, courses 401, 421, 431, 451, 470, and 473 at the senior level are for Computer Science undergraduate majors and Computer Science graduate students only. Other students who meet the following requirements can petition for entry cards:
- all prerequisites have been satisfied for entry into our undergraduate major program,
- all prerequisites have been satisfied for the specific course, and
- either the course is listed as an approved elective for the student's home department, or the student is a US or graduate student from another department.

Effective Winter Quarter 1984 all graduate courses are for Computer Science graduate students only. Other students can petition for entry cards. Among the criteria that will be applied in judging petitions are the student's general academic background, specific preparation for the course and the importance of the course to the student's program.

Petition forms are available in 114 Sieg Hall.
Approved Computer Science Senior Electives
(Schedules subject to change)

Majors must choose a minimum of 13 credits from this list, see page 4.

CSci 401 Introduction to Assemblers and Compilers (3) W
CSci 421 Introduction to Analysis of Algorithms (3) A
CSci 431 Introduction to Theory of Computation (3) Sp
CSci 451 Introduction to Operating Systems (3) W
CSci 473 Introduction to Artificial Intelligence (3) Sp

A student may also take graduate courses in Computer Science to satisfy the elective component. See the instructor to determine the background necessary for grad-level courses.

The following courses offered by other departments are acceptable as Computer Science Electives. Check for prerequisites, course entry requirements, and schedules with the department offering the course.

EE 417, 418 Introductory Communication Theory I, II (4, 3) W, Sp
EE 475 Digital Electronics and Microprocessors (4) AMSp
EE 476 Logical Design of Digital Devices (3) WSp
EE 479 Microcomputer System Design (3) WSp
Geog 365 Intro to Computer Cartography (3) W
Math 402, 403, 404 Intro to Modern Algebra (3, 3, 3) A, W, Sp
Math 405 Introduction to Nematemathematics (3) Sp
Math 407, 408 Mathematical Optimization Theory (3, 3) WSp
Math 465, 466 Numerical Analysis II, III (3, 3) W, Sp
Phys 434 Application of Computers to Physical Measurement (3) WSp
Quantitative Methods 470 System Design & Development I
Quantitative Methods 472 System Design & Development II
Quantitative Science 381 Introduction to Probability and Statistics (5) AMSSp #see Note 1

Stat 341 Elementary Probability (3) AMSp formerly Math 391 #see Note 1
Stat 342 Elements of Statistics (3) WSp formerly Math 392 #see Note 1
Stat 390 Probability and Statistics in Engineering and Science (4) AMSSp #see Note 1

*Note 1: Either Quantitative Science 381 or Math 391 and 392 or Stat 341 and 342 or Stat 390 may be used as Computer Science Electives. Math 391 and 392 are now offered as Stat 341 and 342 in the Dept. of Statistics.

The Faculty

The year the faculty member joined the University is in parentheses.


Hellmut Gollnitz, Professor of Computer Science and Adjunct Professor of Electrical Engineering (1959). Dipl.-Ing., 1953, Technische Hochschule Munich; M.S., 1955, Ph.D., 1959, Stanford. (On leave 83-84) Programming languages, programming systems, compilers.


Adjunct and Lecturer Appointments


David L. Johnson, Professor of Electrical Engineering. B.S.E.E., 1948, Idaho; Ph.D., 1955, Purdue. Switching Theory and logical design, models of learning, concept formation, man-machine interaction.


Pierre A. MacKay, Professor of Classics, Near Eastern Languages and Literature, and Comparative Literature. B.A., 1954, Yale; M.A., 1959, Ph.D., 1964, California (Berkeley). Multi-lingual text-editing and typesetting, especially Arabic script; graphics; peripheral design.


The University of Washington, as a standing policy, does not discriminate against individuals because of their race, color, religion, discriminate against individuals because of their race, color, religion, discriminate against individuals because of their race, color, religion, discriminate against individuals because of their race, color, religion, discriminate against individuals because of their race, color, religion, discriminate against individuals because of their race, color, religion, discriminate against individuals because of their race, color, religion, discriminate against individuals because of their race, color, religion, discriminate against individuals because of their race, color, religion, discriminate against individuals because of their race, color, religion, discriminate against individuals because of their race, color, religion, discriminate against individuals because of their race, color, religion, discriminate against individuals because of their race, color, religion, discriminate against individuals because of their race, color, religion, discriminate against individuals because of their race, color, religion, discriminate against individuals because of their race, color, religion, discriminate against individuals because of their race, color, religion.