COMPUTER SCIENCE
HANDBOOK for
UNDERGRADUATES
1989 - 1990

Department of Computer Science and Engineering
University of Washington
114 Sieg Hall, FR-35
Seattle, Washington 98195
(206) 543-1695
COMPUTER SCIENCE
HANDBOOK for
UNDERGRADUATES
1989 - 1990

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

1 Introduction ................................................. 4
  1.1 The B.S. Degree ........................................... 4
  1.2 The Honors Program ...................................... 6
  1.3 Courses for Non-Majors .................................. 6
  1.4 Minor/Mathematical Sciences Option ...................... 6
  1.5 Further Information and Advice .......................... 6
  1.6 Laboratory Equipment .................................... 6

2 Admissions Procedures ...................................... 7
  2.1 Prerequisites for Application ............................. 7
  2.2 Application Information .................................. 7
  2.3 Selection Process ....................................... 8

3 Information for Majors ..................................... 8
  3.1 Requirements for a Degree of Bachelor of Science ....... 8
  3.2 Approved Senior Electives ................................. 10
  3.3 Scheduling ................................................ 10
  3.4 Graduation Application/Exit Interview .................... 11
  3.5 Engineering Co/op Program ............................... 11
  3.6 The Placement Center ..................................... 11

4 Computer Science Honors Program .......................... 11

5 Computer Science Courses for Non-majors .................. 12

6 Computer Science Courses for Majors ........................ 13

7 Computer Science Faculty .................................. 15

8 Instructions for Admission to Computer Science Undergraduate Degree Program ................................................. 18
  8.1 Application Deadline Dates ................................ 18
  8.2 Application Packet ........................................ 18
1 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 and Engineering began in 1967 when a small group of faculty from a number of departments were authorized to grant graduate degrees in Computer Science. In 1975, the Department of Computer Science was formed and began offering a Bachelor of Science degree in Computer Science, with the first B.S. degree awarded in 1978. In 1989, the department moved from the College of Arts and Sciences into the College of Engineering. It assumed responsibility for the undergraduate Computer Engineering program, previously offered by the Department of Electrical Engineering, and changed its name accordingly.

Today the Department of Computer Science and Engineering includes approximately 25 faculty members, 150 graduate students, and 250 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 in Computer Science, to provide information about the application procedure, to list the requirements for our majors, and to give some basic background on our Department.

1.1 The B.S. Degree

The Department offers two undergraduate Bachelor of Science degrees: a degree in Computer Science, offered through the College of Arts and Science, and a degree in Computer Engineering, offered through the College of Engineering. This handbook offers information only about the Computer Science degree; for information on the Computer Engineering degree, please refer to the handbook entitled “Handbook for Undergraduates in Computer Engineering”.

The Department of Computer Science and Engineering is actively seeking qualified students with an interest in Computer Science and Computer Engineering. Our emphasis is on providing a strong background in the fundamental issues involved in computing. Our courses in the Computer Science curriculum 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 (or "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 and Computer Engineers, as contrasted with computer users. The undergraduate programs are 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 are regularly 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 manufacturer of computer hardware and software manufacturers (Digital Equipment Corporation, Intel, Microsoft, Apple, IBM, Xerox, Bell, etc.), research laboratories (Tektronix, U.S. West, etc.), and large consulting firms.

Upon graduation, our students are well prepared for either of two possible futures. One is graduate study. Our students are regularly 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 manufacturer of computer hardware and software manufacturers (Digital Equipment Corporation, Intel, Microsoft, Apple, IBM, Xerox, Bell, etc.), research laboratories (Tektronix, U.S. West, etc.), and large consulting firms.

Computer Science lends itself nicely to double majors. Many fields require the use of the computer as a tool. Consider economics. The primary skill in this case would be in understanding economic models and forecasting (which must be gained through earning an economics degree). Most modeling is currently done using computers, so strength in Computer Science might also be useful, depending on the degree to which you are involved in constructing the computer models. This symbiotic relationship holds between Computer Science and most other fields as well. However, due to the large overlap a double major of Computer Science with Computer Engineering is not allowed.

1.2 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. Details on the Honors Program appear later in this handbook.

1.3 Courses for Non-Majors

The Department of Computer Science offers several courses for non-computer science majors. Currently, these include CSE 210, 211, 373, 410, 413, and 415. Non-Computer Science majors may petition CSE courses. Forms must be submitted by the week before the quarter begins. Because decisions are usually made during the first week of the quarter, students should plan to attend petitioned classes during that week.

1.4 Minor/Mathematical Sciences Option

It is possible to earn a BS degree with a Mathematical Sciences option with Computer Science being the science option. Application should be made early in your academic career through the Math Department, CSE Pedellord.

If you are considering a minor in CSci, available through the College of Education, you should see an advisor in the Office of Teacher Education Advising, 211 Miller Hall.

1.5 Further Information and Advice

Handouts concerning the undergraduate programs are available in the Computer Science and Engineering main office, 114 Sieg Hall. The Department also holds regular orientation meetings for prospective applicants. Further information about these meetings, including a time schedule, can be obtained from the receptionist in 114 Sieg Hall. In addition, the receptionist can make appointments for you to see either the Undergraduate Academic Counselor, responsible for day-to-day advice and information pertinent to curriculum, or the Undergraduate Faculty Advisor, a faculty member in the Department who is responsible for the overall direction of the undergraduate program.

1.6 Laboratory Equipment

The focus of our program is to teach major concepts and principles that go beyond any individual class of computers. However, we also realize that exposing our students to state-of-the-art equipment is educational, motivational, and fun. We currently have a variety of equipment available...
2 Admissions Procedures

2.1 Prerequisites for Application

Any student applying to the Department of Computer Science Undergraduate Program should have completed the following requirements prior to application. (See the following section for application deadlines.)

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

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

3. Have at least a 3.0 as your overall GPA. Your overall GPA is the average of all transfer units and all UW units combined.

4. Be currently enrolled as a student at the University of Washington, or made application to enroll through the Office of Admissions, 320 Schmitz.

2.2 Application Information

Each applicant is responsible for providing the Undergraduate Academic Counselor with the following items prior to the deadline date.

1. A completed application form. (Application is attached at the center of this brochure.)

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 will be accepted).

The deadline date for completed applications depends on the quarter in which you wish to start 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.

Under no circumstances will students who are admitted to the undergraduate program be permitted to delay their entrance to the program. Students not entering on schedule will be required to reapply and are not guaranteed admission by virtue of their previous acceptance.

2.3 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 Academic Counselor or the Undergraduate Faculty Advisor.

3 Information for Majors

3.1 Requirements for a Degree of Bachelor of Science

There are three components (150 credits total):

1. GENERAL EDUCATION COMPONENT (30 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 121, 125, 126, and 3 courses selected from Math 301, 307, 308, 309, any of the Math or Statistics courses in the senior elective component, or Stat/Econ 311. (Formerly: Math 238, 239, 240, 302, 303, and 304 could also be used in satisfying this requirement. These courses have been discontinued by the Math Department, but students who have already taken them may still use them to satisfy this requirement.)
   - 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: CSE 210, 211, 321, 322, 326, 341, 378.
- Outer Core Component (12 credits): Majors must select at least 12 credits of these classes: CSE 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 CSE courses (not including those used to satisfy the Outer Core), up to 6 credits of CSE 498 Senior Project (Honors students must take exactly 9 credits), and courses chosen from the approved senior elective course list (included below). Courses not on the list can still apply toward the senior elective component, subject to approval by the Undergraduate Faculty Advisor.

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.

3.2 Approved Senior Electives

<table>
<thead>
<tr>
<th>Course Code</th>
<th>Course Title</th>
</tr>
</thead>
<tbody>
<tr>
<td>401, 402</td>
<td>Syntax - (3.3)</td>
</tr>
<tr>
<td>417, 418</td>
<td>Introductory Communication Theory - (4.3)</td>
</tr>
<tr>
<td>471</td>
<td>Computer Architecture and Structure - (3)</td>
</tr>
<tr>
<td>476</td>
<td>Logical Design of Digital Devices - (3)</td>
</tr>
<tr>
<td>478</td>
<td>Design of Computer Subsystems - (4)</td>
</tr>
<tr>
<td>479</td>
<td>Microcomputer System Design - (5)</td>
</tr>
<tr>
<td>486</td>
<td>Formal Semantics and Natural Language - (3)</td>
</tr>
<tr>
<td>489</td>
<td>Principles of Cartography - (5)</td>
</tr>
<tr>
<td>493</td>
<td>Microcomputer Proc. of Geom. Databases - (5)</td>
</tr>
<tr>
<td>495</td>
<td>Analytical Cursography - (5)</td>
</tr>
<tr>
<td>502, 503, 504</td>
<td>Introduction to Modern Algebra - (3.3)</td>
</tr>
<tr>
<td>507, 408</td>
<td>Mathematical Optimization Theory - (3.3)</td>
</tr>
<tr>
<td>514, 515</td>
<td>Number Theory - (3.3)</td>
</tr>
<tr>
<td>543, 544</td>
<td>Combinatorial Theory - (3)</td>
</tr>
<tr>
<td>547</td>
<td>Numerical Analysis I, II, III - (4, 4, 4)</td>
</tr>
<tr>
<td>417</td>
<td>Real-Time Microcomputer Appl. in Mech E - (4)</td>
</tr>
<tr>
<td>434</td>
<td>Appt. of Computers to Physical Sys. - (3)</td>
</tr>
<tr>
<td>400, 401</td>
<td>Systems Analysis and Design I, II - (4, 4)</td>
</tr>
<tr>
<td>545</td>
<td>Introduction to Probability &amp; Statistics - (5)</td>
</tr>
<tr>
<td>403</td>
<td>Style in Scientific and Technical Writing - (3)</td>
</tr>
<tr>
<td>407</td>
<td>Computer Documentation - (3)</td>
</tr>
<tr>
<td>408</td>
<td>Special Documents: Proposals, EIS, and Manuals - (3)</td>
</tr>
<tr>
<td>409</td>
<td>Writing for Publication - (3)</td>
</tr>
<tr>
<td>341, 342</td>
<td>Introduction to Prob. &amp; Stat. Inference - (4.4)</td>
</tr>
<tr>
<td>421</td>
<td>Intro. to Applied Stat. and Experimental Design - (4)</td>
</tr>
<tr>
<td>484</td>
<td>Intro. to Math. Statistics - (5)</td>
</tr>
<tr>
<td>306, 308.355</td>
<td>Probability I, II, III - (3.3)</td>
</tr>
<tr>
<td>306.491.012</td>
<td>Introduction to Stochastic Processes - (3.3)</td>
</tr>
</tbody>
</table>

3.3 Scheduling

The schedule for majors has been carefully planned; hence, 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 Academic Counselor. In particular, students working towards a major in two departments including Computer Science should schedule an appointment if our projected schedule is infeasible.

Schedule for 300-level courses:

Majors entering in Autumn quarter
Autumn  CSE 321  CSE 378
Winter  CSE 322  CSE 326
Spring  CSE 341

Majors entering in Spring quarter

Spring  CSE 321  CSE 378
Autumn  CSE 322  CSE 326
Winter  CSE 341

3.4 Graduation Application/Exit Interview

At least two quarters before completing the requirements for your BS degree, you should make
an appointment with the undergraduate academic counselor, 114 Sieg, to apply for graduation.
Also at that time you will be asked to complete a brief questionnaire about the department. It can
be completed in a few minutes and will be helpful to the department in responding to the many
questions we receive. It will also help us to build a stronger program and be of assistance to future
students.

3.5 Engineering Co/op Program

The Engineering Cooperative Education and Minority Intern Program, located in 353 Loew Hall,
 aids students interested in regular short term employment in local industry while studying at the
University. For more information on the program contact Helene Beaver at 543-8711.

3.6 The Placement Center

The University's Placement Center, located in 301 Loew Hall, offers extensive services for students
searching for career and job opportunities. Many local, national, and international companies send
recruiters to campus each year to interview students in Computer Sciences and other disciplines.

4 Computer Science Honors Program

The Honors Program has been established since Autumn Quarter 1977 for those Computer Science
undergraduates who are within the college of Arts & Sciences.

Requirements for entry are:

1. An overall GPA of at least 3.3.
2. A GPA of at least 3.6 for all courses in the Computer Science core and elective components.
3. Affiliation with the College of Arts and Sciences.

To graduate with honors, a student must:

1. Maintain the above cumulative grade-point averages.
2. Take nine credits of CSE 498H, and complete a Senior Thesis, which must satisfy the following
   requirements:
   (a) It should be a typed document cleanly and clearly presented.
   (b) It should be done independently by the student under the supervision of a faculty
       member. The work should be original and demonstrate creative thinking, as judged by
       the faculty supervisor and the Undergraduate Faculty Advisor.
   (c) Final approval of the thesis must be obtained from the faculty supervisor and the Un-
       dergraduate Faculty Advisor.

Students who qualify for the Honors Program should request the appropriate forms from the Un-
dergraduate Staff Advisor in 114F Sieg Hall.

Note: Honors is noted on your diploma.

5 Computer Science Courses for Non-majors

(Subject to change)

CSE 205 Programming in LISP
An introduction to programming using a dialect of the language LISP. This course will cover simple
data structures (numbers, atoms, lists, and trees) and control structures (conditional, recursive, and
functional composition). Students will study and write small programs that manipulate symbolic
data. Prerequisite: Psych 354

CSE 210/211 Computer Science I, II

Integrated two-quarter, rigorous introduction to computer science via programming. Emphasizes
algorithms (types, declarations, expressions, statements), abstraction (data, subprograms, pack-
egages, generics), and analysis (EDNY, correctness, complexity, numerics). A modern programming
language (currently Ada) is used. CSE 211 ends with a three week overview of computer science
topics. Prerequisite: Math 124 for 210 and 210 for 211.

CSE 373 Data Structures and Algorithms (3)
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: CSE 211. No credit if CSE 326 or EE
374 have been taken.

CSE 410 Computer Systems (3)
Structure and components of hardware and software systems. Machine organization, including cen-
tral processor and input-output architectures; assembly language programming; operating systems.
including process, storage, and file management. Prerequisite CSE 373. No credit to students who have taken CSE 378 or CSE 451.

CSE 413 Programming Languages and Their Implementation (3) 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 CSE 373. No credit to students who have taken CSE 341 or CSE 401.

CSE 415 Introduction to Artificial Intelligence (5) Principles and programming techniques of artificial intelligence: Lisp, symbolic manipulation knowledge representation, logical and probabilistic reasoning, learning, language understanding, vision, expert systems, and social issues. Prerequisites: CSE 373, or CSci 211 and permission of instructor. (No credit if 473 has been taken.)

6 Computer Science Courses for Majors

(Subject to change)

CSE 321 Discrete Structures (4) Fundamentals of set theory, graph theory, enumeration, and algebraic structures, with applications in computing. Prerequisite: Math 126 and CSE 211.

CSE 322 Introduction to Formal Models in Computer Science (3) 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: CSE 321.

CSE 326 Data Structures (4) 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: CSE 321. No credit if CSE 373 or EE 374 have been taken. Prerequisite: CSE 321.

CSE 341 Programming Languages (5) Designed to make the student reasonably fluent in several radically different languages, such as Lisp, Icon, Prolog, and SMALLTALK. Not open for credit to students who have taken 413. Prerequisite: CSE 211.

CSE 376 Machine Organization and Assembly Language (3) Differences and similarities in machine organization; central processors; fundamentals of machine language and addressing; assembly language programming, including macros; operating system interfaces. Prerequisite: CSE 211.

CSE 401 Introduction to Compiler Construction (3) Fundamentals of compilers and interpreters: symbol tables, lexical analysis, syntax analysis, semantic analysis, code generation, and optimization for general purpose programming languages. Prerequisites: CSE 326, CSE 341, and CSE 378. No credit to students who have completed CSE 415.

CSE 421 Introduction to the Analysis of Algorithms (3) 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: CSE 322 and CSE 326.

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

CSE 440 Computer Based Simulation (3) 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: CSE 211 and CSE 326. Some familiarity with probability and/or statistics may be helpful.

CSE 444 Introduction to Database Systems (3) 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: CSE 326.

CSE 451 Introduction to Operating Systems (4) Principles of operating systems. Process management, memory management, auxiliary storage management, resource allocation. Prerequisites: CSE 326 and CSE 378. No credit if EE 474 has been taken.

CSE 457 Computer Graphics (3) Overview of the techniques of computer image synthesis, including both hardware and software. Line drawing and color raster graphics; geometric transformations, hidden surface and smooth shading algorithms. Prerequisite: CSE 326.

CSE 470 Computer Design (4) 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: CSE 378.

CSE 473 Introduction to Artificial Intelligence (3) 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: CSE 226, CSE 341 recommended.

CSE 486, 489H Senior Project (1-6 hours) Honors (9) 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 and permission of instructor.

CSE 499 Reading and Research (1-2)
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. Unable 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.

The Department of Computer Science and Engineering 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.

7 Computer Science Faculty


Tony DeRose, Assistant Professor of Computer Science (1985). B.S., 1981. UC Davis; Ph.D., 1985. UC Berkeley. Computer aided geometric design and modeling, graphical user interfaces, high resolution computer graphics.


7 CSE 498 - 499 and CSE 499 require entry code numbers. Students must complete and have faculty advisor sign appropriate form, available in 111 Step, before picking up entry code numbers from the Undergraduate Academic Counselor.

trace-driven methodology.

Hellmut Gable, Professor of Computer Science and Adjunct Professor of Electrical Engineering (1954). Dipl.-Ing., 1953, Technische Hochschule Munich; M.S., 1955, Ph.D., 1959, Stanford. Programming languages, programming systems, compilers.


analysis, computer graphics, artificial intelligence. (Sabbatical leave 1989 – 90.)


Paul Young, Professor of Computer Science (1982). B.S., 1959, Antioch; Ph.D., 1963, MIT. Computational complexity, computability, and connections with mathematical logic.


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' Reemployment Rights 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. Phillip W. Cartwright, 140 Administration, AF-16, Seattle, Washington 98195, (206) 543-7530.

8 Instructions for Admission to Computer Science Undergraduate Degree Program

8.1 Application Deadline Dates

<table>
<thead>
<tr>
<th>Quarter</th>
<th>Date</th>
</tr>
</thead>
<tbody>
<tr>
<td>Autumn Quarter</td>
<td>July 1</td>
</tr>
<tr>
<td>Spring Quarter</td>
<td>January 15</td>
</tr>
</tbody>
</table>

8.2 Application Packet

All information must be in the Computer Science Office by the deadline date. The packet must include:

1. Application with all sections completed.
   (a) University of Washington students obtain University of Washington transcript from Transcripts Office, 260 Schmitz, 543-5940.
   (b) Arts and Sciences pre-majors obtain transcripts from Central Advising Office, B10 Padelford.
   (c) If current grades are not on transcript, attach copies of grade sheets.
   (d) Copies of Complete transcripts from all Universities/Colleges attended.


3. Statement:
   (a) About 200 words, neatly presented.
   (b) Describe career goals, related work experience, special circumstances not evident from the transcripts.

Allow several weeks if an official transcript is being mailed.

Admission to Computer Science does not constitute admission to the University of Washington. Admission to Computer Science is contingent upon acceptance to the University of Washington.

Under no circumstances whatsoever will students who are admitted be permitted to delay their entrance to the program. Students not entering on schedule will be required to reapply to the program and are not guaranteed to be accepted again. You are responsible for knowing the requirements in effect at the time of application.