Computer Science

Undergraduate Handbook
1995-1996

University of Washington
Department of Computer Science and Engineering
Contents

1 Introduction 1
   1.1 The B.S. Degree ............................................. 1
   1.2 The Honors Program .......................................... 2
   1.3 Teaching Minor/Mathematical Sciences Option ............... 3
   1.4 Relation between the Computer Science and Computer Engineering Degrees 3
   1.5 Computing and Laboratory Facilities .......................... 4
   1.6 Further Information and Advice ............................. 4

2 Admissions Procedures 5
   2.1 Prerequisites for Application ............................... 5
   2.2 Application Information ..................................... 5
   2.3 Selection Process ........................................... 6
   2.4 Appeals Procedure .......................................... 6

3 Information for Majors 7
   3.1 Requirements for a Degree of Bachelor of Science .......... 7
   3.2 Satisfactory Progress ....................................... 8
   3.3 Academic Misconduct ....................................... 9
   3.4 Approved Senior Electives ................................. 10
   3.5 Graduation Application/Exit Interview ...................... 10
   3.6 Co-op Program ............................................. 11
   3.7 Center for Career Services ................................ 11
   3.8 Computer Science Honors Program .......................... 11

4 Computer Science and Engineering Courses for Pre-Majors and Non-Majors 12

5 Computer Science and Engineering Courses for Majors 13

6 Computer Science and Engineering Faculty 16
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 such topics as: 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 to practical developments in computer technology and applications, as well as to theoretical investigations of computers, algorithms, and data.

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 changed its name to the Department of Computer Science and Engineering, and assumed responsibility for the undergraduate Computer Engineering program, previously offered by the Department of Electrical Engineering.

Today the Department of Computer Science and Engineering includes over 30 faculty members, 150 graduate students, and 300 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 and engineering, 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 ratio of undergraduate students to faculty is excellent; we generally aim for a maximum class size of 40 students for our majors courses.

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 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 far more 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 a better understanding of the technological implementations of computers.

Courses in the applications area deal with computers as tools to solve problems. Examples of topics in this area are artificial intelligence, computer graphics, and databases.

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 computer hardware and software manufacturers (Digital Equipment Corporation, Intel, Microsoft, Apple, IBM, Xerox, Boeing, Bell Communications Research, Tektronix, U.S. West, Weyerhaeuser, 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 LucasFilms), 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.

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 in Section 3.8.

### 1.3 Teaching 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, C36 Padelford.

If you are considering a teaching minor in CS, to be used for the UW Teacher's Certification Program, you should also see an advisor in the Office of Student Services, 296 Miller Hall.

### 1.4 Relation between the Computer Science and Computer Engineering Degrees

This handbook contains information about the undergraduate program in Computer Science (CS), offered through the College of Arts and Sciences. The department also offers an undergraduate program in Computer Engineering (CompE), through the College of Engineering. Both programs lead to a Bachelor of Science degree. For detailed information on the CompE degree, please refer to the handbook entitled "Computer Engineering Handbook for Undergraduates."

Computer Science and Computer Engineering are closely related. Many of the Department’s junior courses are required of both CS and CompE majors, and many of the senior courses are taken by students in both majors. However, there are a number of differences between the two degree programs. The key differences are as follows:

1. CS students must satisfy the College of Arts and Sciences distribution requirements; CompE students must satisfy the College of Engineering distribution requirements.
2. CompE requires substantially more courses in computer hardware and electrical engineering than CS.
3. CS requires slightly more theory than CompE.
4. Both degrees require a certain number of credits of senior elective courses in the major or a closely related field. The list of CompE required and elective courses includes all the electives on the CS elective list, plus any 400-level majors course in Electrical Engineering (EE).

The CS major is more appropriate for students who want to have a double major with another Arts and Sciences program (for example, Mathematics or Economics), who want the additional flexibility of the CS requirements (the CompE major has more required courses and fewer electives), or who want more theory of computation. The CompE major is more appropriate for students interested in building systems that include both custom hardware and software components, who have strong interests in EE, or who specifically want an Engineering degree.

Both CS and CompE graduates have had excellent success in finding positions after graduation. Many positions are equally suitable for graduates with either degree. A CS degree would be an advantage for a position needing the knowledge of a double major with another Arts and Sciences area (for example, CS and Economics, or CS and Mathematics), or additional liberal
ars background. A CompE degree would be an advantage for a position involving designing and building systems with a combination of software and custom hardware. Also, some larger, engineering-oriented manufacturing companies favor CompE over CS degrees. For graduate study in Computer Science, or in Computer Science and Engineering, either degree is suitable—the quality of the applicant would be the determining factor for admission.

Admissions for CS and CompE are handled separately. Both are competitive, with more qualified applicants than spaces available.

1.5 Computing and Laboratory Facilities

The focus of our program is to teach concepts and principles that go beyond any individual class of hardware devices or computers. However, providing our students with state-of-the-art equipment is an essential part of the curriculum, as well as being motivational and fun. We currently have a variety of equipment available for use in undergraduate classes.

The primary instructional computing lab contains approximately 50 color and grayscale Xterminals, backed by three DECstation 5000s (RISC-based systems) running Unix. To support digital system design courses, the department operates a design laboratory consisting of 12 Macintosh IIIs, 10 PC systems (386, 486 and Pentium), and Tektronix logic analyzers and other test equipment. The computer graphics instructional laboratory contains 14 Silicon Graphics Indy R4400 color graphics workstations, and a collection of specialized video gear. The Silicon Graphics workstations have hardware and software support for real-time 3D graphics, and have allowed us to dramatically change the way graphics is taught, concentrating on the many interactive graphics applications that this technology allows, such as image morphing, 2-D and 3-D game design, visualization techniques, and animation. Finally, an instructional laboratory containing two dozen Intel PCs running Microsoft Windows NT supports programming language, database, and other courses. Students working on senior and honors projects often make use of more advanced equipment in the department, including parallel processors, VLSI design tools, and graphics and image processing equipment. All Computer Science students have access to campus, national and international networks.

1.6 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 monthly orientation meetings for prospective Computer Science applicants. Further information about these meetings, including a time schedule, can be obtained from the receptionist in the main office. In addition, the receptionist can make an appointment for you to see the Computer Science Undergraduate Staff Advisor, whose office is also in 114 Sieg, and who is responsible for day-to-day advice and information pertinent to curriculum. The Undergraduate Staff Advisor may also refer you to the Undergraduate Faculty Advisor, a faculty member in the Department who is responsible for the overall direction of the undergraduate program.

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 124, 125, and 126 (Calculus with Analytical Geometry), Physics 121 and 131 (Mechanics), CSE/ENGR 142 (Computer Programming I), and CSE 143 (Computer Programming II).

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 have made application to enroll through the Office of Admissions, 320 Schmitz.

Applicants who have satisfied these minimum requirements are qualified to apply for admission, and are assured that their applications will be given thorough consideration by the Admissions Committee. Unfortunately, due to its limited facilities, the Department cannot offer admission to all applicants who meet these minimum qualifications. Applications by students who have not satisfied the minimum requirements will not be considered.

2.2 Application Information

All information must be in the Computer Science Office by the deadline date, which is as follows:

1. A completed application form. (A form 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. Legible copies of complete transcripts from ALL universities and colleges attended (official transcripts or unofficial copies will be accepted). University of Washington students may obtain transcripts from the Transcripts Office (260 Schmitz Hall), Graduations and Academic Records (284 Schmitz Hall), or from the Undergraduate Advising Center (9 Communications Building). The Transcripts Office will only produce official UW Transcripts; the other offices will produce unofficial copies. If current grades are not on the transcript, attach copies of grade sheets. Allow several weeks if an official transcript is being mailed.

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 February 1.

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 to the undergraduate program 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 admission by virtue of their previous acceptance. You are responsible for knowing the requirements in effect at the time of application.

In conformity with a standing policy of the University, the Department provides equal opportunity in education without regard to race, color, national origin, sex, age or handicap in accordance with Title VI of the Civil Rights Act of 1964, Title IX of the Education Amendments, Section 504 of the Rehabilitation Act of 1973, and the Age discrimination Act of 1975.

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 4-5 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.

Strong written and oral communication skills are essential to success in Computer Science, both in the undergraduate program and later in the workplace or graduate school. For this reason, the statement of purpose will be evaluated by the admissions committee both on the basis of its content, and also as a sample of the student’s writing ability. Evidence of written and oral communication skills, for example grades in writing courses, will also be considered.

2.4 Appeals Procedure

If a student is denied admission to the program, he or she may wish to consult with the Undergraduate Staff Advisor for additional information and guidance. Such a student may request a review of the Admissions Committee decision by writing a letter to the Computer Science Undergraduate Staff Advisor, citing the reasons for his or her appeal. This letter must be submitted within two weeks after notification of denial is mailed. The only legitimate basis for an appeal is that there was an error in the data considered by the Admissions Committee. Incomplete or inaccurate records submitted by the applicant will not be considered as a basis for an appeal.

3 Information for Majors

3.1 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 requirements defined by the College and the University. Current comprehensive information on the general education requirements of the College of Arts and Sciences is available from the Computer Science Advisor or in the Undergraduate Advising Center. Communication courses are also highly recommended. The Department strongly recommends that students take 10 credits of Natural Science, Business, or Engineering in addition to the Natural World requirement.

2. MATHEMATICS AND SCIENCE COMPONENT (39 credits):

- Math (24 credits): Math 124, 125, 126, (Calculus with Analytic Geometry) and 3 courses selected from Math 307, 308, 309, any of the Math or Statistics courses in the senior electives component or Stat/Econ 311.
- Physics (15 credits): Physics 121, 123, 131, 132, 133 (Mechanics, Electromagnetism and Oscillatory Motion, Waves, and accompanying labs).

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 (32 credits): Majors must take all of these classes: CSE/ENGR 142, CSE 143, 321, 322, 326, 341, 370, 378.
- Outer Core Component (12 credits): Majors must take at least 4 classes from the following list: CSE 301, 303, 321, 431, 444, 451, 457, 471, and 473. If a student completes more than 12 credits applicable toward the Outer Core requirement, the additional credits may be applied to the Senior Elective requirement.
- Senior Elective Component (7 credits): Majors must select at least 7 credits from courses on the approved senior elective course list included below, not including credits used to satisfy the Outer Core. Up to 6 credits of CSE 498 (Senior Project) are allowed, except for honors students, who must take 9 credits of CSE 498H. 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 Satisfactory Progress

Due to limited resources and the need to maintain program quality, we are not able to accept all students who apply to study Computer Science. The Satisfactory Progress Policy has been developed to use the available resources effectively and to best serve our students. The major tenets of the policy concern time in the program and level of scholarship. In addition, the Satisfactory Progress Policy mandates compliance with English as a Second Language requirements.

1. All students must complete two or more courses applicable to their degree program each academic quarter in residence except for summer quarter. In exceptional circumstances, by approval of the Undergraduate Advisor a student may complete one course applicable to the degree program.

2. Graduation applications must be filed in accordance with University rules and regulations so that graduation takes place once all the requirements are satisfied. The University allows students to register as a non-matriculated student for one quarter immediately after the quarter of graduation.

3. A student who withdraws from the University without prior approval of the Department, or is dropped for non-payment of fees, will have forfeited his/her place in the program. Such a student must reapply for admission and, if re-admitted, must fulfill the requirements in effect at the time of re-admission.

4. The minimum acceptable grade for any required or elective Computer Science course is 2.0. If necessary, a course may be taken a total of two times to meet this requirement.

5. The student's overall grade point average must not fall below a 2.0. If a course is repeated, both grades will be used in computing the grade point average.

6. Students who have not completed their English as a Second Language (ESL) requirements must enroll in and satisfactorily complete at least one ESL class each quarter starting with their first quarter in the CS program and continue to take ESL courses each quarter until all ESL requirements are completed.

3.2.1 Probation and Dismissal

Any student who fails to comply with the criteria set forth above will have his/her case reviewed for possible probationary action. In the absence of compelling extenuating circumstances such a student will be placed on probation for failing to make satisfactory progress. The probationary quarter will be the quarter in residence that immediately follows the one in which the noncompliance
occurred. The student will be notified in writing of the reasons for his/her probationary status and the steps required in order to be removed from probation.

A student who fails to meet the criteria for satisfactory progress for two consecutive quarters in residence will be dismissed from the program. In addition, any student who accumulates three probationary quarters will also be dismissed from the program. Such action will be communicated in writing to the student at the last address on record and registration in Departmental courses will be canceled.

3.3 Academic Misconduct

Academic misconduct is any form of deception by which the student claims as his or her own work that which actually belongs to another, or otherwise gains unfair advantage over other students. This includes such actions as cheating on class assignments or examinations and plagiarism as well as more severe offenses, such as theft or alteration of other persons’ academic materials for purposes of acquiring academic credit or enhancing grades.

All incidents of alleged academic misconduct will be reported to the Office of Academic Affairs. Unless the incident has been resolved to the satisfaction of the student and instructor involved, the Dean’s Office will conduct a hearing. While the deposition of each case will be determined on the basis of its particular circumstances, the following general guidelines may be stated:

1. Any proven instance of cheating or plagiarism will at minimum result in disciplinary probation for the student(s) involved. Such probation will carry with it the possibility of dismissal if there is an additional proven act of academic misconduct on the part of the student(s).

2. If it is proven that a student has stolen or altered academic materials belonging to another student or an instructor for the purposes of achieving academic credit or enhancing a grade, the department will normally initiate proceedings to have the student dismissed from the University for at least one year and be permanently dropped as a CSE major.

In all considerations of academic misconduct, the department will carefully follow the letter and the spirit of the University’s Student Conduct Code. This code has been separately printed in pamphlet form and is available to any interested student at the College of Engineering’s Office of Academic Affairs, Room 371, Loew Hall. The Student Conduct Code is made a part of this document by reference.

Another form of misconduct is the misuse of computing facilities, for example, by obtaining improper access to another student’s account. Before students obtain accounts on departmental computers they sign an agreement stating that they will not misuse their accounts. Violations of this agreement may result in the loss of computer accounts and/or being dropped from the major.

Each student who accepts admission into the Computer Science degree program agrees to be bound by the provisions of this section.
DEPARTMENTAL APPLICATION DEADLINE DATES
Autumn Quarter - July 1     Spring Quarter - February 1
Quarter and year applying for:  ☐ Autumn    ☐ Spring      Year ____________
Have you Applied to this department before?  ☐ Yes  ☐ No  If yes, when?  Indicate Quarter/Year ____________

Legal Name  
☐ Last)  (First)  (Middle)  UW Student ID # __________________
Permanent Address  
(Street and Number)  City  State  Zip
Notification Address  
(If different)  (Street and Number)  City  State  Zip
Phone(s):  (H) ___________________  (W) ___________________

OPTIONAL

Ethnicity:  ☐ American Indian or Alaska Native  ☐ Asian or Pacific Islander  ☐ African American  ☐ Caucasian  ☐ Hispanic  ☐ Other: ___________________
Gender:  ☐ Female  ☐ Male
Residency:  ☐ Yes  ☐ No
Have you been coded by UW as a resident of Washington State?

MINIMUM ADMISSION REQUIREMENTS

1. Have you earned at least 45 quarter credits applicable towards a BS degree?  ☐ Yes  ☐ No  ☐ NC
2. Have you completed the following courses (or equivalent)?  (If you repeated a course, you must indicate both grades):  
   Course  Grade  Qtr/Year Taken  School
   Math 124
   Math 125
   Math 126
   Phys 121
   CSE/ENGR 142
   CSE 143

3. Overall GPA:  ____________________  (average of all transfer units and all UW units combined)
4. Are you a current UW student?  ☐ Yes  ☐ No  ☐ No OR Have you submitted an application to UW Admissions Office, 320 Schmitz Hall?  ☐ Yes  ☐ No  ☐ No

Applicant Signature  ____________________  Date  ____________________
3.4 Approved Senior Electives

CSE 449, 448 Introductory Communication Theory (4,3)
EE 476 Computer Aided Design of Digital Systems (3)
Geog 460 GIS Analysis (5)
Geog 463 GIS Workshop (3,5)
Info Sys 465 Analytical Cartography (5)
Ling/Anth 460, 461 Systems Analysis and Design I,II (4,4)
Ling/Phil 479 Formal Semantics and Natural Language (3)
Math 402, 403, 404 Introduction to Modern Algebra (3,3,3)
Math 407 Linear Optimization (3)
Math 408 Nonlinear Optimization (3)
Math 409 Discrete Optimization (3)
Math 414, 415 Number Theory (3,3)
Math 461, 462 Combinatorial Theory (3,3)
Math 464, 465, 466 Numerical Analysis I, II, III (4, 4, 4)
Mech E 477 Microcomputers in Mechanical Systems (4)
Music 456 Musical Applications of Digital Processing (3)
Physics 434 Appl. of Computers to Physical Meas. (3)
Quan Sci 381 Introduction to Probability & Statistics (5)
Statistics 341, 342 Introduction to Prob & Stat inference I,II (4,4)
Statistics 390 Prob & Stat in Engr & Science (4)
Statistics 421 Intro to Applied Stat & Experimental Design (4)
Stat/Econ 481 Intro. to Math, Statistics (5)
Stat/Math 394, 395 Probability I,II (3,3)
Stat/Math 491, 492 Introduction to Stochastic Processes (3,3)
TC 401 Style in Scientific and Technical Writing (3)
TC 407 Computer Documentation (3)
TC 408 Special Documents: Proposals, EIS, and Manuals (3)
TC 409 Writing for Publication (3)

3.5 Graduation Application/Exit Interview

At least two quarters before completing the requirements for your BS degree, you should make an appointment with the Undergraduate Staff Advisor to apply for graduation. Prior to graduation, 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.

3.6 Co-op Program

The Co-op Office in 335 Loew Hall coordinates co-op jobs and minority student internships. This program gives students the opportunity to investigate the practice of the profession and define their own career goals, by spending one or more quarters in a job in the computer industry while enrolled in the degree program. A co-op job is an opportunity to put academic knowledge to use in a way that helps students to see the relationships between various parts of the curriculum and add to that knowledge in important ways. Students gain a wide variety of skills and abilities related to working in an engineering team setting — a setting that is very difficult to emulate in a university. Many Computer Science students have found co-ops to be a valuable part of their studies, and majors are strongly encouraged to take advantage of the opportunities that co-ops present for professional and personal development.

Students going on Co-op should notify the Undergraduate Staff Advisor and plan a tentative schedule for their returning quarter. The quarters they spend off campus are exceptions to the Department’s Contimation Policy.

Up to 4 credits of ENGR 321 (Engineering Cooperative Education) can be used as free electives, in other words as credits toward the 180 total credits required for the Computer Science degree.

3.7 Center for Career Services

The University’s Center for Career Services, 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 Science and other disciplines.

3.8 Computer Science Honors Program

The Honors Program was established in 1977 for undergraduates in the College of Arts and 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 well-written document, clearly presented.
4 Computer Science and Engineering Courses for Pre-Majors and Non-Majors

CSE 135 Computational Techniques (4)
Principles of computation and techniques for using the computer as a tool in science and engineering. Basic programming techniques (conditionals, iteration, procedural, and data abstraction). Introduction to simulation, visualization, and symbolic mathematics. Solving engineering and science problems using programmable tools rather than conventional programming languages. Prerequisites: Math 124. Note: this course is not a requirement for either the Computer Science or the Computer Engineering degree.

CSE/ENGR 142 Computer Programming for Engineers and Scientists I (4)
Basic programming-in-the-small abilities and concepts. Highlights include procedural and functional abstraction with simple built-in data type manipulation. Basic abilities of writing, executing, and debugging programs. Not available for credit to students who have completed CSE 210 or ENGR 141.

CSE 143 Computer Programming for Engineers and Scientists II (5)
Continuation of CSE/ENGR 142. Concepts of modularity and encapsulation, focusing on modules and abstract data types. Covers some basic data structures. Not available for credit to students who have completed CSE 211. Prerequisite: CSE 142.

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 143. No credit if CSE 326, CSE 374, or EE 374 has been taken.

CSE 410 Computer Systems (3)
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: CSE 373. No credit if CSE 378 or CSE 451 has been taken.

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 if CSE 341 or CSE 401 has been taken.

CSE 415 Introduction to Artificial Intelligence (5)
Principles and programming techniques of artificial intelligence: LISP, symbol manipulation knowledge representation, logical and probabilistic reasoning, learning, language understanding, vision, expert systems, and social issues. Prerequisites: CSE 373, or CSE 143 and permission of instructor. No credit if CSE 473 has been taken.

Note: CSE 373, 410, 413, and 415 are intended for non-majors only. Prospective Computer Science or Computer Engineering majors should not take these courses. Instead, they should wait until they are admitted to the major and take the majors courses listed in Section 5.

5 Computer Science and Engineering Courses for Majors

The curriculum is subject to change, since it is regularly updated to reflect a rapidly changing field.

CSE 321 Discrete Structures (4)
Fundamentals of set theory, graph theory, enumeration, and algebraic structures, with applications in computing. Prerequisite: Math 126 and CSE 143.

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)

CSE 341 Programming Languages (5)
Designed to make the student reasonably fluent in several radically different languages, such as Lisp, Prolog, and Smalltalk. Prerequisite: CSE 143. No credit if CSE 413 has been taken.

CSE 370 Introduction to Digital Design (3)
Introductory course in digital logic and its specification and simulation. Boolean algebra, combinational circuits including arithmetic circuits and regular structures, sequential circuits including finite-state-machines, use of programmable logic devices. Simulation and high-level specification techniques are emphasized.

CSE 378 Machine Organization and Assembly Language (4)
Difference and similarities in machine organization; central processors; fundamentals of machine
language and addressing; assembly language programming, including macros; operating system interfaces. Prerequisite: CSE 143. No credit if CSE 410 has been taken.

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 if CSE 413 has been taken.

CSE 403 Software Engineering (4)
This course presents the fundamentals of software engineering using a group project as the basic vehicle. Topics covered include the software crisis, managing complexity, requirements specification, architectural and detailed design, testing and analysis, software process, and tools and environments. Prerequisites: CSE 321, CSE 341, CSE 378. Either CSE 401 or CSE 451 is recommended.

CSE 421 Introduction to 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 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 CSE 410 or EE 474 has been taken.

CSE 457 Computer Graphics (3)
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 325.

CSE 461 Introduction to Computer-Communication Networks (3)

CSE 467 Advanced Digital Design (3)
Advanced techniques in the design of digital systems. Hardware description languages, combinational and sequential logic synthesis and optimization methods, partitioning, mapping to regular structures. Emphasis on reconfigurable logic as an implementation medium. Memory system design. Digital communication including serial/parallel and synchronous/asynchronous methods. Prerequisites: CSE 370 and CSE 326.

CSE 471 Computer Design and Organization (4)
CPU instruction addressing modes, CPU structure and functions, computer arithmetic and logic units, register transfer level design, hardware and microprogram control, memory hierarchy design and organization, I/O and system components interconnection. Laboratory project involves design and simulation of an instruction set processor. Prerequisite: CSE 370 and CSE 378.

CSE/Ling 472 Introduction to Computational Linguistics (3)
Introduction to computer applications of linguistic theory, including syntactic processing, semantic, and pragmatic interpretation and natural language generation. Prerequisite: Ling 471, or permission of instructor.

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 326, CSE 341 recommended. No credit if CSE 415 has been taken.

CSE 477 Digital System Design (4)
Students use the laboratory to design, simulate, construct, and debug a substantial project that includes hardware, software, and communication components. Lectures focus on use of embedded processors in digital system design and interfacing techniques. Writing and debugging of real-time reactive software emphasized. Prerequisites: 378 and 467.

CSE 490 Special Topics in Computer Science & Engineering (1–5)
Lectures and discussions on topics of current interest in computer science and engineering not covered by other CSE undergraduate courses.

CSE 498, 498H Senior Project (1–6) 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 and engineering 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 (1–24)
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 and Engineering faculty member. Useable 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.

Note: The department offers a series of courses for premajors and majors in other departments, including both introductory and advanced courses (see Section 4). In unusual circumstances, non-majors may petition for admission to a CSE majors course. Please request a form from one of the
staff undergraduate advisors. Forms must be submitted by the week before the quarter begins.

6 Computer Science and Engineering Faculty


Theodore H. Kehl, Professor, Professor of Physiology and Biophysics (1961). B.S., 1956, M.S., 1958, Ph.D., 1961, Wisconsin. Real-time hardware and software systems, computer design, VLSI.


Paul Young, Professor (1982). (On leave from the National Science Foundation.) B.S., 1959, Antioch; Ph.D., 1963, MIT. Computational complexity, computability, and connections with mathematical logic.

COMPUTER SCIENCE
HANDBOOK
FOR UNDERGRADUATES
1995-1996

Department of Computer Science and Engineering
University of Washington
114 Sieg Hall
Box 352350
Seattle, Washington 98195-2350
(206) 543-1695
email: ugrad-CS-advisor@cs.washington.edu

The University of Washington provides equal opportunity in education without regard to race, 
color, creed, religion, national origin, sex, sexual orientation, age, marital status, disability, or 
status as a disabled veteran or Vietnam era veteran in accordance with University of 
Washington policy and applicable federal and state statutes and regulations.