This is the CSAB self-study document for the **Computer Science & Engineering Program** *in the* **Computer and Information Science Department** *at the* **Ohio State University**. *It is in the format specified by CSAB.*

A couple of important points:

- *Not all the information required by CSAB has been included. For instance, information about faculty workloads etc. have been omitted.*
- *There are some differences between the program as it is today and the information provided in this self-study. The main differences are with respect to the assessment procedures. The set of assessment procedures described here is more than we have in place today, but is what we plan to implement later this year and early next year.*

VOLUME II

QUESTIONNAIRE FOR REVIEW
of the
COMPUTER SCIENCE PROGRAM

submitted by

The Ohio State University
Institution

May 4, 1988
Date

to the

Computer Science Accreditation Commission

for the
1998-1999
accreditation cycle

Primary contact: Stu ZWEBEN
Telephone number: (614)292 5973 **FAX:** (614)292 2911
Electronic mail: `zweben@cis.ohio-state.edu`

VOLUME II
THE PROGRAM

VIII. OBJECTIVES AND SELF-APPRAISAL

A. Objectives of the program under evaluation. Provide a statement of objectives for the program under evaluation.

The objectives of the *Computer Science & Engineering* program are as follows:

1. To provide graduates with a thorough grounding in the key principles and practices of computing, and in the basic engineering, mathematical, and scientific principles that underpin them.
2. To provide graduates with an understanding of additional engineering principles, and the mathematical and scientific principles that underpin them.
3. To provide graduates with an understanding of the overall human context in which engineering and computing activities take place.
4. To prepare graduates for both immediate employment in the CSE profession and for admission to graduate programs in computing.

B. Changes in philosophy and patterns since the last visit if the program was previously evaluated. In what respect, if at all, has the pattern and philosophy of computer science education changed at your institution during the last 5 years?

The program has not been previously evaluated.

One important change in the program during the last few years has been to integrate *software engineering* issues into our beginning courses, *CIS 221, CIS 222*, and *CIS 321*. This is not so much a fundamental change in our philosophy, as a natural evolution of the approach we have taken in our program. Although many of these issues had been addressed in these courses we felt, since many of our students came into the program with a background in programming, that additional topics, in particular the *component-based* approach to developing software systems, could be presented and that the discussion in the three courses could be much more closely integrated. The pilot versions of the new *CIS 221, CIS 222,* and *CIS 321* have been completed successfully, and full implementation is under way. We have also taken this opportunity to move from *Modula-2* to a version of *C++* as the main student programming language; *RESOLVE/C++*, the discipline in *C++* that we teach, strongly encourages students to design their systems with correctness issues firmly in mind from the start rather hacking them together. We expect the changes in these courses to ripple through the rest of the curriculum over the next 2 to 3 years as students with the new sequence progress through the program.

At the more advanced levels too the changes have been evolutionary. Thus we have introduced several new courses in networking partly because of the growing importance of that area and partly because of our tremendously improved faculty strength in the area. Another related course that we are considering would deal with computer security issues.

C. Major developments in the past 5 years. Summarize major developments and/or progress made in connection with the program in the past 5 years.

The following are some of the major developments in the last four or five years:

- The Department moved into a brand new building about four years ago. The building has been designed to provide a good working environment for faculty and staff, provide some classrooms wired to support workstations for each student, provide a comfortable environment for undergraduate students to interact with faculty and teaching assistants, and provide room for the growing research needs of the Department.
- About three years ago the computing equipment in the various student computing labs as well as in faculty offices was replaced by a new generation of equipment. Plans are currently under way to go through yet another round of upgrading of the computing equipment.
- The first three courses in the program, *CIS 221, 222,* and *321* have been revised substantially to stress *software engineering* principles. While the previous version of these courses did pay considerable attention to software engineering issues, they were not fully integrated with each other. The new version of these courses is tightly integrated and enables the student to develop the ability to design robust software systems. We have also taken this opportunity to move these courses from *Modula-2* to a disciplined version of *C++*, a version that encourages students to pay attention to designing reliable systems rather than just hacking together something that 'works'.
- Several new capstone design courses have been introduced to allow students to engage in a quarter-long design exercise in a field that they find most interesting.
- Our hiring of senior and junior faculty in computer networking has enabled us to introduce several new (elective) courses in this important area.
- We have hired new faculty both junior and senior not just in networking but in several important areas. This will enable us to offer the best possible and most up-to-date courses and research opportunities to our students.

D. Previous evaluation.

1. *If the program has been evaluated previously by CSAC, indicate how each of the deficiencies and concerns listed in the most recent Final Statement has been addressed.*

   Not applicable.

2. *What significant changes have occurred with respect to curriculum, faculty, and computing facilities since the most recent evaluation visit? Describe when the changes were implemented and what impact they have had on the program.*

   Not applicable.

E. Plans for future development. Indicate any significant plan for future development of the programs of instruction and research in the unit offering the computer science programs.

We plan to continue building our faculty strength. Recently the *Ohio Board of Regents* provided us support for hiring a *Distinguished Professor* (in any suitable area of computing), and we are actively searching for an appropriate candidate for this position. We believe this position, as well as the other more junior faculty we plan to hire in the coming years, will add considerably to our already strong research program. Since faculty typically teach both graduate and undergraduate courses, these new faculty also strengthen our undergraduate program.

Further, we plan to encourage more undergraduate participation in our research activities. The recent changes in the beginning courses have already helped with this.

F. Strengths of the unit offering the computer science program.

The primary strength of the *Computer and Information Science Department* is without a doubt its faculty. Several of our faculty, in a range of areas from Artificial Intelligence to Software Engineering to Computer Graphics to Networking, and Parallel and Distributed Computing, are highly recognized for the important contributions they have made. Many of the faculty have won prestigious research awards for their contributions. Many of the faculty are also excellent teachers and enjoy teaching undergraduate courses. One of our faculty has won the prestigious University Alumini Award for Distinguished Teaching. SEveral have won *NSF Career Development Awards*, which recorgnize excellence in both teaching and research. Graduate teaching associates who teach some of our beginning courses are also excellent teachers and generally receive very high ratings from their students.

The Department also has a very capable Academic Advising office that advises students before they have faculty advisors appointed for them, and provides students any needed help as they progress through the curriculum. The Administrative staff of the Department is very competent and ensures the smooth running of the program.

The professional computing staff of the Department is also highly qualified and they ensure that students and faculty have access to, and are able to fully exploit, the most up-to-date software and hardware products available. The University has also generally provided sufficient resources to make this possible.

G. Weaknesses or limitations of the institution or unit offering the computer science program.

One important weakness, perhaps the inevitable consequence of the large size of our program and of the University, is the general lack of close contact between undergraduate students and faculty. While there are notable exceptions, students tend not to approach their advisors or other faculty as frequently as we would like them to. We are attempting to alleviate this problem via, for instance, the recently instituted tradition of the annual Departmental banquet.

Another important problem, a somewhat recent one, is the growing demand for computing courses. This has appeared both in the form of increasing numbers of students, both graduate and undergraduate, wanting to major in the field, as well as non-majors interested in taking one or more of our courses. Our primary commitment is, of course, to our majors, and we have preferential registration mechanisms in place that ensure that our majors are able to take their courses in a timely manner. We would like to increase the number of sections of the most popular courses, but limited resources limit what we can do. We have an "enrollment management" policy that has recently been reinvoked and which limits the number of students who are accepted int the Computer Science major each year.

H. Short range goals. What do you consider the principal short range goals of the unit offering the computer science program? To what degree are these goals being satisfied?

Our main short range goals are to ensure that the changes to *CIS221, 222, 321* work smoothly and to make necessary adjustments to later courses to take advantage of these changes. We also plan continue to exploring other possible course offerings (such as a course on computer security).

We are currently in the process of planning for an update to the computers in the student to a new generation.

I. Long-range goals. What do you consider the principal long-range goals of the unit offering the computer science program? To what degree are these goals being satisfied?

Our major long term goal is to continue to improve faculty research strength and visibility. Attracting undergrads to research is a key part of this plan.

J. Support services. Discuss the adequacy of support services, specifically library and computer services and facilities.

(a) *Library*

The *Science and Engineering Library* is an excellent library; the collection is not exhaustive but *Ohio Link* makes it possible to borrow from any library in the system; and on-line catalogs make it easy to find titles.

(b) *Computer Services*

We are very proud of our state-of-the-art computing facilities. Students have access to the best technology avaialble, both software and hardware – more details need to be provided.

(c) *Other - specify*

None.

IX. INFORMATION REGARDING PROGRAM ADMINISTRATORS

This information has not yet been compiled in the required format.

X. INFORMATION REGARDING FACULTY MEMBERS.

This information has not yet been compiled in the required format.

Faculty Summary Information

This information has not yet been compiled in the required format but answers to some questions are included below:

Please answer the following questions regarding the faculty associated with the undergraduate program(s) being evaluated. Each question relates directly to the CSAC/CSAB criteria for faculty in the unit offering the program.

1. *How do you assure that courses in the curriculum are offered in such a manner as to permit students to complete the program in a timely manner? Please give as many specifics as possible.*

   Each required course is generally offered more than once a year; several sections of the beginning courses as well as many of the advances courses are offered each quarter. A handful of courses are offered only once a year. We use a priority registration systems that gives preference to our majors to ensure that they are able to take all their courses in a timely manner.

   The Discrete Mathematics course (Math 366) is taught by the Mathematics department but it is taken only by our students. As a result, they are able to take this course in a timely manner.

   Our students are also required to take a number of other courses offered in other departments, such as Physics 131, 132, etc; all these courses are offered at least once a year, and usually several times a year; hence our sutdents have no difficulties in enrolling in these courses when they need to.

   Some of our students choose to work full time during a portion of the year. This would, of course, delay their graduation.

2. *Are all of the regular, full-time faculty members in the department with primary commitment to the program able to teach a broad range of undergraduate courses in the program and able to contribute to the discipline?*: Yes.

   Please provide an explanation of your answer: All of the regular, full-time faculty members in our department either have a Ph.D. degree in Computing or a related field, or have considerable teaching and research experience in the field. As a result, they are fully qualified to teach a range of undergraduate courses, and able to contribute to the field.

3. *For each course offered by your department that is a required or elective course in the program list the name of the course coordinator. Please describe the way in which course coordinators are assigned and what their duties are:*

   List of course coordinators will be available at the time of the evaluation visit.

   The course coordinators are appointed by the chair of the Department on an annual basis. The course coordinator is chosen from among those faculty who regularly teach the particular course, and generally rotates among them.

   For high level courses, the coordinator's job is to ensure that appropriate text books are ordered, and to ensure informally, often via e-mail, with other faculty teaching the course. These interactions occasionally lead to ideas for changes in courses.

   For the 200 level courses, many sections of which are taught by Graduate Teaching Associates, the course coordinator has a much greater responsibility (and correspondingly gets credit, often in the form of a slight reduction in his teaching load); he orders the text books and class notes, meets regularly with the GTAs teaching the course, often sets the exams, etc.

XI. CURRICULUM

A. *Title of degree for program under review, as specified on the transcript and diploma.*

   Transcript: *Bachelor of Science in Computer Science and Engineering*

   Diploma: *Bachelor of Science in Computer Science and Engineering*

B. *Credit hour definition. One semester hour normally represents one class hour or three laboratory hours per week. One academic year normally represents at least 28 weeks of classes exclusive of final examinations. Any differences should be noted below.*

   The Ohio State University uses a quarter system. A quarter is ten weeks long, not counting final examinations; so an academic year is 30 weeks of classes exclusive of final examinations. A credit hour represents one class hour or three lab hours per week. To convert to semester hours, quarter hours were multiplied by 2/3.

C. *Prerequisite flowchart.*

   Attach a flowchart showing the prerequisite structure for computer science courses:

   Flowchart is attached (*Appendix 1*).

Required and elective courses. In the table on the following pages, the courses should be listed in the order in which they are normally taken in the curriculum, and they should be classified in the appropriate categories. The data should clearly indicate how the program satisfies CSAC/CSAB curricular criteria as prescribed in the current issue of Criteria for Accrediting Programs in Computer Science in the United States.

Required Courses:

The required courses should be listed by department abbreviation (i.e., Math., Chem., C.S.), course number, title, and number of credits. The credits for each course should be listed under the appropriate category.

Elective Courses:

Elective courses should be designated as "elective". If an elective is restricted to courses in one particular category, the credits should be tabulated in the appropriate column, and the listing should indicate the category - for example: Elective - science. IN ADDITION, supplemental information should be provided to indicate how the institution ensures that the elective is chosen from courses in the category. If an elective is not restricted to a particular category, the credits should be listed under Other. Use footnotes to elaborate on any listings that require additional explanation.

Note: Individual courses can be split between curriculum areas if material content of the course justifies the split. For example, a discrete mathematics course may qualify partly as Mathematics and partly as Computer Science based upon its material content.

Course credit assignments to the categories should be listed in multiples of one-half credit.

D. *Course requirements of curriculum (year by year and term by term)*

Two tables are appended to this self-study; *Table 1* shows the course requirements of the program; *Table 2* shows the order in which students normally take the courses.

E. *The Computer Science Curriculum.*

Please describe the computer science curriculum for the program being reviewed. Please list the course number, course title, and the number of credits for each required and elective course. If electives may be selected from one or more groups of courses, provide the rationale for the composition of the group(s). You may provide a narrative to help the reader understand the curriculum more clearly or to show how the curriculum satisfies CSAC/CSAB criteria.

Core Courses:

Course number	**Course Title**	**Credit hours**
CIS 221	Software Development Using Components	4
CIS 222	Development of Software Components	4
CIS 321	Design & Analysis of Component based software	4
EE 261	Logic Design	3
EE 206	Switching Circuits Lab	1
CIS 360	Introduction to Computer Systems	4
CIS 459.XX	Various Prog. Lang.	1
CIS 570	File Des. & Analy.	3
	Total (Quarter Hour) Credits:	24

These courses provide a fairly uniform coverage of algorithms, data structures, software design, concepts of programming languages, and computer organization and architecture. The table below describes (approximately) how many credit hours of each of the courses covers each area:

Course	Alg.	Data Str.	Software Des.	Prog Lang	Org/ Arch
CIS 221	1		1	2	
CIS 222	1	1	2		
CIS 321	1	2	1		
EE 261					3
EE 206					1
CIS 360			1	1	2
CIS 459.XX				1	
CIS 570	1	2			
Total	4	5	5	4	6

Theoretical foundations, as well as analysis and design are stressed in every one of the above courses with the exception of CIS 459 and EE 261. Substantial lab work, in the form of software design and development, occurs in all of the above courses except EE 261, EE 206, and CIS 459. Switching circuit design occurs in EE 206. CIS 459 involves software development but the focus in this course is on learning a new (typically second) programming language. Students become proficient in a disciplined style with *C*++, and learn to program in assembly language in CIS 360. They learn a second high level language in CIS 459. In most of these courses students are required to use modern software tools such as e-mail and web browsers; and they also become proficient in using the facilities provided by the operating system (UNIX at the moment).

Advanced Courses:

Course number	Course Title	Credit hours
CIS 541	Elementary Numerical Methods	3
CIS 560	Systems Software Des., Development, & Doc.	5
CIS 625	Introduction to Automata and Formal Languages	3
CIS 655	Introduction to the Principles of Prog. Lang.	4
CIS 660	Introduction to Operating Systems	3
CIS 675.01 or EE 662	Introduction to Computer Architecture Theory and Design of Digital Computers	3
CIS 680	Data Structures	3
	Total (Quarter Hour) Credits:	24

Each of these courses builds on the core and has one or more of the core courses as a prerequisite. Several advanced areas, including numerical computation (CIS 541), software methodology and engineering (CIS 560), programming languages (CIS 655), operating systems (CIS 660), computer architecture (CIS 675 or EE 662), and algorithms and data structures (CIS 680) are covered.

Other Required Courses:

Course number	Course Title	Credit hours
CIS 601	Social and Ethical Issues in Cptg.	1
Math 366	Discrete Mathematics	3
CIS XXX or EE 769	Capstone Design Course; One of: CIS 731, CIS 758, CIS 762, CIS 772, CIS 776, CIS 778, or EE 769	3 or 4
	Total (Quarter Hour) Credits:	7 or 8

CIS 601 deals with social and ethical issues in computing. Math 366 deals with discrete mathematics. The last course in the table above is a capstone design course; based on their interests choose one of the following courses: CIS 731, Knowledge Based Systems; CIS 758, Software Engineering Project; CIS 762, Advanced Operating Systems Lab; CIS 772, Information Systems Project; CIS 776, Hardware/Software Interface Design Project; CIS 778, Computer Aided Design of VLSI Circuits; EE 769, Microprocessor Lab; the design course is either three or four credits.

In addition to these courses, all of which are required, students must also take several elective courses. A minimum of 18 credit hours of technical electives is required at least 9 of which must be computing courses, the rest being either computing courses or courses from a related field. These courses tend to be in areas such as networking, computer graphics, AI, etc.

F. *Computer Science*.

1. *The program must have at least one and one-third years of computer science topics*. The minimum number of computer science credits required of all students in the program is *64 quarter hours (approximately 43 semester hours).*
2. *The computer science segment of the curriculum must have a broad-based core consisting of the equivalent of 16 semester hours.* The minimum number of required hours in the program's core is *24 quarter hours (16 semester hours).*
3. *Core computer science.*
   a. *Basic coverage of algorithms, data structures, software design, concepts of programming languages, and computer organization and architecture. List the core courses that provide significant coverage in each of these areas*:
   CIS 221 provides coverage of algorithms and programming languages; both CIS 222 and CIS 321 provide coverage of algorithms, data structures, and software design; EE 261, EE 206, and CIS 360 provide coverage of computer organization and architecture; CIS 360 also provides limited coverage of software design and some coverage of programming languages (assembly language). CIS 459 provides coverage of languages. CIS 570 provides coverage of algorithms and data structures.
   b. *Theoretical foundations. List the required core computer science courses in which there is a significant (the equivalent of at least 3 hours of lecture) theoretical foundations component and briefly describe the material covered*:
   CIS 221, CIS 222, and CIS 321 provide considerable coverage of theoretical foundations. Material covered includes the ideas of formal specification of program behavior, algorithm analysis techniques, and use of formal notations such as regular expressions.
   c. *Analysis and design experiences. List the core computer science courses in which there are significant analysis and design experiences, and briefly describe those experiences*:
   CIS 221, CIS 222, CIS 321, CIS 360, EE 206, and CIS 570 all provide significant analysis and design experiences. The first three courses form an integrated sequence in learning to design software using a component-based approach; in CIS 221 students start by designing programs to meet given requirements using pre-existing components; in CIS 222, they design implementations of components to meet given specifications, and consider issues of performance; in CIS 321 they continue these activities, designing larger pieces of software and also using other more formal notations such as regular expressions.
   CIS 360 requires students to design small pieces of software and program them in assembly language. They are expected to continue the good design habits learned

in the earlier courses. CIS 570 requires the students to design solutions to data base related problems, and analyze the behavior of their programs; here they are required again to use predicate calculus notations.
EE 206 teaches students to design simple digital circuits.

d. *Programming languages and systems.*

   1. *What programming languages and computing systems are used in required courses in the program?*
      Students program in a disciplined style in *C++* in most of the courses. In CIS 360 they program in *SPARC* assembly language; in this class they use a *SPARC* simulator. The operating system that students work on is UNIX. In addition they use a variety of e-mail systems, web browsers, newsgroups, etc.

   2. *In which programming language(s) do students in the program become proficient?*
      C++. They also learn a second high level langauge in 459.XX.

4. *Advanced computer science.*

   Advanced courses typically have, as a prerequisite, one or more of the core courses. The computer science curriculum must include the equivalent of at least 16 semester credits of advanced course work. The minimum number of advanced credits required in the program being reviewed is 24 quarter credits (16 semester credits).

   As noted before, the 24 quarter credits is only for the required courses in the program; in addition, students are required to take a least 18 credit hours of tech electives, at least 9 of which must be advanced computing courses; and students are required to take a capstone design course (3 or 4 hours) from a specified list of computing courses.

   a. *Advanced portion of the curriculum. Please list the courses that are required in the advanced portion of the curriculum. If students are required to select courses from specific lists or groups, please provide the rationale for the composition of the lists or groups and indicate how such selection is handled and monitored to assure compliance of all students.*
      The list of advanced courses taken by all students is: CIS 541, Numerical Methods (3 credits); CIS 560, System Software Design, Development, and Documentation (5 credits); CIS 625, Introduction to Automata and Formal Languages (3 credits); CIS 655, Introduction to the Principles of Programming Languages (4 credits); CIS 660, Introduction to Operating Systems (3 credits); CIS 675, Introduction to Computer Architecture, or EE 662, Theory and Design of Digital Computers (3 credits); and CIS 680, Data Structures (3 credits).

   b. *Depth of knowledge built on the core through the equivalent of at least one course in each of 3 advanced areas. Please indicate how the requirement for depth in 3 separate advanced areas is satisfied.*
      CIS 655, CIS 660, CIS 680 (among others) meet this requirement.

   c. *Electives in the program. If elective courses are used to satisfy either core or advanced computer science requirements, describe the advising and monitoring system that ensures that each and every student's program of study satisfies the*

requirements. Attach examples of documents used to inform students of the requirements and/or to verify that they have satisfied those requirements.
Electives are not used to satisfy either core or advanced CS requirements.

G. *Mathematics:*

Summarize how the curriculum satisfies the requirements (CSAB/CSAC) for each of the following.

1. *The curriculum must include at least one-half year (at least 15 semester hours) of mathematics. The minimum number of hours required by students in the program is* 33 quarter hours (22 semester hours). This includes 6 quarter hours of Statistics.
2. *List the course(s) that provide coverage of each of the following. If the material in an area is not covered in a course of its own, please describe the extent of the coverage of material in the listed courses.*
   (a) *Discrete mathematics*: This is covered by Math 366 (3 credits).
   (b) *Differential and Integral Calculus*: Covered by Math 151, 152, 153, 254, 415 (27 credits).
   (c) *Probability and Statistics*: Stat 427, 428 (6 credits).
3. *If elective credits are used to satisfy mathematics requirements, describe the advising and monitoring system that ensures that each and every student's program of study satisfies the requirements. Attach examples of documents used to inform students of the requirements and/or to verify that they have satisfied those requirements.*

   No electives are used to satisfy mathematics requirements.

H. *Science requirements.*

Summarize how the curriculum satisfies the CSAC/CSAB requirement for the equivalent of each of the following.

1. *The curriculum must include at least two-fifths (the equivalent of 12 semester hours) of a year of science. The minimum number of science credits required of all students in the program is*: 24 quarter hours (18 semester hours).
2. *List the courses that can be taken to provide a one-year (two-semester) sequence in laboratory science for science/engineering majors*: Physics 131, 132, 133 (15 quarter hours (12 semester hours)).
3. *List the courses that can be taken to provide the equivalent of two additional one-semester courses in science. Each of these courses must be a course in science or a course that enhances the student's abilities in application of the scientific method. Each of these courses should be a course for science/engineering majors or a course with a strong emphasis on quantitative methods. Mathematics and engineering courses do not normally satisfy this requirement. See the CSAC/CSAB Science Advisory for further information on this requirement*: Chemistry 121, 125 (9 quarter hours (6 semester hours)).

4. *If elective credits are used to satisfy science requirements, describe the advising and monitoring system that ensures that each and every student's program of study satisfies the requirements. Attach examples of documents used to inform students of the requirements and/or to verify that they have satisfied the requirements*: Electives are not used to satisfy science requirements.
5. *The minimum number of required hours of math and science combined is hours. CSAC/CSAB criteria require that this total be at least one year of study (equivalent to 30 semester hours) in math and science*: 57 quarter hours (38 semester hours).

I. *Humanities, social sciences, arts, and other disciplines. Humanities, social sciences, arts, and other disciplines; that serve to broaden the background of the student. Courses mentioned in one or more of the above categories typically may not be counted in this area.*

1. *The curriculum must contain at least one year of course work in these areas. The minimum number of credits in these areas is*: 45 quarter hours (30 semster hours). List the courses or groups of courses that all students in the program must take.

   Students must all take English 110 or 111, a Second Writing Course from an approved list of courses, 9 credits of Social Science Courses, 9 credits of Visual/Performing Arts, and 10 credits of Historical Survey.
2. *Summarize the distribution of the courses in this portion of the curriculum over the humanities, social sciences, arts, and other disciplines that serve to broaden the background of the student*: Students are required to take 9 credits of Social Science, 9 credits of Social Diversity (which may also be in another category such as the Second Writing course), and 10 hours of Historical Survey.

J. *Oral and written communications skills. Describe how this curriculum assures, for all of its graduates, the development (possibly through general education requirements) of oral and written communications skills, and describe how those skills are applied in the computer science portion of the curriculum.*

Students develop their communication skills in a variety of courses. In CIS 221, 222, and 321, they are required to work in teams to solve portions of problems to ensure that they are able to communicate with small groups. This is much more formalized in CIS 560 where students remain in the same team for the entire quarter, have specific responsibilities for design and documentation, and are required to orally present their work to their team and the instructor for the course in oral group presentations. CIS 601, the one credit course on social and ethical issues in computing requires students to make a 5 to 10 minute presentation to the entire class, or participate in a formal moderated (by the instructor) debate; in this class they are also required to evaluate the effectiveness of other students' oral presentations.

Written communication skills are developed in English 110 or 111, the second writing course. In CIS 560 students are required to write technical documentation, including user manuals to ensure that their writing skills are applied in ways that they are most likely to encounter in their professional life. CIS 601 also has a strong writing component; students are required to write two essays, defending a particular position on an ethical question involving computing.

K. Social and ethical implications of computer science.

1. *Describe how this program assures the development of an understanding of a broad range of issues regarding social and ethical implications of computing. List the required courses that provide coverage of these topics. For each course, describe the relevant topics covered.*

   Ethical issues facing computing professionals, such as the need to take responsibility for systems they design, as well as intellectual copyright issues with regard to software, are often mentioned in several of the required courses such as CIS 221, 222, etc. In addition, CIS 601 another required course is devoted to this topic, and it also explores legal issues having to do with the computing profession.

2. *Describe how the program assures that every section of the above-listed courses consistently covers the material described in part (a) above.*

   The topics are part of the official syllabii of the courses; course coordinators ensure that the topics are covered properly.

L. Course Outlines

Course outlines in the CSAB specified format are not yet completed. But the information in a somewhat different format (but only for some key courses) are attached to this self-study.

XII. COMPUTER FACILITIES

This section is not yet completed.