THE UNDERGRADUATE PROGRAM IN INFORMATION SYSTEMS DEVELOPMENT

J. D. Couger

Rapporteurs : Dr. N. M. Newman
Mr. J. F. Dunn
Mr. R. M. Simpson

Part 1 Concepts and Problems

In discussions with computer scientists in many European and American universities over the last three years, it has become clear that a wide gulf exists between them and their colleagues in business. To many computer scientists, there is no unified body of knowledge involved in business applications. Some feel business applications do not provide the challenge given by many of the problems in computing science. These presentations are intended to show that a valid discipline exists for the study of business applications.

Some of the challenges are illustrated by the marketing system, which is only one subsystem in the overall management system. Development of a marketing system requires sophisticated techniques for solution. The typical marketing system is comprised of 10 subsystems.

In the 1950s, only very large companies, those with annual sales exceeding $100 million, had the resources required for an integrated management information system. Because of the improvement in cost/performance of computers, it is possible for quite small companies, with annual sales of as little as $2 million dollars, to have an integrated management information system.

In the belief that systems concepts existed which could be applied to firms of all sizes, the University of Colorado conducted research on a number of small firms. In one, faculty and students assisted in the design of a data base which enabled the firm to capture a larger share of its market. Despite a period of recession, over the past two years the firm's annual sales increased from $2 million to $11 million.

The definition of a management information system is best accomplished by identifying three objectives for success. These objectives are:

1. To capture or generate all data pertinent to the firm's operations.

2. To process the data in the most efficient and economic manner, utilising the scientific disciplines to the greatest extent feasible.

3. To produce concise and timely information, as required by each level of management, for optimum execution of its functional objectives.

In the statement of the first objective, the key phrase is "all data pertinent". An example of a case in which this objective was not fulfilled occurred in a chemicals company. After ten years of providing a reasonably successful information system, the computing division decided to upgrade the system. The information needs were discussed with the senior managers and with managers further down the line. Finally, on the first day of the new system, the senior manager was presented with a sizeable tome; it took him a full week to digest half the information presented, and since the volume was to appear weekly, the final result was that he had to give up reading it altogether. He was getting less information than he had before. What the computer specialists had done was to give him all the data, instead of all the pertinent data. Another example occurred in an electronics firm, where a manager who had worked his way up in engineering was replaced by a man who had little knowledge of electronics. The reports which he received were unintelligible to him, because of the level of detail involved; but he was able to identify decision criteria for the analysts to use in redesigning his information system. The system provided only the exceptional circumstances he needed to know about, truly a "management by exception"system.

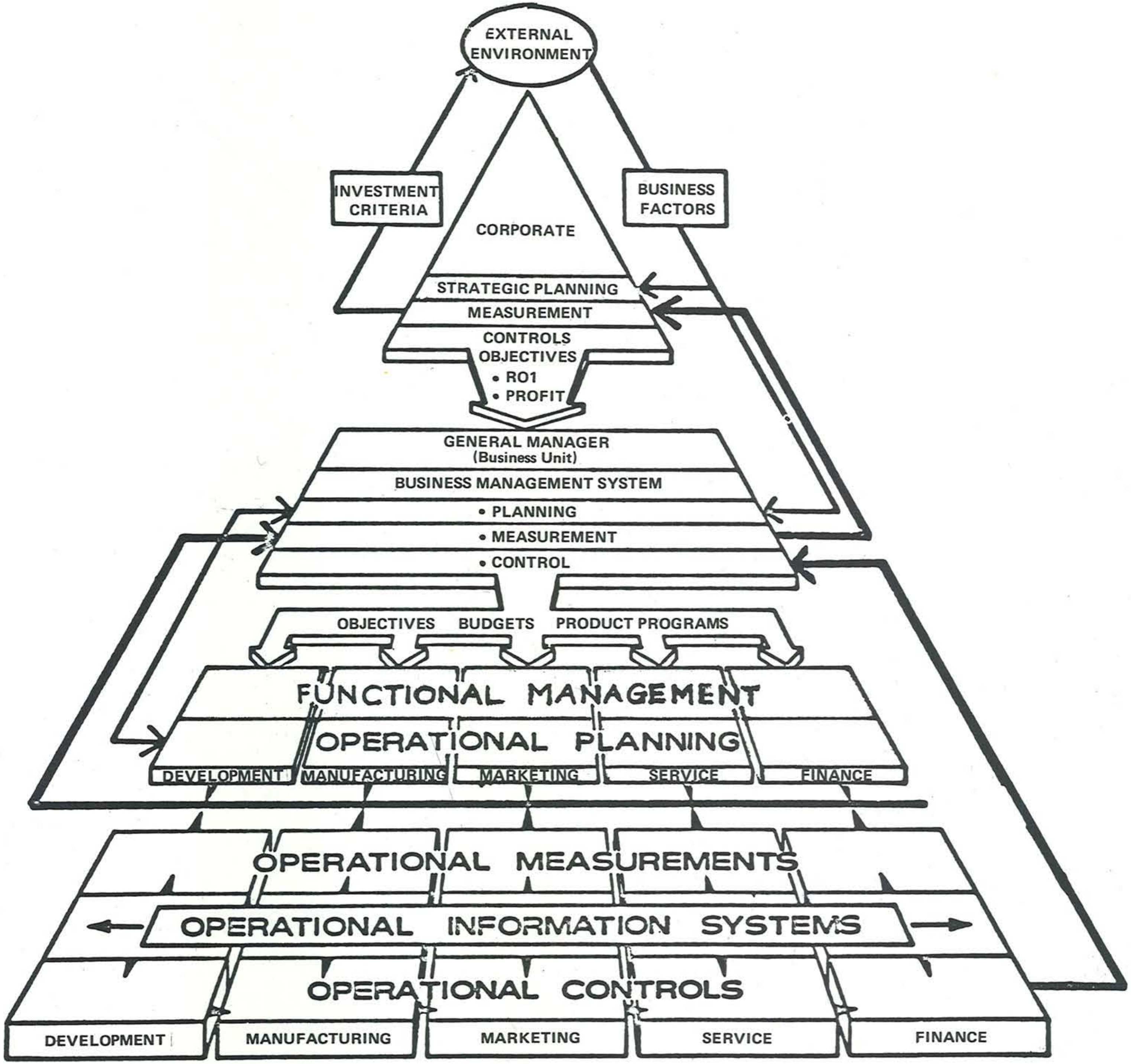
EXTERNAL ENVIRONMENT
INVESTMENT CRITERIA
BUSINESS FACTORS
CORPORATE
STRATEGIC PLANNING
MEASUREMENT
CONTROLS
OBJECTIVES
• RO1
• PROFIT
GENERAL MANAGER
(Business Unit)
BUSINESS MANAGEMENT SYSTEM
• PLANNING
• MEASUREMENT
• CONTROL
OBJECTIVES
BUDGETS
PRODUCT PROGRAMS
FUNCTIONAL MANAGEMENT
OPERATIONAL PLANNING
DEVELOPMENT
MANUFACTURING
MARKETING
SERVICE
FINANCE
OPERATIONAL MEASUREMENTS
OPERATIONAL INFORMATION SYSTEMS
OPERATIONAL CONTROLS
DEVELOPMENT
MANUFACTURING
MARKETING
SERVICE
FINANCE

Figure 1.

In the statement of the second objective, the key phrase is "the scientific disciplines", (or management sciences). In successful management information systems the management sciences are embedded in the systems to produce a cost/effective system.

The emphasis in the statement of the third objective is on providing information to "each level" of management. In one firm, a manager had to decide whether to use some spare capital to buy back some of the company's stock, or to invest in another venture. When he asked the computer specialists to provide this information, he was told it would take six months for a revision in the system to meet his request. Since the decision had to be made before the next board meeting in a month's time, the manager decided to take a hand himself. He taught himself programming, wrote a simple program to make the necessary calculations and comparisons, and had the results ready in time to make his decision with all the necessary information. In this case the information needed for his level of management was not available on the system.

Information must also be appropriate to the level of management to which it is being submitted. A highly profitable railroad has as its President a man of great experience on the railways; his decisions about the railway are based on daily operating reports of vast complexity and great detail. While he remains as President, a report at this level of detail meets the top manager's needs; but when he is succeeded, as is likely, by a man with less railway experience, the information will be quite inappropriate. His successor will, undoubtedly, request a revision to summarize information and provide it on an exception basis.

There are in fact three levels on which the management information system must operate: operational, tactical and strategic. A diagram of the way in which these levels are made up is shown as Figure 1. There is a strong tendency for information systems to operate only at the operational and tactical levels, ignoring strategic problems. But attention must be paid to factors outside the firm's control, such as the actions of Government, changes in the economy and so on. Inclusion of external data is a necessary ingredient for an integrated management information system.

The capabilities of management information systems should not, however, be exaggerated. Some, such as Professor Herbert Simon, have suggested that systems will in the near future be able to automate all management decision-making. This is certainly not in prospect in the immediate future. This is the era of improved quality and timeliness in management information.

A further characteristic of successful management information systems is the concentration on the area of greatest opportunity. Since by far the greatest cost of most products lies, not in the administrative structure of the firm, but in their production and distribution, the greatest potential for increased profitability lies in that area. Deeper levels of analysis are required. For example, the marketing systems include pricing policies, demand forecasts, competitor behaviour, and so on, in some depth, before interrelating the results. A common data base is needed to ensure that all systems are working on identical data. Priority systems are those which initiate action, such as the forecasting system.

Too often independent systems have been designed for interdependent activities; the emphasis needs to be on integration and on management needs at the tactical and strategic levels. The complexity of such systems necessitates development of new techniques, such as computer-aided systems analysis and design. A recent SRC report confirmed British efforts along these lines.

In conclusion, the increase in scope of the typical business application and the increase in complexity caused by designing systems at the operational, tactical and strategic levels require a new sophistication in systems analysis design. A body of concepts and techniques has been developed for this field which justify an academic program at both undergraduate and graduate levels.

Discussion

Professor Melkanoff asked about the problems Colorado had had in constructing their courses. Schools of management tend to be interested in applications, not in ideas; computer scientists tend not to be interested in business ideas.

Professor Couger replied that to overcome these problems it was essential to utilise all the academic resources of the University, and to integrate contributions from progressive people in several different departments.

Professor Melkanoff stressed that there were problems in bringing business people and computer specialists together.

Professor Couger suggested that the way was through the design of a broad but integrated curriculum. Business needs people with a mixed management-computing background.

Professor Page pointed out that in Britain there are very few schools of management, and that if a lead is to come in this area it will have to come from departments of computing science.

Professor Couger felt that the course of the initiative was irrelevant; these people were needed by business, and it was up to the universities to respond to the challenge as best they could.

Dr. Williams suggested that in the United States and Britain computing science departments tended to be heavily mathematics oriented, and that it was in the polytechnics that one found greater integration between management studies and computing. He felt that the two should come closer together in all academic institutions, and that industry should be more closely involved in project work.

Professor Ashenhurst thought that computer scientists were unaware of the complexity of management problems and of possible approaches to them. The main problem was to promote an interest in management problems among computer scientists.

Professor Melkanoff suggested that one area in which mutual interests already exist is that of data base systems and file management problems,and that this might provide an avenue for initial contact.

Professor Verrijn-Stuart commented that there is a deep seated conflict between academics and businessmen, on the question of fundamental research versus practical research, and that there is no tradition of combining the two in universities.

Part 2 Undergraduate Curriculum : Contents and References

This session is complimentary to the presentation by Professor Ashenhurst and will compare the proposed Undergraduate Curriculum in Information Systems Development with the published Graduate Curriculum [1]. In the Appendix is a copy of a paper [2] given at the 1972 National Conference of the ACM in Boston which is a preliminary report of the Committee responsible for the Undergraduate curriculum. This report discussed the philosophy behind the curriculum and the general content of each course. Their detailed content and the time allocated for each topic, possibly the more important aspect, are yet to be finalised but it is hoped to be able to consolidate the whole document at the 1972 Fall Joint Computer Conference, after having submitted a draft for comment to various organisations in business and government.

The structure of the course is shown in Fig. 2 of the appendix. It divides into two streams with some common courses in each stream. The Administrative Systems option sees Management Information Systems through the eyes of management and, in fact, it is hoped that these graduates will be influential in educating existing management; the Computer Science option - perhaps better called Computer Oriented option - takes a more technical, system viewpoint. When compared with the Graduate Curriculum some of the courses are combined (e.g. A1 with A4 to make A14) to produce courses more suitable in length and depth for the particular Undergraduate options.

The prerequisites are the same for both Undergraduates and Graduates. The Undergraduates are expected to fulfil these requirements during their Freshman and Sophomore (first and second) years. In addition, they take courses in the area where they plan to help design applications such as Accountancy, Business, Political Studies Hospital Administration etc.; see Figure 1 of the Appendix. The third and fourth years are taken up by either of the two Information Systems options, which themselves constitute 50% of the students' workload during these two years.

Discussion:

Professor Dijkstra asked what the arrows meant on Figure 2; it was not clear whether they showed prerequisites or time scales.

Dr. Hanani suggested that there should be two types of arrows, the one indicating prerequisites needed to follow the technical demands of the course and the second to show that familiarity in one course would give a more mature background for another.

Dr. Williams questioned whether the similarity of the Graduate and Undergraduate curricula implied a down-grading of what was normally expected of a Master's degree.

Professor Page stated that this was a problem facing most computer science departments with their own courses and should be left for discussion at another time.

Dr. Scoins asked for clarification of what the student workload would be.

Professor Couger replied that there were 45 contact hours per semester for each course, for which the student was expected to prepare for 2 hours for every hour in class. This compared with 3 hours preparation per class-hour for a Graduate level course.

Dr. Lauer asked what provision was made for a student entering the course in an irregular manner, e.g. from industry, where he might already be familiar with some of the early material.

Professor Couger replied that it depended upon the institution. At Colorado special courses using computer-aided instruction were put on to allow students to take selected topics rapidly, in their own time. Also, students can take tests in lieu of formal course work.

Professor Melkanoff said that it was common for special students to enroll early so that they could take summer courses.

Professor Page pointed out the dissimilarity with the UK system where, at the extreme, a student taking a Special Honours degree in Computing Science would spend, respectively, 33%, 85% and finally perhaps 95% on his speciality in his three years. The general degree was less extreme in this respect but regrettably found little favour with the better students.

1. Ashenhurst, T.L. (ed); A report of the ACM Curriculum Committee on Computer Education for Management. Comm. ACM 15, 5 (May 1972), 363-398.

2. Couger, J.D.; The Undergraduate Program in Information Systems Development. Proc. - National Conference of the ACM, Boston, August 1972. Reprinted in the Appendix.

Part 3 Implementation of the Curriculum

Now for the brutal facts about the Curriculum. Firstly, as Professor Page mentioned in his introduction, government and industry want people with education in computing and business. Who is to provide this education - universities, polytechnics, schools of business? Someone will provide it! In the USA, the universities have decided that it's a respectable academic discipline. Now, the universities accept the top 50% of high school graduates, while the state colleges accept any graduate. So, in general, the top students go to university, and the top universities are implementing the ACM Curriculum in information systems.

Secondly, business data processing is where the money is, and where the growth is. For example, Colorado University responds to the needs of local industry, and so local businesses provide money for the University. The University supplied only one third of the cash for building our School of Business Laboratory; industry provided the rest. The Organizational Behaviour section of the laboratory is a tiered case room with surrounding small group rooms for interpersonal studies. There are facilities for self instruction in concepts and principles in the Programmed Instruction Laboratory. The classroom is used merely to reinforce this learning. This means that there are few lectures; lecturers are able to spend contact hours with small groups discussing their problems. In the Statistics section of the laboratory, there are sophisticated calculating machines with memory, costing $7,000 - $9,000 each, and time sharing terminals connected to the main computer. The next laboratory houses the remote batch terminal. The needs of local industry are certainly met - many ex-students are now in middle management ranks, having come up via the systems analysis route, and are doing well.

Incidentally, the students at the laboratory have access to many packages for all the disciplines. Such programs are described in the publication, "Computing Newsletter for Instructors of Data Processing", now entering its sixth year. This publication, which has a circulation of 700 colleges and universities, aims at sharing information about curricula, textbooks, visual aids, and computer programs for research and instruction [1].

There are four major approaches used in the USA to give students practical experience:

(1) Sandwich programme. A student spends a term or só in industry, usually between his 3rd and 4th years. This is not, in fact, currently used in the USA.

(2) Work/Study programme. A candidate alternates regularly between university and industry - say each semester. Only a dozen U.S. schools use this approach.

(3) Project course. A candidate may receive six credit hours per week (one third of his total load that semester) for projects conducted. This is the most common U.S. approach.

(4) Project within course. This is the same, basically, but only about one third of a formal course is taken up with project development.

Note that (1) and (2) give no university credit; (3) and (4) do give some credit.

The work/study approach is used at the Northeastern University in Boston. Work and study alternate every other term, beginning after the second year. The candidate is employed by the firm in question and receives the same pay as other employees. He gets higher jobs progressively. For example, after his sophomore year, he may be a data control clerk or computer operator. By the time he receives his degree, he may be a junior systems analyst. Northeastern University has contacts with fifty major firms in New England.

Considering now the project approach, a number of different types of project are assigned. For example, Jim McKenney, in the School of Business at Harvard, has a design project whereby students co-operate with managers in planning situations, defining decision criteria and developing a model for planning, just as systems analysts do. He has a waiting list of companies! Jim Emery at the University of Pennsylvania had two projects last semester: one was to develop a cash flow analysis for the university - an extremely important task - and the other was to develop an accounting system for the Faculty Club. The latter project seemed mundane, but in fact was harder than the other; it was a beautiful example for behavioural study. Gary Dickson at the University of Minnesota has every MBA candidate given a project <u>in industry</u> at the end of his course, usually with others in a team. They developed, for example, a portfolio analysis system for an investment firm, and this scheme was subsequently adopted by the firm. Two other projects were a data base for billing for an electric company and an on-line enquiry system for a bank.

[Note the kind of universities giving these courses : Harvard, MIT, Stanford, Pennsylvania - the top class universities.]

However, it is surprising how long universities waited to liaise with industry when there is plenty of material for projects in the university itself. At the University of Colorado, where I am responsible for master planning, students participate in developing models for master planning and gain practical experience in that systems analysis task. They have also developed our business game, and co-ordinated the modules for that. (Programming projects are not considered in this discussion; these have existed for many years. We are referring to system design projects.) Another project was to develop a data base for a small firm, and this project, which was too large for one semester, is into its third year now. (The project was designed in a modular fashion.) A data base was developed for the student information system in the University - a very important application. As Dean of the School of Business, I had an MBA course-scheduler built as a project, and this saves me work! I learned yesterday that at Durham University students also do system projects within small firms. To start with, there was little enthusiasm at Colorado for working with small firms; it was felt that managers would be unsophisticated and projects shallow. But in a small firm one can see the whole of the firm's activities - the interaction of all the major modules - and one has the chance to interact with top management. Also, small firms are the ones which need help! - large firms are generally doing well enough as they are. With small firms, there are fewer problems: they have fewer worries over privacy of data, and they are a little more lenient with mistakes since their own employees tend to be of lesser calibre than in large firms. They make mistakes too; therefore, they are patient with our students. A feasibility analysis for a small firm was done at Colorado; that's what was done at Durham too.

Next, there is the question of control procedures for projects in industry. What should not be done, to avoid things going wrong? Here are six rules for control of projects:

(1) There must be formal Faculty administration of the project. Make sure that it is clearly understood who has responsibility for the project.

(2) Industry contact is made by Faculty, not students. Assure the company that things are under control.

(3) Project objectives are discussed and negotiated, with the students and with the company; then they are recorded.

(4) Written interim progress reports must be made by students.

(5) There is joint supervision by the Faculty and the firm. This makes a channel available to students in case they feel that they can't get relevant information.

(6) Formal reports are made by students: draft and written. Firstly, the report is given orally to the class to debug, and then to the firm's officers. Finally, a written copy is presented.

Proper control of any project is essential. The goodwill of firms must not be lost.

Dr. Andersen asked how many projects were going on at a time.

Professor Couger replied that all the students in a class would be doing projects together, although, of course, the projects need not all involve the same number of students. Each Faculty member would have 15 to 20 students to supervise.

Professor Dijkstra asked why so much emphasis was placed on practical work.

Professor Couger replied that it was to reinforce theoretical materials. The projects were more valuable than hypothetical projects. He thought Professor Parnas might have a different view.

Professor Parnas agreed. He had firm control of his projects, which he designed to give maximum benefit to students. He felt that, when working with industry, conflict of interests might arise.

Professor Couger pointed out that that was not a real life situation - the students might not see everything. He had on occasion rejected projects, and he preferred to start his students off on university projects to make sure that the students were capable of not botching a project and that companies would not take advantage of them.

Professor Melkanoff affirmed that the students received the right education and, of course, the Faculty got contacts as well.

Dr. Williams said he would like to support Professor Couger. At the Bendigo Institute, Victoria, they had been carrying out projects for five years and had found them of benefit to students, though that was not to say they were the best way to produce good men and women. He had found that industry preferred graduates who had had industrial experience in their university courses.

Professor Seegmüller asked if the essential difference could be explained between the given scheme and the work of companies' education departments, or industrial training units.

Professor Couger gave several points of difference. The work of an industrial training unit is oriented towards specific situations; it is training for the environment and the hardware of a particular firm. The given scheme provides general training. Also, while there are lots of excellent private institutes teaching systems analysis and design, there are not many good industrial training departments teaching systems analysis and design. To whom does one give the job of training in the first place? To people not properly trained themselves? These are the people who teach in industrial training units. Principally, however, the answer is that training is general on the one hand and specific on the other.

Mr. Land asked at which point in the curriculum projects were tackled.

Professor Couger replied that usually it was in the final semester.

Mr. Land then asked if projects were of specific size and duration.

Professor Couger replied that they were. He tried to get projects active by October, and they lasted until May. The format of the final report was fixed, but not its size.

Dr. Hanani asked if Professor Couger was preparing people enough for government jobs rather than industrial ones, as there was probably more employment there.

Professor Couger said that the choice of courses let people prepare themselves for specific areas. He was oriented towards business; his colleagues in the Political Science Department worked with local government.

Dr. Hanani wondered if they were computer oriented, as they should be.

Professor Couger affirmed that they were, for local government uses computers widely.

Dr. Andersen tried projects last year, but made the mistake of letting the students contact industry themselves. He was afraid of botching, and wondered if Professor Couger had any experience of mistakes.

Professor Couger said he had, but usually projects were simply not completed, rather than botched, and so could be continued the following year. Rarely had he been told not to come back.

Professor Melkanoff noted that a by-product of projects at the University of California, Los Angeles, was that they were helpful for job preparation.

Professor Couger commented, tongue in cheek, that that smacked of vocationalism.

Professor Ashenhurst pointed out the need for continual modification of both curriculum and projects. He asked how the follow-up was handled.

Professor Couger said that he had a good mixture of students and could use mature students for the follow-up. Ex-students who worked in a firm informally reported to him and gave him insight into what was going on. Employers sometimes hired the students doing these projects.

Professor Verrijn-Stuart imagined that occasionally there might be a shortage of new projects, and wondered if a backlog of projects had been built up at Colorado.

Professor Couger confirmed that there was a large backlog.

Mr. Bromberger asked what credit was awarded for a project.

Professor Couger replied that the credit was as for one course only (three hours per week), even though, as Professor Page remarked, students might do as much work as for two or three courses. In this respect, project work was similar to laboratory work.

Professor Ercoli did not understand what relationship builds up between a manager and his employees and the students. His experience was that when someone like these students was sent into a firm nobody wanted to deal with him because the staff considered it a waste of time. Similarly, when a computer consultant was sent in time was wasted because he didn't know enough about procedures. What, he asked, was the incentive for a firm to have two or three students in?

Professor Couger related his own story, because he thought that his situation was possibly not too dissimilar to Professor Ercoli's. He started six years ago at Colorado Springs, a city with a population of 65,000, in which there was no experience of contacts between industry and university. Starting thus from scratch, he spent two years building up confidence. He appointed an advisory council of twelve top executives which met three or four times a year. Its purpose was both educational and money-raising. Its members were led to understand the aims of the university well through contact, and so they took on students. When satisfied, they provided references and made encouraging reports to other firms. So, in time, a relationship was established between university and industry.

Professor Parnas did not think that answered the question: what advantage was it to a firm to have students coming in for six months and possibly using up other people's time, then disappearing never to be seen again? Did the firms not want full time people?

Professor Couger replied that the students' services were free, and the firms could benefit from the counsel of an instructor who was on the premises. In fact, the instructors sometimes became consultants with the firms involved. The scheme had certainly worked and provided a service.

Professor Parnas then said that he had been afraid to ask this, but when instructors are consultants with companies is there not surely a conflict of interests?

Professor Couger had never had any trouble with this. After all, the instructors were instructors in the first place. The firms had taken the Faculty into their confidence, too, where privacy and security of data were concerned. The companies and the university were heavily dependent on one another.

Professor Page interrupted the discussion to allow Professor Couger to finish his lecture.

* * * * *

In conclusion, let us examine the implementation of a business studies curriculum. Most European universities have Sociology and Psychology Departments and a Mathematics Department which could provide prerequisite courses. With co-operation and some outside assistance, it would not be so difficult to implement six or seven course curricula. As for Faculty training, IBM were extremely helpful in the USA. Also, a Faculty member with a basic interest in a subject can take a course in that and have extra funds to bring in outside experts to lecture to his class in areas in which he does not feel confident, so that he himself will be taught by these lectures.

Industrial co-operation, which has already been discussed, is essential. The firms provide money as well as experience.

Mr. Bromberger commented that some people think it wrong that universities should do practical work. But in teaching business studies one can't do without it - it's like teaching swimming. He had organized some projects himself, but had experienced difficulty in finding suitable firms, as industry in Glasgow (as in Newcastle) is in transition just now. Accordingly, projects were found within the university administration. At Strathclyde, it was not possible to have so much supervision as at Colorado, and so students looked after themselves more. Attempts had been made to integrate projects with other courses and aspects of computing, e.g., data structures. Employers in Britain were unwilling, he found, to take on students for projects since they did not fit in with the normal avenues of promotion within the firms. It was worth while, though, to have students employed at a low level.

Mr. Land had had experience with sandwich courses at polytechnics. Here the standard of projects was quite high, and though there was some difficulties there were enormous benefits.

Mr. Bromberger commented that sandwich courses carried no status! They were not much in evidence in Scotland. Some people had said that they produced conflict between the theoretical concepts learned and practice, with the theoretical concepts ignored in industry.

Mr. Land disagreed, and pointed out that many employers prefer students of computer studies from polytechnics rather than from universities because of the sandwich courses.

Dr. Hanani said that he had found firms suspicious of universities; they felt that if they were making money they were all right. One should start off in a small way and go slowly, as he was doing, and one's first case must be a success. Patience and little by little are the keywords - the university still has to prove itself to the community.

Professor Dijkstra noted that with respect to information systems Professor Couger had not included lectures on industrial espionage.

Professor Couger replied that he was leaving that up to the marketing troops!

Dr. Scoins was puzzled as to the conditions under which projects were undertaken. He had gathered that students were in their fourth year, after they had spent time in industry. What proportion of this year, he asked, did a student spend on his project?

Professor Couger reminded him that there were no sandwich courses in the USA: the project work was a student's first experience in industry. It was equivalent to a course of three hours per week for a term, this being called a three credit course.

Professor Page computed that this was 10% of a year's work.

Dr. Scoins thought that that was much less than Professor Parnas had had in mind.

Professor Page repeated his earlier comment that although the project work gave 10% of a year's credit it might take up 25% of a student's real time.

Professor Parnas said that he had had no particular amount of time in mind.

At this point, Professor Page closed the discussion and thanked Professor Couger for his interesting and enlightening lectures.

[1] Computing Newsletter for Instructors of Data Processing. P.O. Box 9630, Colorado Springs, Colorado, 80932, U.S.A. ($11 per year).

THE UNDERGRADUATE PROGRAM IN INFORMATION SYSTEMS DEVELOPMENT

J. D. Couger
Professor of Computer and Management Science
University of Colorado

The need for education related to information systems in organizations is discussed, and a curriculum is proposed for an undergraduate program. Material necessary for such programs is identified, and courses incorporating it are specified. Detailed course descriptions are presented, program organization discussed, and implementation questions considered.

Key Words and Phrases: education, management systems, systems analysis, information systems development, information analysis, system design.

CR Categories: 1.52, 3.51

Preface: This paper is the preliminary report of the committee on Undergraduate Curriculum in Information Systems Development, a subcommittee of the ACM Curriculum Committee on Computer Education for Management. The work of the committee has been supported in part by a grant from the National Science Foundation.*

Work on the undergraduate curriculum in information system development began after completion of the graduate curriculum.

Representatives of industry and educational institutions will be invited for a two-day, intensive review of the curriculum. Upon incorporation of the suggestions resulting from this meeting, the curriculum will be published in its final form in the Communications of the ACM.

Chairman of the undergraduate curriculum committee was J. Daniel Couger (University of Colorado). Other members of the C^{3}EM Committee, who participated in the preparation of this report, are:

Daniel Teichroew, U of Michigan, Chairman
Russell M. Armstrong, H.B.R. Singer Co.
Robert L. Ashenhurst, U of Chicago
Robert I. Benjamin, Xerox Corporation
Gordon B. Davis, U of Minnesota
John F. Lubin, U of Pennsylvania
James L. McKenney, Harvard U
Howard L. Morgan, Calif. Inst. of Technology
Frederic M. Tonge, Jr., U of Calif., Irvine

*NSF Grant GJ-356

INTRODUCTION

Whereas the graduate program in Information Systems prepares an individual to be a practitioner in the field, the undergraduate program prepares a person to be an effective computer user. The undergraduate student, therefore, combines his information systems course work with the academic area of emphasis in the field he plans to enter upon graduation, such as business or government. With the five-course option in information systems, the student essentially has a double major with, for example, marketing or political science or hospital administration, etc. Upon entering his career field, he will be able to participate effectively on a systems development team of users and practitioners. Until recently, an additional option would have extended a student's program beyond the normal four-year requirement. Today most programs allow enough electives to easily accommodate a second area of emphasis.

Not only does this approach produce a person who will operate effectively in a computer-oriented environment, it facilitates graduate work. After some experience in the field, he will be in a good position to decide whether he wants to take graduate work in that specialization or to enter the professional program in Information Systems. The undergraduate option covers graduate prerequisites, permitting the graduate program to be completed in only one instead of two academic years.

Most professional programs are one-year programs for persons who have an undergraduate degree in that field. A year of prerequisites is typically required for persons entering a graduate program from a different discipline.

The committee could have followed this practice, splitting the graduate curriculum into one-year segments, offering the first at the undergraduate level and the second at the graduate level. However, the committee decided that some areas of philosophy of the Information Systems curriculum needed to be stressed at both levels. Therefore, a different set of courses has been designed for the undergraduate option.

The committee recommends two options: (1) an administrative system option for students whose entry level job will be in an application area (e.g., marketing or hospital administration) and (2) a computer science option for students whose entry level job will be in the computer department.

OUTPUT CHARACTERISTICS OF GRADUATES

The undergraduate program in Information Systems Development equips the person to function in an entry-level position and also have a basis for continued career growth. The knowledge and abilities necessary to work effectively in this field may be characterized as obtainable by integrating concepts relating to people, models, and systems for the application of computer technology in the context of organizations and society.

Thus the requisite knowledge and abilities are conveniently grouped in six categories: (a) people; (b) models; (c) systems; (d) computers; (e) organizations; and (f) society. The first three categories are fundamental, and may be looked upon as providing tools for applications in the last three categories. A suggested list of needed knowledge and abilities is:

(a) people
- ability to hear others, as well as listen to them;
- ability to describe individual and group behavior and to predict likely alternative future behavior in terms of commonly used variables of psychology and economics;

(b) models
- ability to formulate and solve simple models of the operations research type;

(c) systems
- ability to view, describe, define any situation as a system--specifying components, boundaries, and so forth;
- ability to present in writing a summary of a project for management action (suitable to serve as a basis for decision);
- ability to present in writing a detailed description of part of a project, for use in completing or maintaining same;

(d) computers
- knowledge of basic hardware/software components of computer systems, and their patterns of configuration;
- ability to program in a higher-level language;
- ability to program a defined problem involving data files and communications structures;
- knowledge of sources for updating knowledge of technology
- ability to outline specifications for the computer-based part of a major information system, with details of task management and data base management components;

(e) organizations
- knowledge of the function of purposeful organizational structure, and of the major alternatives for that structure;
- knowledge of the functional areas of an organization--operations, finance, marketing, product specification and development;
- knowledge of typical roles and role behavior in each functional area;
- ability to identify information needs appropriate to issues and roles above;
- knowledge of how information systems are superimposed on organizational patterns, on the operational, control, and programming levels;
- knowledge of techniques for gathering information;
- ability to gather information systematically within an organization, given specified information needs and/or specified information flows;
- ability to specify, given information needs and sources, several alternative sets of information transfers and processings to meet needs;
- ability to make "rough-cut" feasibility evaluations of such alternatives;
- ability to develop positive and negative impacts of a specified information system on specified parts of an organization;
- ability to develop specifications for an information system, addressing a given organizational need, and determine the breakdown into manual and computer-based parts;

(f) society
- ability to articulate and defend a personal position on some important issue of the impact of information technology and systems on society (important, as defined by congressional interest, public press, semitechnical press, etc.).

The last four abilities in both the "computers" and "organizations" categories, (d) and (e), are the key to the information systems development approach. The ability to analyze alternatives and to make "rough-cut" designs is particularly critical in the changing information systems environment of today.

The knowledge and abilities listed are testable in the academic environment--by written or oral examinations, successfully operating computer programs, case discussions, judgment by a panel of experts and/or peers, and other commonly accepted means. Besides attaining knowledge and abilities, however, it is important for the student to have gained some experience in prototype work situations. A suggested list is:
- having gathered information in "real" organization;
- having participated in a project where operations research techniques are utilized;

having served as a member of a project team developing a specified information system;
having participated in planning and conducting an oral presentation (and selling) of the results of a team project.

BACKGROUND ON THE INFORMATION SYSTEMS FIELD

Two earlier reports of the committee provided the background for degree programs in information systems. The position paper[1] provided the justification for the degree programs, citing surveys on demand for such persons and the kinds of capabilities required of practitioners in the field. The second report provided detailed course descriptions for a graduate-level professional program.[2] It is unnecessary, therefore, to repeat that material in this paper. The reader is encouraged to obtain copies of those papers as a background for the undergraduate curriculum recommendations.

THE CURRICULUM

During his first two years of college work, the information systems major should satisfy the following prerequisites:

(1) finite mathematics, including the fundamentals of formal logic, sets and relations, and linear algebra;
(2) elementary statistics, including the fundamentals of probability, expected value, and construction of sample estimates;
(3) elementary computer programming, including problem analysis and algorithm synthesis, and competence in a higher-level language;
(4) elementary economics, including micro-economics and theory of the firm, and price theory;
(5) elementary psychology, including fundamentals of personality formation, attitudes, and motivation.

Upon completion of prerequisites, the student begins the required courses in his field of interest. Two options are suggested:

(1) the administrative systems option for the student whose entry level job will be in a field of application of the computer;
(2) the computer science option for a student whose entry level job will be in the computer department.

[1]Teichroew, D. (Ed.), "Education Related to the Use of Computers in Organizations," (Position Paper - ACM Curriculum Committee on Computer Education for Management), Communications of the ACM, Vol. 14, No. 9, Sept. 1971, pp.573-588.
[2]Ashenhurst, R. L. (Ed.), "Curriculum Recommendations for Graduate Professional Programs in Information Systems," Communications of the ACM, Vol. 15, No. 5, May, 1972, pp. 363-398.

The Administrative Systems Option

The student selects an area of application to combine with his information systems option. For example, he may desire to work on a team developing marketing systems. If the student takes an area in a functional field such as accounting, finance, or marketing, he would be required to satisfy the graduation requirements of the School of Business. However, he may be interested in governmental applications; therefore, his field would be obtained from the Public Administration Department, often in the College of Arts and Sciences. If he is interested in a field in production, he may find that content in the Industrial Engineering Department in the College of Engineering. A person interested in Hospital Administration may take requirements from the Medical School and the School of Business.

All of the above programs allow for electives, easily accommodating the five courses in the system development curriculum.

Figure 1 shows the courses in the Administrative Systems option. The five courses in the field of application are illustrated by an accounting option. The left-hand side of Figure 2 provides course sequence for the Administrative Systems option. Most schools include courses like B1 and B2 in their core requirements. For these schools the Administrative Systems option consists of 5 courses.

The Computer Science Option

It is assumed that the core requirements in the Computer Science option include the following courses specified in Curriculum 68:[1] Data Structure (Course I1), Programming Languages (Course I2), Computer Organization (Course I3), and System Simulation (Course A4).

For schools which have adopted Curriculum 68, the Computer Science option in Information Systems will consist of 5 courses. Referring to Figure 2 (right-hand side of the page), the following courses are equivalent: C1(I1), C2(I3,I4), and B1(A4).

Course descriptions will be provided in the next section.

[1]"Curriculum 68, Recommendations for Academic Programs in Computer Science," Communications of the ACM, Vol. 11, No. 3, March, 1968, pp. 151-197.

Prerequisites Freshman and Sophomore Years	Finite Mathematics	Elementary Statistics	Introductory Computer Programming	Microeconomics	Elementary Psychology
	CORE COURSES, SELECTED AREA OF EMPHASIS (e.g., Business, Political Science, Hospital Administration, etc.)				
Junior Year	A14 System Concepts and Implications	C14 Information Structures and Software Design	C23 Computer and Communication Systems	Course in Field of Application (e.g. Managerial Accounting)	Course in Field of Application (e.g., Intermediate Accounting)
Senior Year	A3, D1 System Analysis	D23 System Design and Implementation	Course in Field of Application (e.g., Cost Accounting)	Course in Field of Application (e.g., Auditing Theory)	Course in Field of Application (e.g., Accounting Theory)

Figure 1

Prerequisites and Required Courses in Administrative Systems Option

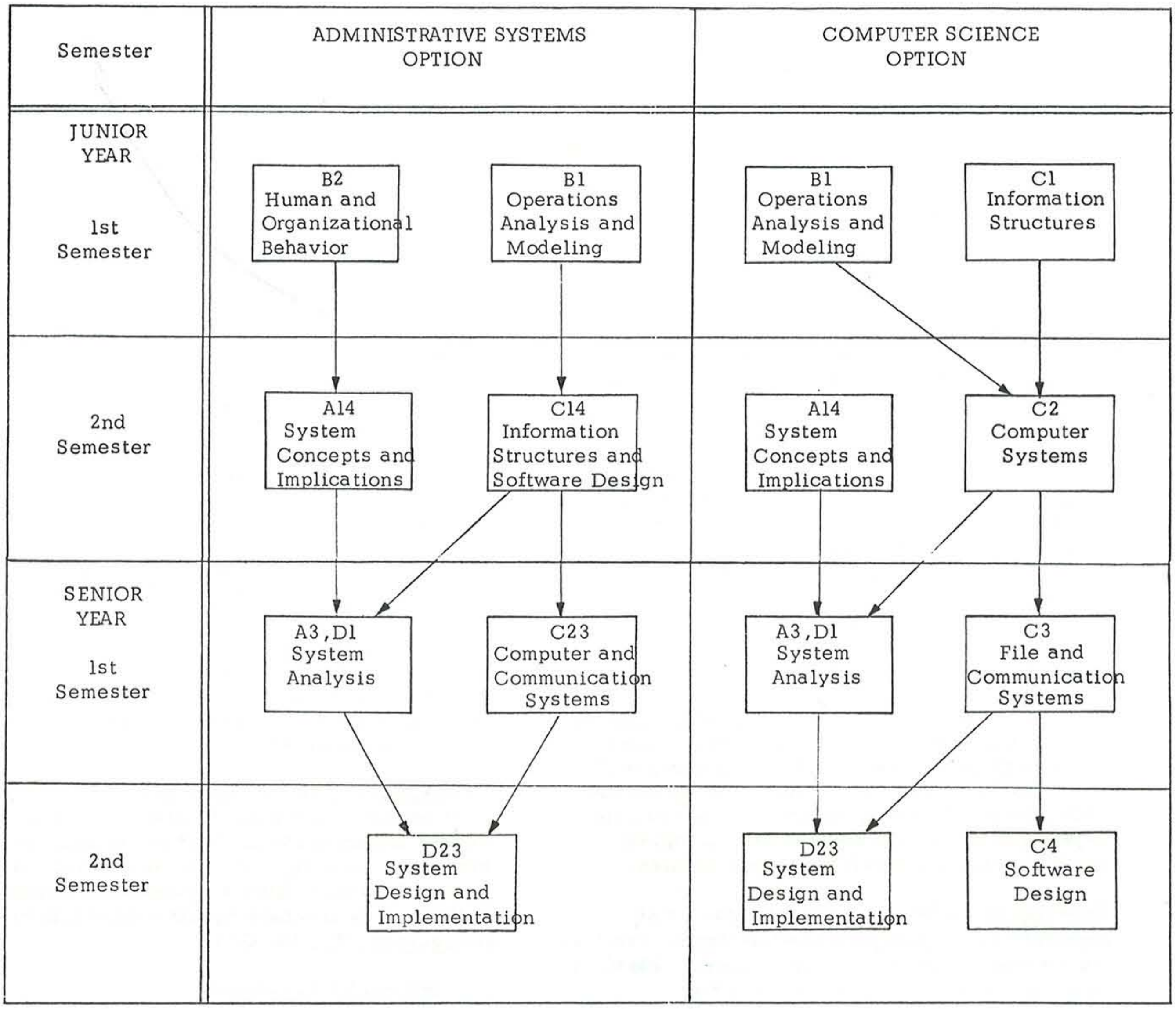

Figure 2
Course Sequences

COURSE DESCRIPTIONS

The full list of courses for both options are provided below: (Some modification will occur before the final report.)

A14. System Concepts and Implications

Objectives: To introduce the student to the information analysis and system design curriculum. To identify the basic concepts that subsequent courses will draw upon: the systems point of view, the organization as a system, its information flows, and the nature of management information systems. To explore the current and projected social and economic effects of information systems in organizations.

Description: The systems concept. Defining a system. Systems analysis. Management systems. Management information systems. Effects on organizational practice. Privacy and the quality of life. The individual and the social system. Prerequisites: elementary economics, B2.

A3, D1. System Analysis

Objectives: To identify the decision requirements for the management of an organization. To analyze the design of an information gathering and processing system intended to facilitate decision making and planning and control. To analyze the concept of an information system. To review the approaches and techniques available to evaluate existing systems. To examine the concept of common data base for all functional modules.

Description: Information requirements for an organization. Operational level systems. Tactical level systems. Strategic level systems. Planning for a comprehensive information system. Introduction to the system life cycle. System life cycle management. Basic analysis tools. Defining logical system requirements. Summary and introduction to physical system design. Prerequisites: A14, C14.

B1. Operations Analysis and Modeling

Objectives: To introduce and exercise a range of analytical and simulation modeling techniques useful in decision making in the system design environment. To consider the function of such models as guides for data collection, structures for data manipulation, and as systems for testing assumptions and generating a variety of alternatives. To identify the problems of data collection, maintenance and accuracy when using models to assist decision making activities.

Description: Characterization of scheduling situations. Analysis of allocation problems with mathematical programming. Queuing models. Inventory models. Use of simulation models. Prerequisites: finite mathematics, elementary statistics, elementary computer programming.

B2. Human and Organizational Behavior

Objectives: To introduce the student to the principles governing human behavior, particularly as they relate to organizations and to the introduction and continued operation in organizations of computer-based information systems.

Description: Individual behavior. Interpersonal and group behavior. Organizational structure and behavior. The process of organizational change. The implementation and introduction of information systems.
Prerequisite: elementary psychology.

D23. System Design and Implementation

Objectives: To provide the knowledge and tools necessary to develop a physical design and an operational system from the logical design. To provide students with supervised and structured practical experience in the development of computer-based systems.

Description: Basic design tools and objectives. Data base development. Program development. System implementation. Post implementation analyses. Develop a system; alternatives: a system for a local firm, a system for a university/college, a system for a hypothetical application. Prerequisite: A3, D1, C23.

C1. Information Structures

Objectives: To introduce the student to structures for representing the logical relationship among elements of information, whether program or data, and to techniques for operating upon information structures. To examine the methods by which higher-level programming languages implement such structures and facilitate such techniques.

Description: Basic concepts of information. Modeling structures--linear lists. Modeling structures--multilinked structures. Machine-level implementation structures. Storage management. Programming language implementation structures. Sorting and searching. Examples of the use of information structures. Prerequisite: elementary computer programming.

C2. Computer Systems

Objectives: To provide a working view of hardware/software configurations as integrated systems, with (possibly) concurrently functioning components.

Description: Hardware modules. Execution software. Operation software. Data and program handling software. Multiprogramming and multiprocessing environments.
Prerequisite: elementary computer programming.

C3. File and Communication Systems

Objectives: To introduce the basic functions of file and communication systems, and to current realizations of those systems. To analyze such realizations in terms of the tradeoffs among cost, capacity, responsiveness. To examine some systems integrating file and communication functions, such as the organizational data base system or the computer utility.

Description: Functions of file and communication systems. File system hardware. File system organization and structure. Analysis of file systems. Data management systems. Communication system hardware. Communication system organization and structure. Analysis of communication systems. Examples of integrated systems.
Prerequisite: C2

C4. Software Design

Objectives: To examine how a complex computer programming task can be subdivided for maximum clarity, efficiency, and ease of maintenance and modification, giving special attention to available programming and linking structures for some frequently used interface programs, such as file and communication modules. To introduce a sense of programming style into the program design process.

Description: Run-time structures in programming languages. Communication, linking, and sharing of programs and data. Interface design. Program documentation. Program debugging and testing. Programming style and aesthetics. Selected examples.
Prerequisite: C2

C14. Information Structures

Objectives: To introduce the student to structures for representing the logical relationship among elements of information, whether program or data, and to techniques for operating upon information structures. To examine how a complex computer programming task can be subdivided for maximum clarity, efficiency, and ease of maintenance and modification.

Description: Basic concepts of information. Storage management. Programming language implementation structures. Examples of the use of information structures. Linking, and sharing of programs and data. Interface design. Program documentation. Program debugging and testing.
Prerequisites: elementary computer programming, B1.

C23. Computer and Communication Systems

Objectives: To provide a working view of hardware/software configurations as integrated systems. To introduce the basic functions of file and communication systems, in terms of the tradeoffs among cost, capacity, responsiveness. To examine some systems integrating file and communication functions, such as the organizational data base system or the computer utility.

Description: Functions of file and communication systems. Analysis of file systems. Data management systems. Communication system organization and structure. Examples of integrated systems. Hardware modules. Execution software. Operation software. Data and program handling software. Multiprogramming and multiprocessing environments.
Prerequisite: C14.

IMPLEMENTATION

The section on implementation in the graduate curriculum report is equally appropriate to the undergraduate curriculum.

Although the committee wishes to refrain from recommending which academic departments house the system development curriculum, several comments are necessary in this regard.

This curriculum may be housed in the School of Business, Computer Science Department, Industrial Engineering Department--depending on the particular academic structure and interests of faculty in a given institution.

However, unless an institution is particularly rich in computer faculty, it is important to pull together all faculty talent through joint appointments with the department where the system development program is to be housed. This approach enables implementation in the shortest time span. Later, when the program grows to the point of multiple sections of courses, it may be advisable to begin hiring specialist faculty and to establish a separate department for the program.