Computing Fundamentals for IT and IS Programs

George Stockman
Computer Science and Engineering
Michigan State University

Abstract

Information technology impacts the careers of all professionals and thus its concepts are important for education. Our Department developed a unique sophomore level course to provide fundamentals to students with majors in Business, Telecommunications, or Engineering who with a Specialization in IT. While the initial course was experimental in nature and the availability was only guaranteed for students in the IT Specialization, the course was designed as if it would eventually become available to all university majors. The course philosophy, objectives, and content are described below. The course has been a successful prototype that may be used in part or in whole by other academic units preparing professionals for a future where computers and communication are pervasive.

Introduction

Computers, communication networks, and digital representation of all kinds of information have thoroughly transformed commerce and the way people work and play. Not only has education itself been transformed by using IT, but it also faces the task of getting students to be fluent in the concepts and practical use of IT. Universities have undergone various restructuring along IT lines and have designed new majors and curriculum. Two cases of special note are Indiana [1] and RPI [2], both of which have made IT or "informatics" pervasive in their curricula. Well developed program proposals for Information Technology and Information Systems majors were presented at ASEE2004[3].

In order to meet demands of students and employers, Michigan State University introduced an IT Specialization for students from three of its Colleges [4]: the Specialization is a set of courses designed to supplement the programs of students enrolled in bachelor's degree programs in the Eli Broad College of Business, the College of Communication Arts and Sciences, and the College of Engineering. Students completing the specialization should be prepared for employment in technology-oriented environments and should understand the evolving impact of information technology on society. Completion produces an annotation on the diploma. There are three required core courses, one from each college, and a required capstone course; and, there are six selective credits that can be chosen from courses offered by the three colleges. Each core course requires no more than one prerequisite course from MSU.

The Department of Computer Science and Engineering provides two courses critical to the IT Specialization. The first is an existing computer literacy course [5] that is taken by over 4000 MSU students each year and that is required for most majors eligible for the IT specialization. The second course, the required CSE 240: Informatics, was developed in 2002 to provide fundamentals of computer science for non CS majors and is the subject of this paper. This course was designed by studying courses taught at other universities, by considering what background would be needed by non CS students in their upper level IT courses, and by anticipating what lasting value graduates should take into their careers.

The formal catalog description of CSE 240 is: "***digital representation of objects, such as numbers, signals, and 3D shapes; algorithms that operate on digital objects; devices that store digital objects and networks that distribute them; how information systems support various applications***."

CSE 240:Informatics Course Objectives

1. with specific information encoding techniques and specific algorithms. to provide an understanding of several fundamentals of information representation and management.
2. to provide prerequisite background to those continuing in additional course work in the broad IT/IS area.
3. to provide skills needed to manage information, such as choosing tools, system components, or information representations and algorithms.
4. to familiarize the student with specific applications.
5. to familiarize the student

Overall, the course CSE 240 deals with the science of information and how it is used by people and information systems. This includes how we represent, or encode, objects and concepts, and how these encodings are processed by algorithms, communicated via networks to machines and people, and stored in memories. Students learn, for example, how music is represented digitally, how Internet messages are coded, transmitted, and used, how database records are used to answer general queries, and how the geometry of a manufactured part is encoded and used in computer-aided-engineering. The concept of an algorithm is studied along with the activity of programming: example programs are done for homework to reinforce the concept, but not to learn a particular programming language. (It is important to emphasize that CSE 240 is NOT a "programming course", in contrast to comparable level courses in some other IT curricula.) Similarly, several computer tools are used, such as GIMP to work with images, and VRML to design and display 3D scenes. The particular tools themselves are not important: what they do for us and why are important. Moreover, it is important for students to become comfortable in learning new tools. Applications and case studies are very important in the course so that students can appreciate how information and information systems impact manufacturing, medicine, transportation, etc. It is postulated that these fundamentals will be of great value to our graduates in their future careers, whether as managers or executives, as technical people responsible for maintaining databases or computing facilities, or in many other capacities, perhaps not yet defined by society.

Incoming students are expected to be prepared at the high school level mathematical background in algebra, functions, and sets. Students should also be computer literate; that is, accustomed to using the Internet and www, email, and tools such as a spreadsheet and word processor. They should also be familiar with the most important computer systems vocabulary and the hardware and software making up a computer system. Students completing CSE 240 will be able to take more advanced CS, IT, and IS courses in data structures, databases, networking, security, multimedia systems, geographic information systems, business information systems, etc., whether taught by a CSE Department or any other department.

Fundamentals from a Computer Science perspective

Perhaps there are three majors themes threading through the entire course. One major theme is how information is represented, stored, and shared in a network of computers. A second related theme is how information is transformed by algorithms, and a set of significant algorithms is studied. The course is not a programming course, but it helps students understand what a program is and does. Algorithms are implemented via high level programming in the home works and students are responsible for carrying out hand calculations. The third major theme involves case studies of a significant set of real applications. Classical cases are presented by the instructor and students present cases from their own experience and from current events.

Some example units

Although a weekly topic outline is given below, it is important to give a deeper description of sample course units in order to better convey the course objectives, level of coverage, and relationship to career and curriculum paths. The unit sketches below describe some of the content details as well as some teaching/learning strategies.

Information representation:

Encoding various sets of symbols using binary bits is the basic goal. Coding small integers and character sets is studied. Students must understand the fundamental principle that 2^n objects can be represented by a string of n bits. (The author has had fun teaching students date and card guessing games based on binary coding, where the binary decision is red versus black, animate versus inanimate, up versus down, etc. For example, "I am envisioning a man and a woman on a ship embedded in an iceberg" results in the binary code of 1100 or the equivalent decimal number ten.) Algorithms for converting numbers between binary and decimal are studied. A/D conversion of continuous quantities is discussed. Within this topic, the author may access a water resources site that shows the water level in the Red Cedar River plotted over the last five days. We go through an algorithm for converting the apparently continuous signal of river levels to an integer sequence, say, one reading for each hour. This should certainly be adequate for a canoeist deciding whether or not the river is passable: the question is raised whether or not is adequate for computing pollution concentrations. An interesting divergence is to brain storm about how those river levels are sensed – is there a little man with a ruler and a cell phone along the river? This could be a homework problem to be researched.

In communicating binary information, possible errors and error detection and/or correction is discussed. Students must understand the use of odd or even parity for error detection with the ASCII table as an example. The notion of Hamming distance is studied in an informal manner. Given a set of binary strings with minimum Hamming distance of 3, students should understand how single errors can be corrected, but that additional errors are still a problem. Communication of information over networks is organized into another related unit. Information encoding continues with discussion of digital audio – speech and music. The analog representation of sound is studied along with the characteristics of human hearing. Protocols for digital storage on CDs and DVDs is studied along with the capacities of such media, ultimately leading to discussions about how such relatively simple devices have had an important impact on society. The mechanical characteristics of accessing information on rotating media is also covered, although in a related unit that also treats the materials, physics, cost, and history of such devices. The information representation theme continues with the representation of images, video, and 3D objects (see below).

Organizing data via sorting or via trees:

Linear search of an array of keys is studied in contrast to binary search of an ordered set of keys. Implementations of these simple algorithms via programs in C++ or Javascript are given to the class and their performance is studied live. Recorded runtimes show a linear time complexity for linear search and a logarithmic complexity for binary search. The point is easily made that a phone book of ten million entries should not be maintained and linearly searched as an unordered array. Students can also readily see that, while search speed may be adequate using binary search of an ordered array, keeping the data in order with updates is a problem. This motivates the B-tree data structure. Students are expected to be able to describe the search and update mechanisms (via English): later on, they are expected to describe the time costs incurred in finding a telephone entry in a particular B-tree stored on rotating media. At a higher level, storing an English dictionary using a B-tree is discussed along with use of the dictionary to do spell-checking in a word processor. The structure of a trie is introduced for contrast. In the database unit, some discussion is done on how individual "flat files" can be accessed and coordinated to answer a compound query involving multiple flat files. Simple sorting algorithms, such as selection sort and bubble sort, are studied and simulated and their n^2 behavior compared to an efficient sort, such as quicksort.

Computer-Aided Design/Manufacturing/Engineering (CAD, CAM, CAE):

Representing object geometry by volumetric and mesh methods is discussed. Basics, such as representing points, edges, and faces, are examined in detail along with translations and rotations of these objects via transforming individual points. Transformations are done with detailed trigonometry in two dimensions only. (On an aside, students were taught how to derive the 2D linear formulas to correspond feature points of aerial images to maps. Having the formulas enables new structures seen in the aerial image to be placed on the map in order to update it. This is an important application of IT for land use planners, geographers, tax accountants, and even politicians.) The study of CAD/CAM/CAE provides a high level understanding of how industrial production and the supply chain is supported by highly technical IT. CAD enhances the designer's ability to create object designs, which are also precise enough for contracting with suppliers and for the physical analysis of CAE. It is a major contribution of computer science that much of the design can be evaluated using the representation and without building the actual object, thus saving time and material cost. Once a design is accepted CAM is used to manufacture the real objects and possibly to inspect those instances using model-based computer vision. There is a treasure chest of good material on the web to motivate this unit. Software vendors praise the functionality of their software products and often provide links to applications by other companies and there also are many examples on university research web pages.

Weekly topics for the Fall 2003 course

Below is the initial topic calendar for CSE 240 for Fall 2003; the actual sequence of delivery varied somewhat from the order in this list due to use of a new textbook.

1. Course intro. and motivation. Analog and digital information; advantages of digital coding; ASCII code, binary integers, principle of representing N objects or values using log N bits for each.

2. The notion of an algorithm; an algorithm for number base conversion; an iterative averaging algorithm to compute the square root of any positive number. Strings, arrays, and 1D signals; Linear search of an array for a target string.

3. Insertion sort in sequential memory; binary search; comparison of linear and binary search; characteristics of storage devices, especially rotating disks; linked memory organization for lists

4. continue with storage devices; the notion of B-trees and storing shallow bushy trees on disk. Storing data in databases.

5. Case study: implementing a large dictionary using a B-tree; implementation using a trie; database concepts;

6. continue with database concepts; answering compound queries by combining multiple flat files

7. The image as an object with rows of pixels; Encoding pictures; binary images; algorithms for thresholding, smoothing using averaging; median filtering; edge detection ; color representation; video as a sequence of images; case study: telemedicine and distributed image interpretation

8. intuitive methods of image and video compression; color tables; lossless versus lossy compression with examples; MPEG compression using "motion vectors".

9. representing 3D objects; VRML notions and scene graphs; shape, location, appearance;

10. CAD systems; CAE; case study in use of CAD/CAM/CAE.

11. 1D signals; audio signal representation and transmission; bandwidth; noise;

12. transmission of information using wires, wireless, satelite; cellular phones; telematics and telemedicine.

13. basic ideas of networking; circuits and packets; integrated services and convergence; overall organization of LANs and the Internet; case study of iMan (Info. Manager used by GM).

14. concepts in information management; integrity, security, privacy, ownership, ethical issues; encryption and biometrics

Textbooks:

The following two textbooks have been used to support the course. Both are excellent books, but neither completely addressed all of the course goals and topics. Several other texts have become available and more are expected due to the importance of IT and the curricula related to it. Due to the uniqueness of our course, we must expect that parts of it must be covered by outside and auxiliary information from the instructor.

The text by Cyganski *et al* [7] covers a good set of topics at an appropriate sophomore level of technical detail. It was designed for engineering students. It is strong on networking and information coding. It has perhaps too much emphasis on networks and the web and is deficient on database concepts. The text is deficient in coverage of algorithms also; however, the text comes with a valuable CD containing several nice java applets illustrating topics of the text.

The text by Synder [6] was designed for first year students and thus contains a lot of material that CSE 240 students know coming into the course. Also, there is more verbosity than a sophomore would require. It has good material on algorithms and on databases as well as the web and Internet. Of primary interest at the time of text selection, the text develops programs in Javascript. These programs can be created by a simple editor, are easy to run from the browser, and can produce satisfying output for the student familiar with HTML.

Course format

Class time is roughly divided into (a) 50% lecture/discussion, (b) 20% group work and subsequent discussion, and (c) 25% discussion of Internet items, computer programs, or practical problems (d) 5% for exams. The classroom has an Internet connection to provide live browsing and also the running of programs. Software tools can be demonstrated. Some example programs in C++ and Javascript were studied in the first two offerings. Excel programs were used in the third offering.

At least half of the classes included a session for collaborative learning [8,9], where students discussed solutions to important problems and presented their conclusions to the entire class after about 20 minutes of group deliberation. Two examples are given here to help convey outcomes intended by the course design. One problem concerns *how IT can improve the delivery of medical care.* The problem is given to the students on paper along with a few questions to address. One paper from each group is usually collected for the purpose of attendance and to make sure all important issues are reviewed for the entire class. Most groups will not think of several issues but will concentrate on one or two; however, over all groups most critical issues will arise. For example, regarding the problem of medical care, telemedicine will come up as will the use of networked databases for patient records. While the integrity of information is likely to come up, the issue of privacy protection may not. One or two groups thought about how expert systems might do a better job than some staff members in checking a patient record for harmful drug interactions. A second successful group discussion topic was *"telematics", defined as mobile information access from an automobile.* As a group, students are familiar with cell phones and products for automobiles, such as General Motors' OnStar, and their growing use in society and could relate this to location aware and mobile computing. An important role of the teacher is to make sure that students understand how the features of the popular commodities are implemented by fundamental IT.

The course has been supported by a teaching assistant (TA). The TA had office hours in addition to those of the instructor to support students with their homework. However, the most important value was in being able to schedule open laboratory time so the TA could assist students in using new software tools.

Graded homework problems are assigned every two weeks. Group work is encouraged for some problems and forbidden for others. Two hour exams and a final exam are given. Some exam problems are quantitative, where students are required to apply algorithms and produce an answer. For example, students are expected to be able to convert between binary and decimal number representation and smooth an intensity image by neighborhood averaging. Others questions are essay/reflective; for example, students are expected to be able to give a technical discussion of the characteristics of rotating memory, and to discuss the impact of networking on enterprise computing.

Problems encountered:

Choosing a textbook for a uniquely defined course is a problem. With both textbooks [6] and [7], auxiliary information was critical, particularly with respect to the topics of algorithms, images, and CAD/CAM/CAE. Some students had trouble learning from sources other than the text. Of course, the new course is also a problem for instructor preparation. Materials, exercises, software tools, etc. have to be identified and collected by a person who probably is not familiar with some topics. The first offering of the course overlapped too much with the core required course TC 201, which also covers networking and multimedia. This overlap was reduced in the second offering yielding more class time for increased coverage of database concepts. Launching a new specialization and a new course creates a ramp in attracting and advising the targeted students. In the first offering, about half of the students were lower division students intending to complete the IT specialization and about half were upper division students using the course as an elective. Thus there was a large variance of student background and motivation. A few of the students were actually majors in CS,EE, and Computer Engineering – not at all in the clients for whom the course was designed. Class makeup shifted to about 70%/30% in the second offering.

Concluding Discussion

The course CSE 240 was designed from requirements and is not the result of evolutionary changes to previous courses. The course design, philosophy, and text by Cyganski and Orr [6] was a big help; not only did it provide good material, but it also established that such a course could be done and was important. After CSE 240 was created, a report appeared from the

National Research Council's Committee on Information Technology Literacy entitled "What Everyone Should Know About Information Technology". This report and the related preface to the Snyder text provided additional support [7]. Some of the topics taken from Cyganski and Orr, and some that were not, originated in prior higher level CS or EE courses. This material has to be presented in a different manner to students with less CS background. Mathematical details and computer programming details can not interfere with conceptual and procedural understanding. For example, justification for the iterative square root algorithm has to be given using logic other than that based on the calculus of Newton-Raphson. Implementing and testing programs, while necessary for CS majors, would add an unnecessary amount of time and frustration to non CS students. Similarly, students have to get comfortable with the idea and use of .jpg images, even though only a small fraction of professors understand the mathematical basis of all of .jpg's features. The use of indexed color in .gif is, on the other hand, understandable in detail. Although students sometimes struggled with the algorithms and quantitative work, and perhaps the first and second year students more than the upperclassmen, students at all levels and in all majors showed that they could do the work. The instructor had to dynamically assess learning and slow down or review when students struggled with assignments. This happened with the database homework: an additional two 30-minute lecture/discussions were needed before students could perform well on the assignment.

The makeup of the class has been more varied than classes taken by CS majors due to differences in majors and in experience. While this made it more difficult to create appropriate assignments as well as lecture level, it added to the collaborative work done in and out of class[8]. Business students might have a different perspective on an issue than would engineering or telecommunication students. This resulted in good dialog on the several group work issues and a healthy variety of proposed solutions.

A final thought is on the "research aspect" of the course. Every student, even the engineering senior, was confronted with something new in this course. Similarly for the instructor! The instructor tried to nurture an attitude toward looking forward to problems or questions and researching and discussing possible solutions. No doubt that this mode will be a major part of their future. What is a gene chip and how does it revolutionize scientific work? How can a surgeon in the US operate on a patient in Europe? How can we know that a particular baseball player hit 245 against left handed pitchers and 288 against right hand pitchers over 11 years? Why should businesses be permitted by government to send us advertisements in emails or pop-ups? Is government policy different in the US relative to the European Common Market? As science and society march on, we see news items in the papers and on the web. We all need to maintain some fundamental knowledge and procedures on which to build future knowledge. An instructor of an IT course should always be ready to switch to research mode and use IT itself to research issues and technology [9]. So should students.

References

[1] www.informatics.indiana.edu website for informatics at Indiana University
[2] www.rpi.edu/dept/IT/index.html website for IT at Renssalaer
[3] B.M. Lunt, J. Ekstrom, E. Lawson, R. Kamali, J. Miller, S. Gorka, and H. Reichelt, Defining the IT Curriculum: The Results of the Past 2 ½ Years, Session 2558, Proc. ASEE National Conference, (17-21 June 2004) Salt Lake City.

[4] www.bus.msu.edu/undergrad/its.html gives current details of the IT Specialization at Michigan State University.
[5] www.cse.msu.edu/~cse101 website for the literacy course at MSU
[6] D. Cyganski and J. Orr, Information Technology, Inside and Outside, Prentice-Hall (2001) ISBN 0-13-011496-0
[7] L. Snyder, Fluency with Information Technology, Addison-Wesley (2004) ISBN: 0-321-12201-1. For the report from the NRC defining IT fluency visit http://www.cra.org/CRN/html/9905/affiliate/ls.6 1.t.shtml
[8] Gabelnick, F., MacGregor, J., Matthews, R., & Smith, B.L. (1990). Learning communities: Creating connections among students, faculty, and disciplines. *New Directions for Teaching and Learning*, No. 41. San Francisco: Jossey-Bass Publishers.
[9] www.howstuffworks.com one excellent source of information on how devices work

Biographical Information: GEORGE STOCKMAN

Research interests include computer vision and graphics. Teaching interests are in computer vision, artificial intelligence, data structures and programming. He received a BS and an MAT in Math-Ed from East Stroudsburg State and Harvard, respectively, an MS in Computer Science from Penn State in 1971,and the PhD in Computer Science from Maryland in 1977. He has worked in industry for four years and been at MSU/CSE since 1982.