



An Accelerated Introductory Computer Science Course Sequence for Non-Traditional Master's Students

J. Denbigh Starkey Ray S. Babcock Anne S. DeFrance
Computer Science Department
Montana State University
Bozeman, MT 59717

[starkey|babcock|defrance]@cs.montana.edu

Abstract

All Computer Science Departments that offer graduate degrees share a common problem—many students, with degrees in other disciplines, apply to pursue a Master's degree in Computer Science, but have to make up a large number of undergraduate deficiencies before taking graduate classes. These students are typically above traditional age, have jobs and families, and cannot easily afford to make the time commitment that is necessary to be able to complete the deficiencies before taking the classes that become their graduate program. The Computer Science Department at Montana State University has a solution to this problem.

We have developed two accelerated introductory courses, CS 252 and 254, where enrollment is restricted to students who already have degrees in other disciplines. These classes, which have two lectures and six hours of laboratory work each week, replace 26 semester credits of regular coursework, and let potential graduate students remove their required undergraduate deficiencies about a year faster than would otherwise be possible.

Introduction

The Computer Science Department at Montana State University is a CSAC/CSAB accredited program whose undergraduate curriculum is based on the breadth-first model. The development of this breadth-first approach at our university has been extensively reported in the literature [6, 7] by the first author of this paper and his colleagues, so we will not describe it further here.

Permission to make digital/hard copy of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage, the copyright notice, the title of the publication and its date appear, and notice is given that copying is by permission of ACM, Inc. To copy otherwise, to republish, to post on servers, or to redistribute to lists, requires prior specific permission and/or a fee.

SIGCSE '97 CA, USA
© 1997 ACM 0-89791-889-4/97/0002...\$3.50

Traditional undergraduate course sequences like our breadth-first sequence are geared towards entering freshmen, who are not always highly motivated. We have developed two courses, CS 252 and 254, which are designed for older, non-traditional, students, who have no formal training in Computer Science, but want to complete graduate degrees in the discipline. The success of these courses is the topic of this paper.

The underlying philosophy behind these courses is that older students, who have previous degrees in other disciplines, are highly motivated, and have learned the skills required to pick up new material far faster than the traditional freshmen can. We also make the assumption that many of these students have the potential to become excellent graduate students. These courses are designed to help them to achieve this potential.

In this paper we will

1. describe the material that we consider to be “undergraduate deficiencies” that students must complete before formally entering our graduate program,
2. describe the two accelerated courses, CS 252 and 254 in some detail,
3. describe the relationship between the accelerated courses and our distance learning MS degree, and
4. discuss the advantages and possible disadvantages of the approach that we have taken.

Undergraduate Deficiencies for the Master's Program

The Computer Science Department at Montana State University lists the courses shown in Table 1 (next page) as “undergraduate deficiencies” that must be completed by any Master's degree applicant in addition to their graduate coursework.

Number	Title	Lect. Cred.	Lab. Cred.	Total Cred.
CS 160	Introduction to CS	3		3
CS 210	OOP using C++	2	1	3
CS 221	Computer Science I	3	1	4
CS 222	Computer Science II	3	1	4
CS 223	Computer Science III	3	1	4
CS 324	Computer Science IV	3	1	4
CS 331	Computer Architecture	3		3
CS 350	Theory of Computation	3		3
CS 351	Software Engineering	3	1	4
CS 355	Design of Prog. Lang.	3		3
CS 418	Operating Systems	3		3
CS 440	Computer Networks	3	1	4
CS 450	Compilers	3	1	4
Total CS deficiencies:				46
Math 181	Calculus I	4		4
Math 182	Calculus II	4		4
Math 221	Matrices	3		3
Total Math deficiencies:				11

Table 1: CS Undergraduate Deficiency Classes

Applicants with Computer Science undergraduate degrees will usually have already completed all of these courses.

Many applicants, however, do not fit this traditional model. They are older students with degrees in other disciplines who have usually been in the work force for several years, and who now want to complete a Master's degree in Computer Science so that they can change (or advance) their careers.

Some of these students are full time, while others need to continue working at their current jobs, so they must take courses on a part time basis. All of our distance learning students (next page) are part time.

Students with undergraduate deficiencies can select either the thesis option (20 semester credits coursework and 10 credits thesis) or the project option (26 credits coursework and 4 credits project) for their graduate program. These courses must be taken in addition to their undergraduate deficiencies. Students with undergraduate CS degrees are required to take the thesis option.

Accelerated Courses

The focus of this paper is two courses, CS 252, Accelerated Introduction to Computer Science I, and CS 254, Accelerated Introduction to Computer Science II, which replace 26 credits of the undergraduate prerequisite courses. The catalog descriptions for these two courses are given in the Appendix on the last page of this paper.

The two classes are five credits each, with two hours of lecture and six hours of scheduled laboratory work each week (our university assigns one credit for two scheduled labora-

tory hours). CS 252 is offered every Fall Semester, and CS 254 is offered every Spring. The lectures are taught by faculty (the authors of this paper) in a lecture room with 2-way interactive video capability to the distant site. The laboratories are scheduled in a computer laboratory (the students use X-terminals or workstations) which is supervised by a Graduate Teaching Assistant (GTA) who is assigned full time (20 hours per week) to this class.

The lectures compress material from the introductory courses that they replace. For example, when discussing Graph Theory during the "regular" breadth-first undergraduate courses we will allocate enough lectures to cover the major algorithms and techniques in some detail. In the accelerated courses, however, only one week (two lectures) is assigned to the topic, and the lectures just cover a few of the algorithms and algorithmic techniques. The students are then given reading assignments from Gersting [3] and will be responsible for the additional material in tests and assignments.

When the students have problems with the material from the assigned readings, they have three ways to get help.

1. Other Students: This course only succeeds if the students work together on their assignments, and they are expected to help each other. Since the students come from very diverse backgrounds, they bring different problem solving skills and weaknesses to bear on the problem. E.g., if the students are given an assignment to read Section 5.3 of Gersting [3] on certain graph algorithms, then they will also be given an assignment to apply Warshall's algorithm to a specific adjacency matrix. They will be expected to do this working in groups. Most of these student run mutual aid sessions which will begin during the scheduled three hour laboratories, and will then continue when groups of students can get together outside of scheduled class or laboratory times.
2. GTA: The GTA is available during the scheduled laboratory times, and also during his office hours. He will help groups with any problems, including with additional reading assignments.
3. Instructors: The three instructors each have five to six office hours every week, and also have an open door policy where they can often be found at other times during the week. They are also available for much of the week through e-mail.

Required texts

The two accelerated courses effectively cover the material from seven other courses, and so we require four texts, that are used throughout the two courses. They are books on Unix [9], algorithmics [4], mathematical structures [3], and C++ [2].

Year 1	Fall	CS 252 Math 181	Accel. Intro. to CS I Calculus I	5 4
	Spr.	CS 254	Accel. Intro. to CS II	5
		CS 350 CS 355	Thy. of Computation Design of Prog. Lang.	3 3
	Sum.	Math 182	Calculus II	4
Year 2	Fall	CS 425	Computer Graphics	3
		CS 418	Operating Systems	3
		CS 440	Networks	4
		Math 221	Matrices	3
	Spr.	CS 331	Architecture	3
		CS 435	Database Systems	3
		CS 450	Compilers	4
	Sum.	CS 575	Master's Project	4
Year 3	Fall	CS 430	Image Processing	3
		CS 436	Artificial Intelligence	3
		CS 500	Seminar	1
		CS 510	Computability	3
	Spr.	CS 500	Seminar	1
		CS 525	Graphics & Sci. Vis.	3
		CS 536	Advanced AI	3
		CS 580	Comput. Complexity	3

Table 2: A Typical Three-Year Schedule

The course syllabus (maintained on the course Web page) gives detailed reading assignments to be completed before each class. Since the students in this sequence are highly motivated, they usually complete their assigned reading assignments on schedule.

Example Schedule

In Table 2 we show a typical Master's degree schedule for a student who has been evaluated as having the maximum number of undergraduate deficiencies. I.e., they have no significant previous formal training in Computer Science, and they also need to take the Calculus and Matrices courses from the Mathematics Department.

This schedule is not intended to be a template for students to follow, but is just intended as an example schedule to show that it is possible for these students to graduate in three years if they can take approximately 10 credits per semester.

In the table the undergraduate deficiency courses are shown in regular type, and the graduate courses are shown in bold type.

Students taking a schedule like this will normally be required to enter the program under the "second degree" classification at Montana State University. They will then apply for graduate school during the Spring Semester of their first year, and if they are accepted they will enter graduate school and be eligible for GTA/GRA positions then.

Distance Learning

In Fall Semester, 1996, the department was awarded a contract by the Malmstrom Air Force Base to deliver its MS degree to their site located 185 miles from Bozeman in Great Falls, Montana.

There are two groups of students applying for the degree program at this distant site, officers from the Malmstrom AFB and civilians from the base and from the town. Typically, the Air Force officers have engineering degrees, and the civilians have degrees from a wide range of different disciplines (usually non-technical).

The Air Force officers are assigned to the base for about 3.5 years, and so without the accelerated course structure they would be unable to complete their degrees before they are reassigned. These officers have all completed the required Mathematics courses as part of their previous degrees, and so to graduate they need to take CS 252 and 254, 20 other credits of undergraduate deficiencies, and 30 credits of graduate level coursework.

The civilians applying for the program have a wide range of previous degrees, and an even wider range of work experiences. E.g., one of our new students has been training cutting horses and shoeing horses for 15 years since completing an Animal Sciences degree.

For the last two years we have heavily relied on the use of the Web for online class syllabi and other materials. With the distance learning component, this has become even more important. We use the class Web page for all class handouts and laboratory assignments, in addition to the more traditional uses. For student questions, assignment submission, etc., we use e-mail. The instructors and the GTA all regularly monitor their e-mail during their office hours, scheduled laboratory times, and at other times agreed on with the distant students.

We anticipate that nearly all of our distance learning students will select the project option for their Master's program because of the difficulty of completing a thesis 185 miles from their advisor.

Results

The major positive and negative results that we have found with the accelerated sequence are listed below. It has been very successful, since we have found many significant benefits for both the students and the department, and have found no significant problems.

Positive results:

1. **Earlier graduation:** The original reason for developing this course sequence was to provide courses that would let highly motivated students with non-CS degree backgrounds graduate earlier, without sacrificing quality. This has been successful, as most of these students now

complete their Master's degrees in three years. What we had not anticipated was that these courses would also provide a number of other benefits to the department, as well as to the students. These additional benefits are summarized in the points numbered 2 through 9, below.

2. **Source of students for our MS program:** Before the establishment of this course sequence, our Master's degree program was somewhat smaller than we wanted, and we were relying on foreign students (primarily from China and India) to fill out our numbers. Nearly all of the students in CS 252 and 254 are Americans, and a high percentage of them continue into the MS degree. As a direct result of their recruitment through these courses, all of our new Graduate Teaching Assistants this year are Americans, and our graduate enrollment has reached the levels that we wanted to achieve.
3. **Team skills:** Although we emphasize team skills in the regular introductory courses and software engineering courses in Computer Science at Montana State University [6, 7] the group dynamics that appear in this small, highly motivated, group of students leads to the development of excellent team skills.
4. **Meets the needs of mature students:** The students in CS 252 and 254 are very different from our traditional freshman, and so the courses designed for traditional students do not work well for them. Small intensive classes like CS 252 and CS 254 are much more appropriate for them.
5. **Student morale:** Students enjoy this course. The usual introductory courses are large and impersonal, whereas in these courses there are a small number of very diverse students who work together and become friends. This, obviously, motivates them to succeed in these classes and to continue on into other classes and through the MS degree.
6. **Developing new opportunities for returning students:** The primary motivation for most students in the accelerated classes is to improve their job prospects. In some cases this is to improve their chances of advancement in their current jobs (e.g., the Air Force officers at Malmstrom AFB taking courses under our distance learning program greatly enhance their future prospects with a graduate degree in Computer Science). In other cases the students want to be able to leave their current positions and move into the computer industry.
7. **Faculty morale:** These are very enjoyable courses for the faculty to teach. In many ways it is similar to a graduate course, but without some of the pressures associated with teaching at the graduate level, where one must keep up with all of the latest work in the field, as it happens. I.e., although most of the material covered is basic Computer Science, one has small classes with

mature, demanding, students who interact well during class, and are fun to teach.

8. **Facilitate distance learning:** It is expensive to teach deficiency courses through two-way interactive video into distant sites. Costs include the hardware at each site, the transmission lines, and the equipment operators at each site. CS 252 and 254 replace 26 credits of coursework that would otherwise have to be taken. If one ignores the laboratory components, this means that four credits need to be transmitted over the video system, instead of 20, a savings of 16 credits. At our \$100/credit distance learning fee, this saves each student \$1600, as well as greatly reducing the effort that needs to be made by departmental faculty.
9. **Team teaching:** This can be both positive and negative, and so we have listed it in both categories. The advantages are that it provides diversity in the class, encourages faculty to work closely together, and also provides three times as many office hours when the faculty can be reached by students (the authors have a total of 16 hours).

Negative results:

1. **Production formulas:** The courses are considered to be "very expensive" by university administrators. E.g., the Dean of our College of Engineering [1] considers a 200-level course, with two to three faculty partially assigned to it and one GTA fully assigned to it, to be of questionable value when there are only about 10 to 15 students in the class.
2. **Intransigent students:** Some students insist on doing most of the work by themselves, and not in teams with others. These students usually do significantly worse in the course than the others.
3. **High initial cost:** The initial cost of textbook purchases is high, and this causes problems for some of our students. The cost is, however, less than for the total regular introductory course sequence, since
 - in the regular sequence we start with programming in Ada, and then move later to C and C++, but in the accelerated sequence we begin with C++.
 - in the regular sequence we require a book of readings in social and ethical issues [5], but in the accelerated sequence we use case studies developed by the instructors.
 - in the Software Engineering course, CS 351, we use a textbook [8], but in the accelerated sequence we use notes developed by the instructors.
4. **Requirement for an excellent GTA:** The Graduate Teaching Assistant assigned to the course must be one

of our best GTA's. He/she is required to help the students over a wide range of material, and is also primarily responsible for the laboratories.

5. **Need for team teaching:** We find that we need to use team teaching for this course, because of the diversity of the material that is covered. Team teaching is not an efficient use of faculty resources (but it is an effective use).

References

1. Adams, S. *The Dilbert Principle*. Harper Collins, 1996.
2. Deitel, H.M. and Deitel P.J. *C++: How to Program*. Prentice Hall, 1994.
3. Gersting, J.L. *Mathematical Structures for Computer Science*. Computer Science Press, 1993.
4. Harel, D.H., *Algorithmics: The Spirit of Computing*. Addison-Wesley, 1992.
5. Johnson, D.G. and Nissenbaum, H., *Computer Ethics & Social Values*. Prentice Hall, 1995.
6. Paxton, J.T., Ross, R.J., and Starkey, J.D. An Integrated Breadth-First Computer Science Curriculum Based on Computing Curricula 1991. *Proceedings of the SIGCSE 93 Technical Symposium*, February 1993. Also in *SIGCSE Bulletin* 25, 1, pp. 68-72, March 1993.
7. Paxton, J.T., Ross, R.J., and Starkey, J.D. A Methodology for Teaching an Integrated Breadth-First Computer Science Curriculum. *Proceedings of the SIGCSE 94 Technical Symposium*, March 1994. Also in *SIGCSE Bulletin* 26, 1, March 1994.
8. Schach, S.R., *Classical & Object-Oriented Software Engineering*. Irwin, 1996.
9. Sobell, M.G., *A Practical Guide to the Unix System*. Benjamin Cummings, 1995.

Appendix: CS 252 and CS 254 Catalog Descriptions

CS 252 ACCELERATED INTRODUCTION TO COMPUTER SCIENCE I

Fall, 5 cr. LEC 2, LAB 3

PREREQUISITE: Degree in another discipline.

This course is intended for graduate students who plan to enter the Computer Science graduate program or need an accelerated computer science course to support their graduate program and whose undergraduate background is outside of Computer Science. Covers the material in CS 160, CS 221, CS 222, and part of CS 351.

CS 254 ACCELERATED INTRODUCTION TO COMPUTER SCIENCE II

Spring, 5 cr. LEC 2, LAB 3

PREREQUISITE: CS 252.

This course is intended for graduate students who plan to enter the Computer Science graduate program or need an accelerated Computer Science course to support their graduate program and whose undergraduate background is outside of Computer Science. Covers the material in CS 210, CS 223, CS 324, and part of CS 351.