21st Century Computing Education for South Carolina

Recommendations for Change
The Computing Competitiveness Council
Summer 2007

Computer science is in crisis in South Carolina. It is entirely possible that computer science is no more in crisis here than it is nationally, but that does not change the fact that computer science is in crisis. Our analysis here is not much different from that done by the Computer Science Teachers Association [CSTAa]; this means that if we can find solutions locally to problems that are national problems, we might actually have an edge on the rest of the country.

As in the paragraph above, we will refer most of the time to "computer science." Both Clemson and USC (and other universities in South Carolina) have a "computer information systems" undergraduate degree program as well as a "computer science" program. We will for simplicity refer only to "computer science," in part because the customary "computer information systems" program is a computer science major coupled with what amounts to a minor sequence of courses in business. We do not intend this proposal to refer to "computer engineering" for two reasons. First, computer engineering is more often *not* mixed with computer science (it is mixed in South Carolina only at USC). Second, the job market in computer engineering is decidedly limited (resembling engineering in general) by comparison with computer science and students with an interest in and ability to pursue careers in technical fields might well be advised away from engineering and from other sciences and into computer science due to the wide difference in the job prospects between computer science and most of the rest of the STEM (Science, Technology, Engineering, and Mathematics) disciplines.

We would also like to emphasize that we believe the current situation to be largely the result of sins of omission, not sins of commission. As detailed in the CSTA report, there is substantial confusion regarding the lines of demarcation of computer science, computer literacy, computer applications, and other instances in which software is used in support of some other instructional purpose. Although we do argue in favor of a curricular shift as regards computer science, we realize that this would not only take time, it would also depend on the availability of teachers and of students to teach and of the practicalities of budgets.

What we argue most strongly for here is that the nature of the world of computing be explained clearly to students, parents, teachers, and administrators. This "truth in advertising" of the lines of demarcation mentioned above will allow students to determine if their interests mesh with one or more of the academic areas that involve computing and if so what curricular and career path will optimize their chance of success.

Two factors must also be recognized that contribute to a misunderstanding of computer science (taken in the broad sense). The first is that as a discipline and as a career path it does not fit neatly into any of the standard academic categories. Computer science programs are housed in universities in colleges of science or arts and science, colleges that contain engineering, stand-alone colleges of computing, and in colleges of business. Departmental undergraduate programs are sometimes combined with mathematics and occasionally with business or with electrical engineering. It is not necessarily "science" in the sense that chemistry and physics are science. (Nobel Prize winner Herb Simon once referred to it as an "artificial" science in contrast with the "natural" sciences, since it deals with the artifice that is a computer and not really with the natural world.) It also is not "engineering." And, although by far the majority of jobs are in business applications, the technical expertise necessary for success even in business applications is substantially beyond that required of the average student in a business college.

The second factor to be considered is the common conflation of job prospects for holders of two-year and of four-year degrees. Contributing to the confusion caused by this conflation is that the abundance of jobs and the scarcity of talent to be hired have caused people without formal credentials to be hired in vast numbers and then trained on the job. There has never been an adequate supply of bachelor's degree holders in computer science, so statistics on the qualifications of people hired into various job titles is misleading. What would clear up many questions would be a good analysis of the qualifications that would be used in hiring decisions if it were ever the case that employers had an adequate supply of candidates.

Terminology: We believe part of the problem with computing education stems from a misuse or misunderstanding of terms. We offer in Appendix A a glossary of terms. Most of the definitions have been taken from Wikipedia; although this source is not authoritative, we believe that the definitions and explanations there are reasonably correct, and the discussion and links provided through Wikipedia allow for a reasonable understanding of the terms and the differences among them.

We will refer in this document to the K-12 school system as the "schools." We will refer to the technical colleges and the four year colleges and universities as the "colleges."

Background data: We present, in Appendices B, C, and D, data that shows some of the anomalous situation of computer science with regard to other STEM disciplines. From Appendix B it can be inferred that preparation in computer science leads to employment in the computing professions to an extent not seen in other STEM fields and that shortages of educated computing professionals have led to an influx of people with other background preparation. Appendix C has the Bureau of Labor Statistics data on job projections from 2004 to 2014.

Appendix D shows the decline in computer science enrollments in South Carolina since 2000.

The **Computing Competitiveness Council** is a newly formed organization of interested professionals, educators, parents, and business interests for whom a viable knowledge economy in South Carolina is important and who see improvements in computing education and its enrollments as vital to the state's economic growth. It was initially formed with the participation of department chairs or their designees from the "computer science" departments of the major institutions of higher education offering such degrees in South Carolina, parents, and high school teachers. Its plan is to involve the business community in the state to improve the climate for growth in information technology, telecommunications, software, and related industries that rely on computing as a core requirement for business success.

An Analysis of the Situation

In the following paragraphs we outline some of the issues that we believe contribute to the problem of computer science being misunderstood. We believe this misunderstanding may well be common among students, parents, teachers outside the discipline, counselors, and school administrators, and education administrators at the state level. We do not intend to offend in our analysis. We do not believe that the current situation is the result of a conscious effort to create a bad situation. Rather, we suspect that the current problem in the K-12 system is the result of growth and expansion in a new and changing discipline, but growth that has come without enough coordination to ensure that a coherent message is presented. At the technical college level, we believe the problems stem from the inherent dual nature of technical college offerings and an understandable bent toward satisfying, in an inevitable situation of limited resources, the need for skills training in associate degree and certificate programs.

Our overall proposal, then, is not to try to overhaul the entire K-12 and technical college systems, but to provide information and coordination to those systems. At the technical college level, students are expected to make choices between skills courses that provide a benefit but will not transfer to a four-year computer science program and general education and introductory courses that will translate to a four-year curriculum. We believe the message about those choices could perhaps be made more clear. At the K-12 level, students and parents need to understand the difference between “computer science” courses and courses that use computers but in fact are not computer science; students who eschew computer science because they think it is the same as computer applications are making as wrong a decision as those who begin a program in computer science thinking that it involves only the *use* of software packages.

The K-12 System

The general situation is that computer science education in South Carolina K-12 system and extending into the technical colleges, is disorganized, fragmented and lacks coherence. We do not believe this has been intentional; it has just happened, for many of the reasons described in the CSTA document cited above; but there are factors unique to South Carolina.

Most obvious is that a state law mandates one unit of “computer science, including keyboarding” for all students graduating from high school. Almost none of the courses that satisfy the requirement, however, are in fact “computer science” as defined by the discipline (academic or industrial), as can be seen by examining the standard computer science curriculum on which accreditation is based. We believe this contributes to a serious confusion and marketing misrepresentation of the discipline and its career paths. Allowing keyboarding to satisfy a “computer science” requirement, for example, is not much different from allowing one to satisfy a “mathematics” requirement with a course called “Cash Register Operation”.

To compound the K-12 confusion, there are "computer science" courses (always Advanced Placement?) taught largely by math teachers, and "computer programming" courses taught from the business program in the K-12 system. These are the only courses that satisfy the state computer science requirement that are in fact computer science; all the other courses should be labeled as "computer applications" or something similar.

We believe that much of this mislabeling comes from the ubiquitous nature of computing in our society and today's professional expectations. Computing permeates virtually all activities and it is certain that every discipline requires a significant amount of computing literacy for its members to be successful. Having said that, it is important to recognize that such skills do not constitute computer science and we must of necessity be careful with our use of the terms lest we mislead students into educational paths for which they are not prepared or have significant interest once they discover what is required by the computer science discipline. We see this phenomenon in the colleges every fall – from each entering freshman class we hear the same litany of false expectations.

Teacher certification

The certification of teachers for teaching computer science is similarly disorganized. There are apparently several different paths to endorsements and certification so that one can teach a "computer" course. Since most of the courses are taught from the business program, that's where most of the teachers come from. But even so the critical element to be fostered is the ability to solve problems within the machine environment. This is the heart of computer science and our K-12 teachers should be certified to work within that construct.

We point out that South Carolina is by no means exceptional in the current situation. The CSTA has just published a database [CSTAb] of the state certification requirements to teach computer science, and an article has just appeared [Whitehurst], written by a Clemson graduate in computer science, of the difficulties in getting certified in North Carolina to teach computer science even with a bachelor's degree in the discipline.

The logistical "problem" with the technical college curriculum

The technical colleges have a dual mission, on the one hand to be highly job-related, and on the other hand to provide the first two years of a university education to those students who will start in the technical college system and then transfer to a four-year program. Midlands Tech, in response, has an Information Science and Technology department that teaches a suite of courses that are designed to provide students, within a two-year time-frame, with the programming and other skills needed for entry-level employment, rather than for further study in computer science. However, none of these courses have calculus as a prerequisite and thus none of these courses transfers for credit into the computer science major at USC. The situation is similar with Tri-County Tech and Clemson, and we suspect this to be true of the other technical colleges in the state and the other accredited computer science programs in the universities. To meet the ABET accreditation requirements for computer science undergraduate programs, nearly all universities will require calculus as at least a co-requisite to the first year sequence and a prerequisite for most courses at the sophomore level and above.

It must be remembered, however, that there are more jobs available (in number) for those who hold associate degrees and certifications than for those who hold bachelor's degrees. What we have, then, is a problem caused by an abundance of riches—the 2-year-degree job market is so good that it is too easy to spend all one's efforts there and thus to overlook the 4-year degree options.

Once again, it is not our intent to denigrate the technical colleges. Their dual mission produces an inherent conflict for all professions in which there is a significant demand for skilled people both at the two-year and the four-year level. Just as with the high school coursework, it is probably the abundance (and not the lack) of options that needs to be explained better to students. Those who do not intend to continue to a four-year degree need not be discouraged from their curricular path, but those who have plans to continue need to get the best possible advice at the earliest possible stage of their academic careers.

The marketing of computer science and the job opportunities

Based on admittedly limited anecdotal evidence, it is entirely possible that guidance counselors are not being kept up to date on the job market in computer science. We do not necessarily fault the counselors for this. There is, as we argue here, a systemic problem of computer science being "someone else's responsibility," and it is not unreasonable to assume that this is as true for counselors as is for those involved in the curriculum.

We present in Appendix C the excerpted job projections from the U.S. Bureau of Labor Statistics [Hecker] for "computer specialists" (category 15-1000) as well as for life and physical scientists (categories 19-1000 and 19-2000) and for engineers (category 17-2000). The projections to 2016 should be published in the fall of 2007, and we expect them to present an even more optimistic picture for computing. As a summary statement, we note [Hecker, page 72] that nearly one in six "professional" jobs to be filled in the period 2004-2014 will be in computing, with computing having the largest growth (30%) of any of the professional occupational groups.

The Way Ahead?

Action Items for the CCC

A summarized version of the action plan proposed by CCC is the following.

1. A plan for correcting the misperceptions about the job market in computing and about the nature of jobs in computing, including information for parents and students about what the educational paths are to the various kinds of careers that involve computing.
2. Involvement by the business community to ensure that South Carolina's students learn what is needed to succeed in jobs and careers that involve computing.
3. Gradual changes in the schools and at SCSDE so that computing education is organized, including the preparation of teachers, more in concert with the paths in higher education and in the working world.
4. Curricular initiatives to help provide all schools with resources for teaching real computer science.

The job market in all areas of computing is wide open, and there are relatively few adults in this century who would not benefit from an increased familiarity with computing. The real agenda of the CCC is to increase enrollments in computer science in the colleges, but a general view of the path we have chosen to reach this goal is to increase an awareness of all levels of sophistication of what computing is all about.

1. Correct the Misperceptions

Public Relations: At the heart of the problem is the perception of a poor job market in computing. The CCC will be mounting an aggressive campaign to correct this misperception. This will be directed primarily at students and parents, but apparently there is also a need to ensure that teachers, school administrators, and guidance counselors understand the opportunities available in computing and communicate these to students and parents. Part of the public relations campaign (see also below under "Changes in the Schools") is to improve the understanding of the nature of computing in the general populace. Increasing the understanding of computing in the general population cannot help but translate to better decision-making on the part of students about careers.

Information for Students and Parents: A primary effort of CCC will be to produce a "consumer reports" guidebook for students in the schools and their parents on how to negotiate the curricular offerings of the schools and the colleges that might lead to careers in computing. The multiplicity of courses and of options in the schools and technical colleges is not in itself a bad thing. But students need to know that "computer science" is not the same as the use of software packages. They also need to know that there is no real barrier to beginning a computer science degree at the college level. In a state in which very few schools teach Advanced Placement classes in computer science and many do not teach programming classes, it is important for students to know that they do not need high school preparation in order to study computer science later.

Certainly any related courses they might take in school would contribute to an understanding of computing, but they are not required.

Information for Technical College Students: It is certainly not going to be possible to redirect the technical colleges away from their mandated focus on job training skills, and there is no reason to do this. However, CCC proposes to work with the technical colleges to ensure that students who start their postsecondary education there can, if they choose, take an academic program that tracks into four-year degrees at the state's universities. Not all the courses in the technical colleges will transfer to the four-year universities, but much of the first two years of a computer science bachelor's degree—the general education courses, the mathematics and science requirements, and the courses in the major (if students exist in sufficient number to justify the courses' being taught)--can be obtained at the technical colleges. A student's judicious choice of courses can make the transition as smooth as possible, and CCC proposes making sure that students know how to make that judicious choice of courses.

2. Business Involvement

Nothing that CCC does will make sense if it is not relevant to business and to the general economic development of the state. CCC will endeavor to work closely with the business community in this regard. To a great extent, this can be facilitated through the colleges, since nearly all computer science programs will have an Industrial Advisory Board (by that or some name) that can be called upon. Leadership in coordinating interactions with the business community can be expected to follow to some extent regional divisions, with USC Columbia leading in the Midlands and, with Winthrop, in Rock Hill, USC Upstate and Clemson in Greenville-Spartanburg, and College of Charleston in the Low Country.

3. Changes in the Schools and at the SCSDE

We believe that a number of changes are warranted at the schools level. We would like to emphasize that we do not intend this to be a criticism of what was intended to be implemented at the schools level. Rather, we suspect that the problem is in part due to the ubiquitous nature of computing and to the limited resources available to the school system. One cannot fault a dedicated teacher for creating an exciting computer applications class that tracks into job skills that are appreciated by local employers. We suspect, however, that this has happened a number of times, without a great deal of overall supervision of the curriculum in computing as a curriculum in computing *per se*. The CCC would like to offer some of that supervision, with direction coming both from a coordinated effort of the colleges and from the business interests in South Carolina.

Guidance counselors: It is absolutely imperative that guidance counselors know that the national job market in computer science is vastly more optimistic than the market for almost any other set of career paths. We have anecdotal evidence from one of the largest and best high schools in the state that guidance counselors are advising students *away* from computer science and into STEM areas with substantially more moderate job prospects. This simply does not make sense.

Note: The new *Pathways* brochure for science (which we have not yet seen) should help in this regard and we are very pleased that SCSDE has made changes.

A Clarification of the "Computer Science" Requirement: The required one unit of "computer science" in the schools can be met by a variety of courses offered through a variety of labels in the schools, almost none of which has a strong relationship to computer science. The courses themselves, listed in an appendix, include courses labeled computer science and computer science courses that are labeled computer programming, together with courses in computer literacy, the use of software tools and applications, and some technology certification courses.

The requirement itself should be clarified. The law as written says "computer science," but the SCSDE website also includes a number of documents that say "computer literacy." Although CCC believes that there should be a true "computer science" requirement in the schools, which at present would only be met by the AP computer science and the computer programming classes, we suspect that the law itself is in error and that computer literacy was intended. One way or another, this issue should be clarified.

On the other hand, one must guard against an overemphasis that "programming" and "computer science" are the same thing. Programming is fundamental to computer science. The rigorous analytical and algorithmic thinking that accompanies the actual programming process is crucial to success in computer science. Programming *per se*, however, is not what most computer scientists pursue as a lifetime career. Indeed, the job predictions are for only a 2% growth in the "computer programming" job title; these are the jobs that are being sent overseas. At the core of any computing application there is software, and computer scientists need to be capable of writing software. As a career progresses, however, most computer scientists will enlarge the scope of their concern beyond the actual programming, but still closely connected to the management and orderly flow of information through a computational process. We suspect that these distinctions are not thoroughly understood by the teachers and guidance counselors and that students and parents are thus not as well informed as they could be. And all too often, 2-year-degree job opportunities and their salaries are mixed in with those for holders of 4-year-degrees; great care must be taken to ensure that these different categories are disaggregated.

Re-organizing Computer Science Certification and Curricula: The current situation is that both certification and curriculum seem disorganized and fragmented. Given budget issues, it is no doubt impossible to make any abrupt major changes; changes will be possible only as enrollments rise in relevant and core courses and as teachers gravitate toward those relevant courses with robust enrollments.

However, CCC proposes working with the SCSDE to produce a cover document that explains the certification and the curriculum. We also propose with this that a focus be made on computer science *per se* and not on computer-related courses and certification as optional add-ons to other plans. In essence, we would like to re-organize computing education in the schools to reflect how the discipline is organized in the colleges and how the graduates are perceived by future employers. We believe that the state would be well served by a single document unifying computer science as students and teachers ought to see it.

The CCC also supports a specific SCSDE initiative to support computer science in the schools, much as Project Lead The Way is supported for engineering education. We believe that the demands of the knowledge economy and of economic development in South Carolina, coupled with the shocking declines in enrollment in these critical programs, demands a formal response from the educational system in both the schools and the colleges.

CCC also recognizes the importance of the ongoing education of teachers, and we will work to design appropriate coursework to facilitate recertification. In a field that changes as rapidly as does computer science, this would seem essential for maintaining a modern curriculum in the schools, but once again the problems of distance and dispersion in rural communities can be mitigated by the use of technology for delivery of the coursework.

4. Curricular Initiatives by CCC

Advanced technology education in South Carolina is unquestionably hampered by the disparity between rural and urban school districts and between the affluent and the constrained school systems. If there is any one advantage held by computer science, however, it is that we can take advantage of the technology in order to teach the technology. CCC proposes, and is willing to embark on the development of, introductory computer science courses appropriate for distance delivery to the schools. It has been observed that one of the major public relations problems facing computer science is the fact that an introductory programming course seems essential, and yet programming is only the starting and not the ending of a computer science education. The colleges are undertaking several initiatives to increase enrollments by emphasizing that which computer scientists do that is not in fact programming. These initiatives can be recorded as videos for playback in the schools. An additional course could be developed in modules, explaining the career paths in computing (business information systems, computational science, network administration, computer security, media and animation, etc.). By providing these materials, the professionals in business and higher education could offer the students in the schools the opportunity for expanded horizons.

The Bottom Line

We cannot hope in the short term to completely re-work the curriculum and certification in the K-12 system. We can, however, take steps to provide guidance to all parties—teachers, parents, students, counselors, the technical college system, and the Department of Education—that defines computer science, describes the options, career paths, and sub-disciplines, and correctly categorizes those options. We should view this as a "cheat sheet". The goal is to prepare students to fill those million new jobs in computing and to get the best of them to go on to graduate school. We cannot immediately re-work the system nearer to the heart's desire, but we can provide guidance on navigating through the system so that the end result is success.

References

CSTAa: CSTA, *The New Educational Imperative: Improving high school computer science education*, 2006, ISBN 1-59593-335-2.

CSTAb: http://www.csta.acm.org/ComputerScienceTeacherCertification/sub/TeacherCertificationRequi.html, last referenced 7 June 2007.

Hecker: Daniel E. Hecker, "Occupational employment projections to 2014," *Monthly Labor Review*, November 2005, pp. 70-101.

Whitehurst: Jane S. Whitehurst, "Over-qualified, under-certified," *CSTA Voice*, Volume 3, Number 1, June 2007, pp. 2-3.

Appendix A
A Glossary of Terms

(Unless otherwise noted, all these definitions are quoted directly from Wikipedia.)

Computer science, or **computing science**, is the study of the theoretical foundations of information and computation and their implementation and application in computer systems.[1][2][3] Computer science has many sub-fields; some emphasize the computation of specific results (such as computer graphics), while others relate to properties of computational problems (such as computational complexity theory). Still others focus on the challenges in implementing computations. For example, programming language theory studies approaches to describing computations, while computer programming applies specific programming languages to solve specific computational problems with solutions. A further subfield, human-computer interaction, focuses on the challenges in making computers and computations useful, usable and universally accessible to people.

Computer literacy is the knowledge and ability to use computers and technology efficiently. Computer literacy can also refer to the comfort level someone has with using computer programs and other applications that are associated with computers. Another valuable component of computer literacy is knowing how computers work and operate. As of 2005, having basic computer skills is a significant asset in the developed countries.

The precise definition of "computer literacy" can vary from group to group. Generally, literate (in the realm of books) connotes one who can read any arbitrary book in their native language[s], looking up new words as they are exposed to them. Likewise, an experienced computer professional may consider the ability to self-teach (i.e. to learn arbitrary new programs or tasks as they are encountered) to be central to computer literacy. In common discourse, however, "computer literate" often connotes little more than the ability to use several very specific applications (usually Microsoft Word, Microsoft Internet Explorer, and Microsoft Outlook) for certain very well-defined simple tasks, largely by rote. (This is analogous to a child claiming that they "can read" because they have rote-memorized several small children's books. Real problems can arise when such a "computer literate" person encounters a new program for the first time, and large degrees of "hand-holding" will likely be required.) Being "literate" and "functional" are generally taken to mean the same thing.

Computer fluency is a term of very recent origin (2004 or later?), stemming from a national initiative on computing education and used as the title of a book proposed for high school or university courses. Fluency is an understanding of the way computers work that can only come from using, exploring and experimenting with them. Fluent people do not necessarily "know" more about computers than anybody else, they just take a different approach. They are willing to try and explore until they get the result they desire. Computer fluency is more than just knowing "how to use" a computer. Many people who have taken all sorts of "computer literacy" classes are completely helpless when something happens that they have not been trained to deal with, because they have not cultivated fluency. (This definition is from http://www.connectedfamily.com/frame4/cf0411hotword/words/cf0411acompflu.html).

Computer applications, or application software, is a subclass of computer software that employs the capabilities of a computer directly to a task that the user wishes to perform. This should be contrasted with system software which is involved in integrating a computer's various capabilities, but typically does not directly apply them in the performance of tasks that benefit the user. In this context the term application refers to both the *application software* and its implementation.

A simple, if imperfect, analogy in the world of hardware would be the relationship of an electric light—an application—to an electric power generation plant — the system. The power plant merely generates electricity, itself not really of any use until harnessed to an application like the electric light which performs a service that the user desires.

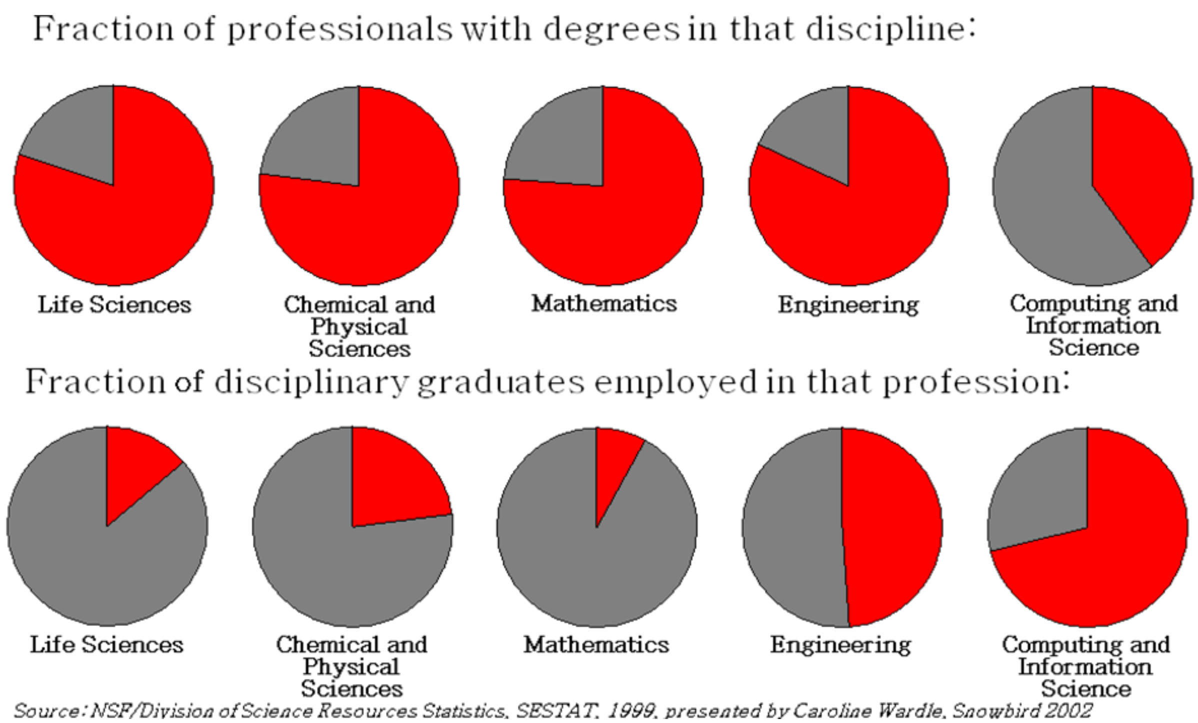
The exact delineation between the operating system and application software is not precise, however, and is occasionally subject to controversy. For example, one of the key questions in the United States v. Microsoft antitrust trial was whether Microsoft's Internet Explorer web browser was part of its Windows operating system or a separable piece of application software. As another example, the GNU/Linux naming controversy is, in part, due to disagreement about the relationship between the Linux kernel and the Linux operating system.

Typical examples of **software applications** are word processors, spreadsheets, and media players.

Computer technology usually refers to the hardware technology used in computers, computer networks, and peripheral devices like printers, monitors and display devices, and such. Academic coursework labeled “computer technology” usually refers to the maintenance and repair of this equipment, not the use of the equipment.

Appendix B
Employment versus degrees in STEM fields

The following chart comes from a National Science Foundation study. The upper row of pie charts indicates the fraction of professionals (in the darker red color) working in a particular discipline who have degrees in that discipline, and the lower row indicates the fraction of individuals (in the darker red color) with degrees in a particular discipline who actually work in that discipline. What is clear from this is that computing professionals tend to work in computing (because they can?) and that professionals in computing often do not have formal degrees in computing (because companies can't ever find enough people with formal degree preparation?)



Appendix C
Bureau of Labor Statistics Job Projections 2004-2014

Table 2 has the projections for the fastest growing occupations. Of the ten for which a bachelor's degree is the normal requirement, six, highlighted in yellow, would normally require a bachelor's degree in computer science. The projected 794,000 new jobs in these occupations constitute 88% of the projected new jobs in these ten occupational titles.

Table 2. Fastest-growing occupations, 2004–14

[Numbers in thousands]

2004 National Employment Matrix code and title	Employment		Change		Quartile rank by 2004 median annual earnings[1]	Most significant source of postsecondary education or training[2]
	2004	2014	Number	Percent		
31–1011 Home health aides	624	974	350	56.0	VL	Short-term on-the-job training
15–1081 Network systems and data communications analysts	231	357	126	54.6	VH	Bachelor's degree
31–9092 Medical assistants	387	589	202	52.1	L	Moderate-term on-the-job training
29–1071 Physician assistants	62	93	31	49.6	VH	Bachelor's degree
15–1031 Computer software engineers, applications	460	682	222	48.4	VH	Bachelor's degree
31–2021 Physical therapist assistants	59	85	26	44.2	H	Associate degree
29–2021 Dental hygienists	158	226	68	43.3	VH	Associate degree
15–1032 Computer software engineers, systems software	340	486	146	43.0	VH	Bachelor's degree
31–9091 Dental assistants	267	382	114	42.7	L	Moderate-term on-the-job training
39–9021 Personal and home care aides	701	988	287	41.0	VL	Short-term on-the-job training
15–1071 Network and computer systems administrators	278	385	107	38.4	VH	Bachelor's degree
15–1061 Database administrators	104	144	40	38.2	VH	Bachelor's degree
29–1123 Physical therapists	155	211	57	36.7	VH	Master's degree
19–4092 Forensic science technicians	10	13	4	36.4	VH	Associate degree
29–2056 Veterinary technologists and technicians	60	81	21	35.3	L	Associate degree
29–2032 Diagnostic medical sonographers	42	57	15	34.8	VH	Associate degree
31–2022 Physical therapist aides	43	57	15	34.4	L	Short-term on-the-job training
31–2011 Occupational therapist assistants	21	29	7	34.1	H	Associate degree
19–1042 Medical scientists, except epidemiologists	72	97	25	34.1	VH	Doctoral degree
29–1122 Occupational therapists	92	123	31	33.6	VH	Master's degree
25–2011 Preschool teachers, except special education	431	573	143	33.1	L	Postsecondary vocational award
29–2031 Cardiovascular technologists and technicians	45	60	15	32.6	H	Associate degree
25–1000 Postsecondary teachers	1,628	2,153	524	32.2	VH	Doctoral degree
19–2043 Hydrologists.	8	11	3	31.6	VH	Master's degree
15–1051 Computer systems analysts	487	640	153	31.4	VH	Bachelor's degree
47–4041 Hazardous materials removal workers	38	50	12	31.2	H	Moderate-term on-the-job training
17–2031 Biomedical engineers	10	13	3	30.7	VH	Bachelor's degree
13–1071 Employment, recruitment, and placement specialists	182	237	55	30.5	H	Bachelor's degree
17–2081 Environmental engineers	49	64	15	30.0	VH	Bachelor's degree
23–2011 Paralegals and legal assistants	224	291	67	29.7	H	Associate degree

[1] The quartile rankings of Occupational Employment Statistics Survey annual earnings data are presented in the following categories: VH = very high ($43,600 or more), H = high ($28,580 to $43,590), L = low ($20,190 to $28,570), and VL = very low (up to $20,180). The rankings were based on quartiles, with one-fourth of total employment defining each quartile. Earnings are for wage and salary workers.

[2] An occupation is placed into 1 of 11 categories that best describes the postsecondary education or training needed by most workers to become fully qualified in that occupation. For more information about the categories, see *Occupational Projections and Training Data*, 2004–05 edition, Bulletin 2572 (Bureau of Labor Statistics, March 2004) and *Occupational Projections and Training Data*, 2006–07 edition, Bulletin 2602 (Bureau of Labor Statistics, forthcoming).

This table excerpts the job projections in computing, engineering, and the life and physical sciences. This does not include the job projections in the health care professions, where one might expect a number of graduates in life sciences eventually to find employment. Note that the "computer programmer" category, with a projected growth of only 2%, stands out in the computing category. These are probably the jobs that are being offshored.

APPENDIX: Continued—Employment by occupation, 2004 and projected 2014

[Numbers in thousands]

2004 National Employment Matrix code and title		Employment				Change		Total job openings due to growth and net replacements, 2004–14[1]
		Number		Percent distribution		Number	Percent	
		2004	2014	2004	2014			
15–1000	Computer specialists	3,046	4,003	2.1	2.4	957	31.4	1,350
15–1011	Computer and information scientists, research	22	28	.0	.0	6	25.6	8
15–1021	Computer programmers	455	464	.3	.3	9	2.0	117
15–1030	Computer software engineers	800	1,169	.5	.7	369	46.1	448
15–1031	Computer software engineers, applications	460	682	.3	.4	222	48.4	268
15–1032	Computer software engineers, systems software	340	486	.2	.3	146	43.0	180
15–1041	Computer support specialists	518	638	.4	.4	119	23.0	183
15–1051	Computer systems analysts	487	640	.3	.4	153	31.4	208
15–1061	Database administrators	104	144	.1	.1	40	38.2	51
15–1071	Network and computer systems administrators	278	385	.2	.2	107	38.4	138
15–1081	Network systems and data communications analysts	231	357	.2	.2	126	54.6	153
15–1099	Computer specialists, all other	149	177	.1	.1	28	19.0	45
17–2000	Engineers	1,449	1,644	1.0	1.0	195	13.4	507
19–1000	Life scientists	232	280	.2	.2	48	20.8	103
19–2000	Physical scientists	250	281	.2	.2	30	12.2	94

Finally, the table below compares supply versus demand. The first two columns are taken from the Bureau of Labor Statistics data referenced immediately above, and the projections of B.S. production nationally come from the National Science Foundation and the National Center for Education Statistics.

	Growth Rate	Total Openings	B.S. Production
Computing	31.40%	1,350,000	574,000
Engineering	13.40%	507,000	647,000
Life sciences	20.80%	103,000	809,000
Physical sciences	12.20%	94,000	181,000

Appendix D
Computer Science Enrollments in South Carolina

The following table shows enrollments in the larger undergraduate programs in computer science at the state's universities, as reported to the Commission on Higher Education.

	Fall 2000	Fall2005	2005/2000
USC	405	180	0.4444
USC Upstate	129	93	0.7209
C of Charleston	129	78	0.6047
Clemson	414	262	0.6329
Coastal Carolina	170	149	0.8765
FMU	128	87	0.6797
Lander	97	57	0.5876
S C State	268	179	0.6679
Winthrop	99	69	0.6970

Appendix
Courses that meet the "computer science" requirement

The various types of courses have been categorized by CCC.

Computer Science (these seem to be the only courses that in fact *are* computer science):

Math—AP Computer Science
Information Technology—Computer Programming

Computer Applications:

Arts/A/V...--Graphic Communication
Business/Mktg Technology—Computer Applications 1
Desktop Publishing
Digital Input Technologies
Document Processing
Integrated Business Applications 1
Multimedia
Web Page Design and Development
Oracle
Animated Computer Production
Digital Imaging 1

Computer Technology:

Business/Mktg Technology—Intro to Computer Technology
Information Technology—Information Technology Foundations
Networking
Computer Service Technology

Non-computer courses:

Architecture and Construction—Drafting with CAD
Agriculture ...—Agri Production and Mgmt
Engr and Ind Tech—PLTW courses
Business/Mktg Technology—Business Computer Mathematics
Keyboarding

(It is entirely possible that some of these courses (Intro to Computer Technology and IT Foundations, for example) are actually more properly categorized as "computer literacy" courses similar to new South Carolina law requiring "financial literacy" such as how to apply for a credit card or balance a checkbook.)