

Teaching Computer Competencies to Today's Computer Age Students

Gregory K. Watkins, Nan A. Byars, Ambrose G. Barry

William States Lee College of Engineering
The University of North Carolina at Charlotte
Charlotte, NC 28223

Abstract

The Engineering Technology (ET) department at UNC Charlotte began offering the first two years of its BSET curriculum in the fall semester 2004, having previously been exclusively a “two plus two” program. Although much of the first two years includes basic studies in English, math, and physics, the department has chosen to teach its own freshman level class in computer competency.

The goal of ETGR 1100, Engineering Technology Computer Applications, is to provide basic computer competencies that will prepare freshman students for the remainder of their studies here at UNC Charlotte. The course covers the use of standard office applications in engineering, advanced use of scientific calculators, and also gives a sampling of specialized engineering software, such as Electronics Workbench and Mathcad.

While the content of the course is fairly straight forward, the real challenge in its development and delivery is the background of today's *Computer Age* student. The vast majority of entering freshmen at UNC Charlotte are traditional students, in that they are about eighteen years old and have come directly from high school. As such, most have grown up with computers, and have been using them for years. While this may at first seem to be an advantage, it turns out that exactly the opposite is true. Because these students have mastered email, the web, and instant messaging, they feel they already know everything a computer can do.

This paper describes the basic methodologies used in developing this course, along with the various strategies employed in its delivery. An assortment of teaching practices is examined, with opinions offered as to their effectiveness with this unique student population. Before and after survey data are presented, including an initial self assessment by the students of their competency in various software packages, which in most cases turned out to be somewhat overestimated.

Introduction

Prior to the fall of 2004, the Engineering Technology department at UNC Charlotte was exclusively a “two plus two” program, offering only the junior and senior years of the BSET curriculum. All students entered with an associate’s degree, predominately from two-year community college programs. In order to facilitate growth and to bring its structure inline with current trends, the department is now offering all four years of its degree programs and has recently welcomed its first freshman class to campus.

As part of admitting freshmen students, the department had to develop curricula for the first two years of its programs, which include civil, computer/electrical, fire safety, and mechanical engineering technology. Referred to as the “lower division,” this new curriculum includes basic studies outside the department, such as English, math, and physics, and also several courses taught within the department. A course that proved most challenging in its development and delivery was ETGR 1100, Engineering Technology Computer Applications.

The goal of the course is to provide basic computer competencies that the students will need as they move through the curriculum (1). It covers the use of standard office applications in engineering, advanced use of scientific calculators, and also gives a sampling of specialized engineering software. Specific titles include Word, Excel, FrontPage, Photoshop, Electronics Workbench, and Mathcad.

While the goals, layout, and structure of this course are fairly straightforward, the background of today’s student (2, 3) has presented significant challenges in the course’s development and delivery. Entering freshmen that take this class are predominantly traditional students, in that most have come directly from high school. As such, they’ve grown up in the *Computer Age*, and many have been using computers for years. To put it in perspective, most of today’s college freshmen were born between 1985 and 1986, about the same time desktop computers were becoming common in the workplace. By the time these students started high school four years ago, computers were in nearly every school and library, and in many of their homes.

While the vast majority of today’s freshmen students do not need a basic course in how to use computers, they do lack many basic competencies necessary to engage in engineering course work (4). For instance, if they are given an assignment to surf the Web or IM (Instant Message) their friends, they’ll likely laugh and be finished in minutes. But if they’re asked to create a table in Word or a simple line plot in Excel, the results will likely be quite different. One of the biggest challenges in teaching this course is the student’s opinion that, since they’ve mastered email and the web, there’s just not much left to learn about using a computer.

Course Content

The course is broken up into sections grouped around the various applications. The intent of each section is to provide basic competencies that will be needed later in the curriculum. As the focus of this paper is less on the course content and more on its delivery to today’s students, further details about the course itself are contained in the appendix.

The Faculty

Veteran faculty were chosen to develop and teach this class for the first time. All sections of the course were team taught by two faculty members, one with a mechanical background, the other electrical. Both had prior experience teaching freshmen at other institutions, and were chosen based on their positive record of teaching and mentoring students. Retention of new students in the program was an additional goal with the choice of these faculty members.

The Initial Survey

During the first class meeting of ETGR 1100, Engineering Technology Computer Applications, students were given an informal survey. They were asked to rank their level of proficiency in various computer applications on a scale from 1 to 10, with 10 being the most proficient. The purpose of the survey was to determine the appropriate starting point for teaching the various applications. The results of the survey are summarized in Table I.

Table I – Initial Proficiency Survey

<u>Application</u>	<u>Average Estimated Proficiency</u>
Word	7.96
Graphing Calculator	6.34
Excel	4.88
Outlook/Email	4.98
Photoshop	3.02
FrontPage	1.88
Mathcad	1.38

Based on the results of this survey, the initial reaction was to rethink instruction in Microsoft Word, since the students professed such familiarity with it. In reality, most students were unable to do something as simple as creating and formatting a table without instruction, despite their high opinion of their proficiency level. A similar observation was made with regard to usage of Excel and graphing calculators.

The Students

Even though the faculty had taught freshmen in the past, they had taught only junior and senior transfer students for the past several years. It turned out that these faculty members were unprepared for the challenges presented by today's freshmen students.

An immediate observation was that a surprisingly large number of the students didn't regularly attend class. Even though the majority of the students lived on campus, and presumably had no issues with commuting or parking, they frequently skipped class, particularly the 8:00 AM section. This problem was apparently exacerbated by the fact that many assignments were sent

out via email ahead of time. Between already having the assignment and feeling they already knew the material, many simply didn't see the need to show up.

Another observation was a surprising lack of effort by the students on many assignments. A lot of their work was open-ended, meaning that formatting, completeness, etc., was largely up to the individual student. A great many appeared to do just the bare minimum, with seemingly very little effort put forth on most assignments. There were exceptions, of course, but a surprisingly high percentage just wasn't compelled to put forth the effort required for thorough and complete assignments. It became clear that, in addition to teaching computer skills, the instructors also needed to begin to teach professionalism.

An additional surprising observation was the student's apparent indifference about offensive material. As this was a computer based course taught in a computer lab, faculty and other students were frequently exposed to images the students had chosen for their desktop. Many were pornographic or nearly so, and would certainly be considered inappropriate by any standard for a workplace environment. Though this was more isolated, a small number of Photoshop assignments turned into a female faculty member would be considered offensive by any standard. Though freedom of expression is certainly an important right of any student, their choices in this area were unexpected. The instructors learned that even seemingly obvious policies must be stated up front and included in the syllabus.

It was not surprising to observe that assignments sent electronically posed no problem at all for these students. Electronic submission of work was also utilized, again with no problems experienced. Portions of assignments frequently required accessing a website and downloading files, which also wasn't an issue for this group.

A final disturbing development was the numerous cases of academic integrity violations. This is a larger issue that deserves additional study and comment beyond the scope of this paper.

Delivery Options – What Worked and What Didn't

The class was scheduled in a computer laboratory with workstations for each student. It was equipped with a white board and an instructor's workstation that was mirrored on monitors throughout the room. Numerous delivery techniques were attempted with varying degrees of success.

Initially, faculty would lecture at the beginning of class, sometimes using the board, then demonstrate the topic on the instructor's workstation. As the students had internet-enabled computers in front of them, the faculty were largely ignored during this time. The students would then be given time in class to begin an assignment. Shortly thereafter, hands would be raised, asking about things that had been lectured on and demonstrated only moments before.

One variation, attempted with limited success, was requiring monitors be turned off during lecture. One technique that actually did work well was to simply let the students begin an assignment with minimal initial instruction, and then run into elements they didn't understand or

couldn't do. At that point, lecture information relative to their immediate problem was willingly received.

Providing general information on handouts wasn't always effective. Handouts were provided for each assignment with detailed information regarding the assignment, expectations, due date, etc. Students frequently asked questions that were specifically addressed in the handout that had been put aside and not read. This was true even when assignments were given in electronic format. An exception, and a method that seemed to work well, were handouts that gave step by step instructions, such as in a keystroke specific tutorial. Handout material would likely be more effective if it were given on the computer and made interactive.

The Final Survey

Despite the issues discussed above, early indications are that many students enjoyed the class, and many more reported learning a great deal from it. An additional theme was acknowledgement by many that they had overestimated their overall computer proficiency. A final survey was administered, with a specific question regarding that topic for each application. Results are summarized in Table II.

Table II – Final Survey

<u>Application</u>	<u>Percentage Overestimating Proficiency</u>
Graphing Calculator	48.5%
Excel	36.4%
Mathcad	21.2%
Photoshop	21.2%
FrontPage	18.2%
Word	15.2%
Outlook/Email	12.1%

Lessons Learned

After the semester's conclusion, the faculty involved in developing and delivering this course met to discuss successes and failures, and to brainstorm ideas that what would improve the course in the future. The following is a summary of the recommendations.

- Attendance: A rigid, punitive attendance policy should be implemented. Some mechanism, be it attendance as a portion of the final grade, pop quizzes, or the like seems to be required with these students.
- Syllabus: The syllabus needs to clearly delineate expectations regarding attendance, academic integrity, testing, late work policies, professionalism, student behavior, grading, etc. It's important that these details be recorded in writing.

- Unstructured Assignments: Assignments should be unstructured in certain aspects so that independent work will have noticeable differences. An example would be instructions to format Excel columns to an “appropriate width” rather than an explicit one.
- Due Dates: Rigid due dates should be given for all assignments, with severe penalties for late work, if it is accepted at all. This of course should be made clear to the students upfront, and be stated in the syllabus. There was an amazing array of excuses given as to why assignments couldn’t be completed on time, when more than adequate time had been given to complete them.
- Grading Criteria: In addition to the expected correctness of numbers, etc., a major portion of an assignment’s grade should be based on professionalism, completeness, and appearance. Again, this should be communicated to the students upfront. It should be clear that just meeting the minimum requirements does not merit an A.
- Inappropriate Content: Students should be made plainly aware that inappropriate content in any assignments will result in a failing grade on that work.
- Instruction: Lectures and handouts cannot compete with the allure of the interactive computer screen. Faculty are well-advised not to try to compete, but to work with this apparent enthusiasm for the computer in planning instructional activities.

Observations for Specific Applications

In addition to the general lessons learned above, issues arose with specific software that were somewhat unexpected. What follows is a summary of those findings.

- Calculators: It was interesting to note that more students overestimated their proficiency with graphing calculators than in any other area. A disappointing observation is that many students could not grasp the concept of solving simultaneous equations, when that exercise was used in the calculator section. A final observation is that 94.6% of students reported having a Texas Instruments calculator, while none had a Hewlett Packard, the standard choice of many “seasoned” engineers.
- Word: It may have been a mistake to begin with a section on Word. Since students were most familiar with it, they may have mistakenly gotten the idea that the course was easy and attendance was unnecessary.
- Excel: Excel was the second most overestimated element with regard to proficiency. Most students knew about adding rows and columns, but had no idea of Excel’s capability in plotting, curve fitting, equation solving, sorting, macros, etc. Use of IF statements proved to be the most

challenging concept.

FrontPage: As those familiar with FrontPage are aware, the program is set up to build web pages in a WYSIWYG (What You See Is What You Get) format. When students were instructed in viewing and editing the actual HTML (Hypertext Markup Language) code, there was great consternation and resistance.

Electronic Workbench: Electronic Workbench is a very user-friendly program used to simulate, design, and troubleshoot electric/electronic circuits. It is highly recommended for students of any engineering discipline at both the introductory and advanced level. Since this course was delivered to many non-electrical majors, assignments were limited to simple applications, such as connecting a battery to a light bulb, voltmeter and current meter.

Mathcad: Mathcad is a very powerful, but not particularly user friendly program with numerous quirks. The students were generally unwilling to work through provided tutorials, and had difficulty with many simple assignments. Both students and faculty experienced frustration with this section, and overall, its introduction was disappointing.

Conclusion

Today's *Computer Age* students present unique challenges with regard to delivering an entry level computer competency course in engineering technology. These students enter with significant computer skills in email, web browsing, and synchronous chat, but lack necessary preparation in applications used in engineering coursework such as Excel and Mathcad.

Many of today's students overestimate their computer proficiency when entering the course. They have little understanding of applications software, and do not appreciate, until the course is well underway, just how much they *don't* know about what a computer can be used for.

A surprising percentage of today's students do not place a high priority on attending class. Most do not exhibit a high level of effort on assignments. Just doing the minimum amount needed to get by seems to be acceptable to a large portion. A small but disturbing percentage are inclined to commit violations of academic integrity.

Traditional delivery methods are often ineffective in teaching an entry level computer competency class to today's students. They do not respond well to traditional board lectures, nor do they digest general information from handouts. They do, however, respond positively to lecture material once the need to know it has been established via an unsuccessful hands-on exercise. Keystroke specific tutorial-type assignments do appear to be effective.

Today's students are most receptive to course materials and assignments distributed electronically. They are perfectly comfortable submitting work electronically, and have no problems accessing websites and downloading files as part of their assignments.

Faculty teaching similar courses can avoid some of the inherent pitfalls by using carefully designed policies regarding expectations for student behavior, including attendance and professionalism, which are clearly stated in the syllabus.

Board lectures and handouts cannot compete with the allure of the interactive computer screen. Faculty are well-advised not to try to compete, but to work with this apparent enthusiasm for the computer in planning instructional activities.

References

1. Pionke, C., Weber, F., Parsons, J., Seat, J., Yoder, D., *A Strategy for Ensuring Minimum Competency in the Use of Engineering Computer Skills*, Proceedings of the 29th ASEE/IEEE Frontiers in Education Conference, San Juan, Puerto Rico, November 1999
2. Hoffman, M., Blake, J., *Computer Literacy: Today and Tomorrow*, Journal of Computing Sciences in Colleges, Volume 18, Issue 5, Pages: 221-233, May 2003
3. Fellows, S., Culver R., Ruggieri P., Beston, W., *Instructional Tools For Promoting Self Directed Learning Skills In Freshmen*, ASEE/IEEE Frontiers in Education Conference, Boston, MA, November 2002
4. Banik, Gouranga, *Introductory Computer Applications for AEC Freshmen*, Proceedings of the 2002 ASEE Annual Conference and Exhibition, Montreal, Quebec, Canada, June

Biography

GREGORY K. WATKINS

Gregory Watkins received a B.S. in Mechanical Engineering from North Carolina State University, a Master of Engineering Management from Old Dominion University, and a Ph.D. in Mechanical Engineering from UNC Charlotte. He has taught in the Engineering Technology department at UNC Charlotte since 2002. He taught in the Engineering Technologies Division at Central Piedmont Community College for 8 years and has 9 years of industrial work experience. He became a registered professional engineer in 1992.

NAN A. BYARS

Nan Byars received a BS in Mechanical Engineering from Clemson University and an MS in Mechanical Engineering from West Virginia University. She has been a Professor of Engineering Technology at UNC Charlotte since 1993. She taught at California Polytechnic State University in San Luis Obispo CA for eight years and has worked as a project and research engineer in industry. She became a registered professional engineer in 1981.

AMBROSE G. BARRY

Ambrose 'Bo' Barry received a B.S. in Electrical Engineering from Arizona State University and a Master of Science in Electrical Engineering from Auburn University. He has taught in the Engineering Technology department at UNC Charlotte for the past 18 years. He taught in the Engineering Technology Department at New Mexico State University for 4 years and part time for Chapman College at Holloman AFB, NM for 5 years. He was an Electrical Engineer in the USAF for 12 years and in industry for 4 years.

Appendix 1 – ETGR 1100 Syllabus

THE UNIVERSITY OF NORTH CAROLINA AT CHARLOTTE

The William States Lee College of Engineering
Computer and Electronics Engineering Technology

ETGR 1100 Engineering Computer Applications

Course Description:

This course is an introduction to computer-aided-engineering (CAE) software and their applications. Topics include electronic circuit analysis programs, mathematical tools, and calculator applications. (3)

Instructors: Professor Bo Barry abarry@uncc.edu 704-687-4185 Smith 214
Professor Nan Byars nabyars@uncc.edu 704-687-4143 Smith 311

Texts: Pond, Robert, *Introduction to Engineering Technology*, Prentice Hall, 2004
Duffy, J.A., *Microsoft Office Word 2003*, Thompson Publishing
Reding, E.E., Wermers, L. *Microsoft Office Excel 2003*, Thompson Publishing

Prerequisite: None

Course Contents:

Week 1 Introduction to PC operations/MOSAIC
Week 2-3 Microsoft Word applications
Week 4-6 Excel: data analysis & plotting
Week 7 Electronic Work Bench
Week 8-12 Math concepts, Calculators, MathCAD
Week 13 Photoshop.
Week 14-15 FrontPage: web page design

Grading:

Weekly quizzes: 40%
Homework & projects: 35%
Final Exam: 25%