In this issue:

Developing Oral and Written Communication Skills in Undergraduate Computer Science and Information Systems Curriculum

Yana Kortsarts
Widener University
Chester, PA 19013, USA

Adam Fischbach
Widener University
Chester, PA 19013, USA

Jeff Rufinus
Widener University
Chester, PA 19013, USA

Janine M. Utell
Widener University
Chester, PA 19013, USA

Suk-Chung Yoon
Widener University
Chester, PA 19013, USA

Abstract: Developing and applying oral and written communication skills in the undergraduate computer science and computer information systems curriculum - one of the ABET accreditation requirements - is a very challenging and, at the same time, a rewarding task that provides various opportunities to enrich the undergraduate computer science and computer information systems curriculum. We discuss a comprehensive three-step program developed at our institution that provides opportunities to efficiently develop oral and written communication skills. We present our successful experience of integrating a project component into a core introductory course designed to focus on developing these skills. We discuss and present the results of interdisciplinary collaborations between Computer Science and English faculty members to achieve the goals of the project. We also discuss a Senior Design Project course sequence and a Research Topics in Computer Science course, and the opportunities that these courses provide to develop oral and written communication skills.

Keywords: oral and written communication skills, interdisciplinary collaboration, computer, computer science and information systems curriculum

This paper was in the 2009 cohort from which the top 45% were accepted for journal publication. Acceptance is competitive based on at least three double-blind peer reviews plus additional single-blind reviews by the review board and editors to assess final manuscript quality including the importance of what was said and the clarity of presentation.

Developing Oral and Written Communication Skills in Undergraduate Computer Science and Information Systems Curriculum

Yana Kortsarts, Computer Science
ykortsarts@mail.widener.edu

Adam Fischbach, Computer Science
jafischbach@mail.widener.edu

Jeff Rufinus, Computer Science
jeff@rufinus.cs.widener.edu

Janine Utell, English
jmutell@mail.widener.edu

Suk-Chung Yoon, Computer Science
syoon@mail.widener.edu

Widener University
One University Place
Chester, Pennsylvania 19013 USA

Abstract

Developing and applying oral and written communication skills in the undergraduate computer science and computer information systems curriculum - one of the ABET accreditation requirements - is a very challenging and, at the same time, a rewarding task that provides various opportunities to enrich the undergraduate computer science and computer information systems curriculum. We discuss a comprehensive three-step program developed at our institution that provides opportunities to efficiently develop oral and written communication skills. We present our successful experience of integrating a project component into a core introductory course designed to focus on developing these skills. We discuss and present the results of interdisciplinary collaborations between Computer Science and English faculty members to achieve the goals of the project. We also discuss a Senior Design Project course sequence and a Research Topics in Computer Science course, and the opportunities that these courses provide to develop oral and written communication skills.

Keywords: oral and written communication skills, interdisciplinary collaboration, computer, computer science and information systems curriculum

1. INTRODUCTION AND MOTIVATION

Developing and applying oral and written communication skills in the undergraduate computer science and computer information systems curriculum - one of the ABET accreditation requirements (ABET accreditation website http://www.abet.org/) - is a very challenging and, at the same time, a rewarding task that provides various opportunities to enrich the undergraduate com-

puter science and computer information systems curriculum. Many colleges and universities recognize the importance of developing oral and written communication skills, and design courses with writing intensive components developed through writing in the disciplines programs (Giangrande, 2009; Bakke, 2008; Becker, 2008; Maurino, 2008; Dansdill, Hoffman, Herscovici, 2008; Dugan & Polanski, 2006; Hoffman, Dansdill, Herscovici, 2006; Osborne, 2006; Takeda, Crabtree & Johnson, 2006; Martin, 2005; Owen & Young 2005; Pomykalski, 2005, 2003; Kaczmarczyk, 2003; Ladd, 2003; Anewalt, 2003; Fell, Proulx & Casey, 1996). While faculty can learn from the experiences of others, designing courses with oral communication and writing enriched components still takes a lot of effort; the individual features of the department should be taken into account, as well as faculty resources.

Our department offers an undergraduate program leading to Bachelor of Science degrees in both Computer Information Systems (CIS) and Computer Science (CS). Both majors culminate their study with the two-course Senior Design Project sequence. One of these courses is a writing-enriched course and requires the writing of a 25-page final paper. Students also give several oral presentations during the academic year and present the final project results at the University's Student Project Day. This sequence provides an opportunity to develop oral and written communication skills, but does have several limitations: (1) students take this sequence at the end of their study without extensive previous experience; (2) the sequence provides limited opportunities to show the improvement of oral and written communication skills.

In Spring 05, we integrated a project component into Introduction to Computer Science II with C (CS 2), the second core course in the computer science and computer information systems curriculum. This course has been successfully running now for five years. The project component has been extensively revised since our initial report (Kortsarts & Rufinus, 2006). In Spring 07 we established a writing-in-the-disciplines collaboration with an English faculty member. Writing-in-the-disciplines (WID) is a crucial component of teaching and learning at our regional comprehensive university. Faculty collaboration in the areas

of writing and speaking, and their assessment across the curriculum, plays a vital role in achieving our learning objective of developing and strengthening written and oral communication. Those faculty who participate in the WID program seek opportunities to collaborate through workshops, shared course design and classroom teaching, and assessment strategies.

In this context, we designed an interdisciplinary collaboration focusing on developing written and oral communication skills in the project component in CS 2. In this course, students must self-teach content material related to an advanced computer science or information systems topic, present their work to the class, and assess their fellow students' learning. Participation in the project component is required and provides undergraduates with enriching experiences that help to develop oral and written communication skills, as well as collaboration skills, from the very beginning of the students' academic career. While the project component has had successful results, we realized that students still had a gap between introductory and senior level courses. Recently, the department developed a one-credit "Research Topics in Computer Science" course that will be integrated into the core curriculum in Spring 2010. We will offer this course in the junior year, which will provide an opportunity for students to further enhance oral and written communication skills and to be fully prepared for the Senior Design Project sequence.

In this paper we discuss in detail the structure of the revised CS 2 project component, its implementation, results of the collaboration with the English faculty member, and project evaluation. We also present the Senior Project and Research Topics courses as a way to develop and apply oral and written communication skills.

2. PROJECT COMPONENT IN CS 2

2.1 Course Description

The CS 2 course combines a thorough introduction to the C language with a survey of advanced computer science and information systems topics; supervised lab includes a sequence of exercises in the C programming language. The structure of the four-credit

course consists of three hours lecture and three hours laboratory.

2.2 Project Component Goals

(1) Developing oral and written communication skills;

(2) Developing collaboration skills;

(3) Engaging students in active learning and knowledge exchange activities;

(4) Improving the effectiveness of teaching and student learning.

2.3 Project Component Implementation

(1) The class is divided into teams of two students;

(2) Each team chooses the topic from the provided list by the end of the first week of classes;

(3) Students are provided three topic choices to avoid duplicating topics.

In Spring 09, two sessions of CS 2 were offered with a total of 18 students who chose to learn about the following topics: (1) bioinformatics, (2) programming languages, (3) computer forensics, (4) cryptology, (5) database systems, (6) artificial intelligence in games, (7) HTML, networking and the Internet, (8) algorithms, and (9) computer organization. The project component contributed 25% to the final grade in the course.

2.4 Project Component Structure

Students are required to (1) self-learn the chosen topic, (2) understand the main concepts related to the chosen topic, (3) complete the programming example in C illustrating the chosen topic, (4) learn to read professional literature by completing and presenting summaries of papers from the ACM or IEEE digital libraries, and (5) collaborate through team work.

2.5 Project Component Assessment Plan

Each team (1) submits a written progress report every week in the lab, (2) provides a very brief oral report every week during lab time, (3) prepares three 5-7 minute PowerPoint oral presentations given during lecture at different points over the course of the semester which provides an opportunity for teams to exchange knowledge and to learn the topics in a progressive way, and (4) prepares a 20-minute final PowerPoint oral presentation that includes a short quiz to assess peer learning.

2.6 Requirements for presentations and progress reports

This part of the project has been continually revised, taking into account experiences from previous iterations. In Spring 09 students received a precise list of requirements for each oral presentation and for some progress reports. We have learned that this model works the best for freshman students. Our experience has shown a direct correlation between student performance and the precision of guidance provided. Students' independent learning improved dramatically with the more specific structuring of the project component. We provide examples of the instruction that students receive for some progress reports and for oral presentations below.

After choosing the topic in the first week of classes, students are required to meet with their teams; they read the corresponding chapters from the course textbook (Schneider & Gersting, 2006) that are related to their topics and prepare their first progress report. Students also receive reading material for the topics that are not covered in the course textbook. The first progress report that students submit during the second week of classes must include the tentative project goals and implementation plan, including a list of examples.

The first oral presentation requires five PowerPoint slides including a title slide, project goals, project plan, including programming and non-programming examples that will be explored, a list of the main concepts with their definitions, and references, including the relevant article to be summarized from Communications of the ACM Magazine (2003-2009).

The second oral presentation, in addition to the title and reference slides, requires a list of the main concepts, their definitions and examples that illustrate the definitions. Students are asked to choose up to five main concepts with specific examples. We ask students to make the examples and definitions as simple as possible. Students must also discuss the project progress and pro-

vide description and details for the programming example: in this presentation students have to formulate the exact example that they will implement in the C programming language. By the second presentation, students are required to have a clear idea of what the input and output are for their program. Students are also asked to present a sketch (might be an algorithm already) of the solution. In the second presentation, students also perform an intermediate assessment of the material learned so far using a short quiz of 4-5 questions. Quiz questions can be multiple-choice or open-ended. We ask students to think about the best way to assess whether the material that they presented was comprehended by their peers.

The third oral presentation includes the project progress and one or two additional project concepts with illustrations that were not covered in previous presentations. For the programming example part, the third presentation must show at least 50% completion of their programming example. Students must provide an illustration of their program with specific examples and a detailed explanation on how their program works. The third presentation must also devote time to summarizing the research article that was chosen at the beginning of the project.

For the final presentation, students follow the following guidelines:

- Length of the presentation - 20 minutes

  o 15 minutes – Talk

  o 3 minutes - Short Quiz - 5 questions

  o 2 minutes - Questions and Answers

  o The PowerPoint and Word files with programming examples have to be posted on the course website

- Content:

  o Summary of the topic: brief description and definition of the topic and list of main concepts with very brief definitions

  o The detailed description of the programming example, including the explanation of the topics that we learned in class and were used in the program

o Explanation of the connection between the programming example and the topic as a whole

o Explanation of the program using specific examples

o Summary - what material was learned during the project, challenges, any future plans

o References

o Quiz: five well-designed questions to check the understanding and comprehension of the presented material

o Handouts (optional)

o The last 2 minutes used to check the quizzes and to answer any questions.

2.7 Collaboration with English / Writing Faculty

To accomplish the goals of the project component, an interdisciplinary collaboration with an English/Writing faculty member was established initially in Spring 2007 and continued in Spring 2009. In constructing and implementing this component in CS 2, an English/Writing faculty member joined the class to observe and assess the students' presentations, providing global written feedback for the students' use, and meeting with the CS faculty member for discussion of student performance. The students moved through foundational, competent, and masterful levels of self-teaching and knowledge exchange over the three stages of the presentation component, and the faculty collaboration was part of each stage of the student work. In Spring 2009 as part of the faculty members' assessment, it was determined that integrating the written and oral component more completely into the project would be beneficial. As a result, the English/Writing faculty member taught lessons on critical reading and effective oral presentation design, which were integrated into the course work at the beginning of the semester before students started to work on the project. The presentation covered the following issues: (a) discussion of how to read critically and what questions to ask while reading the text; (b) discussion of how to summarize the paper using the structure of the essay as a guide and elucidating key points and key moments of evidence while

making connections to the rest of the class material; (c) tips on writing the summary that include prewriting, drafting, revising; and (d) discussion of how to design an effective presentation of information. The English faculty member was present at all oral presentations and provided detailed notes for each team explaining ways the presentation could have been stronger and also pointing out the positive and negative aspects of the presentation. These valuable comments allowed students to improve their performance and their oral and written communication skills, and to complete the course with a well-designed final oral presentation. The collaboration with the English faculty member provided an opportunity to accomplish the goals related to the development of oral and written communication skills. This successful and enjoyable experience showed the value of working with colleagues across disciplines to further student learning.

Goal	*Average*
	Rates from 1 to 5, 1-not accomplished, 5- accomplished completely
Developing oral and written communication skills	3.7
Developing collaboration skills	3.7
Engaging students in active learning and knowledge exchange activities	3.8
Improving the effectiveness of teaching and student learning	3.9

Table 1: Accomplishment of the goals Spring 2009

2.8 Project Component Evaluation and Results

Starting in Spring 2009, to evaluate student performance, we used effective communication rubrics in the course. The general education committee of the university designed the rubrics with the active participation of the English faculty member who joined our course teaching. The rubrics were introduced and explained to students during the initial presentation by the English faculty member early in the course. We evaluated each oral presentation using these rubrics and provided each student with a detailed explanation of their evaluation. (The rubrics are presented in the Appendix.) In addition, the course instructors prepared a short post-survey to assess the results of the project component. The summary of the survey results for Spring 2009 are shown in Table 1.

First, we ask students to rate the level of accomplishment of each project component goal using the scale 1 through 5, where 1 indicates "not accomplished" and 5 indicates "accomplished completely." In Spring 2009, 15 students answered the survey and the average for each objective is presented in Table 1.

While examining these results it is important to take into account students' comments and explanations of their ranking. Students provided the following valuable comments:

- "oral and written skills became developed because we had to present almost every other week and practice these skills"

- "had to talk in front of the class and received feedback from English teacher helped a lot"

- "we learned how to work with the partner effectively and how to teach the class so students will pay attention"

- "learned a lot from article that we read and from writing the program"

- "got to learn to teach myself, teach the class and share our knowledge"

- "learned to talk effectively with my partner and was able to get my ideas out"

- "it helped that everyone had to listen to everyone and participate in quizzes"

- "learned what it is like to program with partner"

- "groups learned good amount on their own"

- "students did have to be able to explain and convey their area of study"

- "it did focus on students making progress along throughout the term"

- "I learned a few technique how to be clear to the audience"

- "it was positive experience to develop a complex program from scratch"

Some students mentioned that there were fewer opportunities to develop writing skills. Some students pointed out that the project was time-consuming, and it was difficult to find time to work with the partner. Some students did not like the idea of the weekly progress reports. Also, some students mentioned that they learned their own topic well, but it was sometimes hard to learn other topics at the same level from their peers' presentations alone. All students reported that the project component met their expectations and most of them would participate in a similar project in the future. We can also compare the results over the years 2005-2009. While not all years are equivalent (2005 and 2007 had an optional Service-Learning component), and the structure of the presentations was continually revised and changed, we can see some change in the accomplishments of the goals, but it was not significant. Since the same computer science and English instructors were involved in the teaching of this course, we had an opportunity to compare student performances over time and see the improvement and the development of complexity in the projects over the years. While these results based on indirect assessment are of a subjective nature, the introduction of the effective communication rubrics in Spring 09 provides opportunities to improve our method and to conduct more formal direct assessment of the project component.

2.9 Some Implementation Challenges

For some projects it was very challenging to find interesting, technically sound programming examples related to the field of study. For the programming languages topic, for example, the initial thought was to design the same program using various languages and show the implementation differences. As students proceeded with the project, we realized that this task would be very difficult to accomplish within the time frame devoted to the project component. A good alternative was found: in addition to the theory of pro-

gramming languages, students learned the column sorting algorithm, read and presented the summary of the research paper related to the column sorting (McCann, 2004), implemented the algorithm in the C programming language, and conducted a comparison of the running times of several sorting algorithms.

2.10 Summary and Future Plans for the Project Component

Based on the project evaluations, students' post-survey results and instructors' observations, we can state that the integration of the project component in CS 2 was successful. For future iterations, we are planning to follow the existing structure, but several changes will be introduced. While students have more opportunities to develop oral skills, we have several ideas to improve the development of writing skills. Weekly progress reports will become more structured and students will receive feedback towards improving their writing on weekly basis. We will also ask students to conduct a more formal summary of the research article and write and submit the summary in the middle of the semester. This will allow us to provide students with feedback on their writing and opportunities to improve their writing toward the end of the semester. In addition, students will be required to write a 5-7 page summary of the project at the end of the semester, including a reflective component taking into account all comments that they received from writing progress reports and the article summary. We are planning to use the effective written communication rubric to assess the writing component of the project. This rubric is presented in the Appendix of the current article.

3. SENIOR DESIGN PROJECT

In their senior year, CS and CIS students engage in a year-long effort to complete a Senior Project. Typically, students work in pairs to research, design and implement a project of their choosing. Near the end of the spring semester, each team presents the final results of their project to a general audience and submits a 25-page report to the computer science faculty. Not only does the Senior Project allow students to demonstrate the computing skills they have attained throughout their undergraduate education, it

also provides them with the opportunity to further develop their communication skills.

The nature of the projects varies according to students' interests, background and aptitude. While many projects involve the development of software, others focus more on the implementation or application of existing software or hardware to solve specific problems. Recent projects have included the development of a Java application that demonstrates data encoding schemes, the design and implementation of an alumni database, and the upgrading and reconfiguring of a computer network for a local real estate company. One theme all of the projects have in common is their practical applicability: The Java application will be used by students in our networking class to understand data encoding algorithms. The alumni database will be used by the Computer Science Department to maintain easily accessible data on our alumni. The computer network will be used by the employees of the real estate firm.

While the students are working on solving these real-world computing problems, they also work to develop real-world communication skills. Beginning in the fall semester, students working on Senior Project are required to give periodic oral presentations to the computer science faculty describing their project goals and the progress made so far. The results of these presentations have convinced us that the development of communication skills cannot be relegated exclusively to English and Speech courses. Students often have difficulty discussing technical topics. In particular, they do not know who their audience is. Since we are computer science professors, they assume we know everything about the field. It does not occur to them that an expert in programming languages may know very little about network encoding schemes, for instance. By providing them regular feedback, we can teach them how to present technical information so that it is accurate, clear and accessible to the audience.

Students begin giving oral presentations within a few weeks of starting their projects, and receive regular feedback from faculty. In the past, a faculty evaluation form was used to provide feedback for these oral presentations. The forms were very simple and allowed faculty to provide basic com-

ments and recommendations for the next round. In the future, we are planning to use the effective communication rubrics (described earlier and included in the Appendix below) for the Senior Project presentations to further improve student performance.

After making several presentations to computer science faculty throughout the academic year, students are finally required to present their project before a general audience of college faculty and students during the annual Student Project Day. This final presentation allows students to demonstrate how well they have truly learned to master oral communication skills. Concomitantly, as part of developing written communication skills, each team of students writes a 25-page report in which they detail the motivations and results of their project and discuss what challenges they faced and what lessons they learned over the course of the year.

Unlike the oral communication component, however, the writing, unfortunately, is often something the students put off until the end of the spring semester. Since the reports are essentially thrown together at the last minute, they tend to be of poor quality. This suggests that, though students learn and practice writing skills extensively in numerous "writing enriched" courses, they do not transfer those skills to their computer science courses. We recently created a Research Topics course, in part to address this phenomenon.

4. RESEARCH TOPICS IN COMPUTER SCIENCE

In the spring semester of their junior year, students will take a one-credit Research Topics course designed to prepare them for their Senior Project. The purpose of the course is to 1) introduce students to a variety of research topics, 2) enable them to make an informed decision about which topic to pursue, and 3) provide them with an opportunity to conduct initial research into their chosen topic.

In previous years, students were handed a list of project topics to choose from. (They could also develop their own project.) Students would select one of the topics, often not knowing what they were getting themselves into. We had a situation in the 2008-2009 year in which one team was forced to

abandon their topic halfway through the fall semester when they realized it was not something they could reasonably pursue. The Research Topics course is designed to give the students a thorough introduction to a set of potential topics to give them a better understanding of current research in the computing field and a better opportunity to choose a topic that suits them.

During the course of the semester in Research Topics, each member of the computer science faculty will give a two-hour lecture on research topics in his or her field of computing. After this series of lectures, students will meet with their chosen faculty member to discuss a potential project. For the remainder of the semester, the students will begin initial research into their topics and prepare formal project proposals.

At the end of the semester, students will submit a five to ten page paper discussing what they have learned from their initial research and proposing a set of goals for their Senior Project. By requiring them to write a formal proposal, we hope to emphasize the importance of written communication skills early in the process of completing the Senior Project. By also requiring periodic *written* updates of their projects during their senior year (in addition to the oral presentations), we will be able to provide them with continuous feedback and guide them in thinking about writing from proposal to completion of the Senior Project.

5. SUMMARY AND FUTURE PLANS

It is a challenging task to design a comprehensive program for the development of oral and written communication skills for CS and CIS majors. In this paper we present an effective way to overcome some of the difficulties and create a three-step program to develop these skills. It is important to emphasize that the proposed approach uses only core courses in the CS and CIS curriculum; this ensures that all CS and CIS students have an opportunity to gain this knowledge. We also carefully take into consideration the tightness of the CS/CIS curriculum and integrate oral/writing enriched components into classic core courses that are offered by most programs and departments, such as introductory programming and senior project/capstone courses. To conclude, our department also makes constant efforts to

integrate oral/writing enriched components into a wide selection of elective courses, complementing the work done in the core courses. Some of these experiences include Introduction to Bioinformatics (Kortsarts, Morris and Utell, 2008) and Programming Languages courses.

6. REFERENCES

ABET Accreditation Website: http://abet.org

Anewalt, Karen, (2003), **"**Experiences teaching writing in a computer science course for the first time". Journal of Computing Sciences in Colleges, Volume 18, Issue 2, Pages: 346-355

Bakke, Sharen, (2008), "Try This Sure-Fire Way to Get Students Involved." In The Proceedings of the Information Systems Education Conference 2008, v 25

Becker, K, (2008), **"**The use of unfamiliar words: writing and CS education". Journal of Computing Sciences in Colleges, Volume 24, Issue 2, Pages: 13-19

Dansdill, Timothy, Hoffman, Mark, Herscovici, David, (2008), "Exposing GAPS, exploring legacies: paradoxes of writing use in computing education", Journal of Computing Sciences in Colleges, Volume 23, Issue 5, Pages: 24-33

Dugan Robert, Polanski, Virginia, (2006), **"**Writing for computer science: a taxonomy of writing tasks and general advice", Journal of Computing Sciences in Colleges, Volume 21, Issue 6, Pages: 191 - 203

Fell, Harriet, Proulx, Viera, John, Casey (1996), "Writing across the computer science curriculum**".** Proceedings of the 27th SIGCSE technical symposium on Computer science education, pp. 204-209

Giangrande, E. Jr. (2009), "Communication skills in the CS curriculum", Journal of Computing Sciences in Colleges, Volume 24, Issue 4, Pages: 74-79

Hoffman, Mark, Dansdill, Timothy, Herscovici, David, (2006), "Bridging writing to learn and writing in the discipline in computer science education**".** Proceedings of the 37th SIGCSE technical symposium on Computer science education, Pages: 117-121

Kaczmarczyk, Liza (2003), **"**A technical writing class for computer science majors: measuring student perceptions of learning**".** Proceedings of the 34th SIGCSE technical symposium on Computer science education, pp: 341-345

Kortsarts, Y, Morris, R, Utell, J, (2008), "Interdisciplinary Introductory Course in Bioinformatics". In The Proceedings of the Information Systems Education Conference 2008, v 25

Kortsarts, Y, Rufinus, J, (2006), "Integrating a Project Component into the Introductory Computer Science and Information Systems Curriculum". In The Proceedings of the Information Systems Education Conference 2006, v 23

Ladd, Brian (2003), **"**It's all writing: experience using rewriting to learn in introductory computer science**".** Journal of Computing Sciences in Colleges, Volume 18, Issue 5, Pages: 57-64

Maurino, Paula San Millan (2008), "Writing to Learn and Writing in the Disciplines: Enhancing Computer and Technical Classes". In The Proceedings of the Information Systems Education Conference 2008, v 25

McCann, Lester (2004), **"**Contemplate sorting with columnsort**".** Proceedings of the 35th SIGCSE technical symposium on Computer science education, Pages: 166-169

Osborn, Lawrence, (2006), "Thinking, speaking, and writing for freshmen**",** Proceedings of the 37th SIGCSE technical symposium on Computer science education, pp: 112 - 116

Owen, William N, Young, William (2005). "Tips for Improving Writing in IS/IT Courses". In The Proceedings of the Information Systems Education Conference 2005, v 22

Pomykalski, James J (2005), "Teaching Systems Analysis and Design as a Writing-Intensive Course". In The Proceedings of the Information Systems Education Conference 2005, v 22

Pomykalski, James J (2003), "Critical Thinking through Writing in Information Systems Courses". In The Proceedings of the Information Systems Education Conference 2003, v 20

Schneider, Michael, Gersting Judith, (2006), "Invitation to Computer Science: Java Version". Course Technology; 3 edition

Takeda, Hirotoshi, Crabtree, Sara, Johnson, Roy D (2006). ""WAC"ked: A Case Study Incorporating a Writing Process into an IS Class". In The Proceedings of the Information Systems Education Conference 2006, v 23

APPENDIX

**Table 1: Assessment Criteria for
"Students will be able to give a presentation before a group"**

Level / Criteria	Masterful	Competent	Developmental
Content	Student presenter has a clear purpose, grounded in knowledge of the subject matter and reflecting critical thought upon that knowledge. Student presenter demonstrates evidence of thorough preparation reflected in quality and depth of information and/or argument presented, use of appropriate details, visual aids, research and documentation, etc.	Student presenter has a purpose, although focus is at times lost, and evidences basic understanding of the subject matter and some critical thought. Student presenter is for the most part prepared, but may be lacking in some thoroughness or detail; supplementary material/evidence/research/ visual aids may be less than satisfactory.	Student presenter lacks a clear purpose or foundation in course material or knowledge of subject matter. Student presenter is unprepared and unfocused; discussion of information and/or argument lacks depth and detail.
Performance	Student presenter's stance conveys credibility, confidence, and expertise. Student presenter crafts a delivery that includes appropriate presentation of physical presence, voice, and language.	Student presenter's stance is credible overall, but may lack some confidence. Student presenter has a generally strong delivery, but may have weakness in physical presence, voice, and/or language.	Student presenter demonstrates no credibility, confidence, or expertise. Student presenter has poor or inappropriate delivery; physical presence, voice, and language are used poorly or inappropriately.
Awareness of Audience	Student presenter has constructed a clear organizational pattern to facilitate audience listening and understanding, with a strong introduction and conclusion supported by a coherent and logical presentation of information or argument. Student presenter has a strong engagement with the audience and is conscious of appropriate pacing and individual and group response, including during Q&A.	Student presenter creates an organizational pattern that facilitates audience understanding, but may not have coherence all the way through; introduction, conclusion, and/or transition points may be weak. Student presenter is aware of audience needs and responses, although may not engage effectively through adjustments in pacing, strong Q&A, etc.	Student presenter constructs a disjointed or flimsy organizational pattern that fails to lead the audience effectively through the material. Student presenter lacks awareness of audience needs and responses; presentation is poorly paced and presenter fails to engage.

Table 2: Assessment Criteria
"Students will be able to write papers that require locating,
analyzing and formally referencing information sources to support conclusions"

Level *Criteria*	Masterful	Competent	Developmental
Claim	Writer presents an arguable claim, grounded in deep understanding of the discipline and reflecting critical and original thought. Writer reaches reasonable and interesting conclusions based on claims and evidence	Writer presents an intelligible claim, evidencing basic understanding of the discipline and some critical thought. Writer reaches conclusions that are, for the most part, solid.	Writer presents a shaky or simplistic claim which seems to reflect weak grasp of the discipline. Writer reaches tenuous, illogical, or irrelevant conclusions.
Evidence	Writer provides appropriate, relevant evidence, chosen to further claims and establish credibility and evaluated and analyzed according to writer's purpose and context. Writer demonstrates an awareness of disciplinary contributions and synthesizes the ideas of others with his/her own.	Writer provides some evidence that while not fully analyzed is mostly relevant. Writer demonstrates some awareness of disciplinary contributions, although synthesis may be lacking.	Writer provides no evidence, or evidence presented has little to do with the purported claim. Writer offers little or no synthesis of information or research with the writer's own ideas.
Audience	Writer constructs and maintains an organizational pattern that facilitates reader understanding of the argument and information presented. Writer employs style and mechanics suited to the genre of academic writing and the specifics of the discipline, including appropriate word choice, usage, and documentation.	Writer constructs an organizational pattern that allows for general understanding, although components of the structure may be weak or ill-sustained. Writer follows the expectations of academic writing, although there may be flaws in diction, usage, or documentation.	Writer constructs a disjointed or flimsy organizational pattern that fails to lead the reader effectively through the text. Writer employs style and mechanics inconsistent with the expectations of academic writing: misuse of diction, poor usage, flawed documentation.