Volume 11, No. 6
December 2013
ISSN: 1545-679X

INFORMATION SYSTEMS EDUCATION JOURNAL

In this issue:

The **Information Systems Education Journal** (ISEDJ) is a double-blind peer-reviewed academic journal published by **EDSIG**, the Education Special Interest Group of AITP, the Association of Information Technology Professionals (Chicago, Illinois). Publishing frequency is six times per year. The first year of publication is 2003.

ISEDJ is published online (http://isedjorg) in connection with ISECON, the Information Systems Education Conference, which is also double-blind peer reviewed. Our sister publication, the Proceedings of ISECON (http://isecon.org) features all papers, panels, workshops, and presentations from the conference.

The journal acceptance review process involves a minimum of three double-blind peer reviews, where both the reviewer is not aware of the identities of the authors and the authors are not aware of the identities of the reviewers. The initial reviews happen before the conference. At that point papers are divided into award papers (top 15%), other journal papers (top 30%), unsettled papers, and non-journal papers. The unsettled papers are subjected to a second round of blind peer review to establish whether they will be accepted to the journal or not. Those papers that are deemed of sufficient quality are accepted for publication in the ISEDJ journal. Currently the target acceptance rate for the journal is about 45%.

Information Systems Education Journal is pleased to be listed in the 1st Edition of Cabell's Directory of Publishing Opportunities in Educational Technology and Library Science, in both the electronic and printed editions. Questions should be addressed to the editor at editor@isedj.org or the publisher at publisher@isedj.org.

INFORMATION SYSTEMS EDUCATION JOURNAL

Editors

ISEDJ Editorial Board

An Exploratory Study of the use of Video as an Instructional Tool in an Introductory C# Programming Course

Jason H. Sharp
jsharp@tarleton.edu

Leah A. Schultz
lschult@tarleton.edu

Computer Information Systems
Tarleton State University
Stephenville, TX 76402, USA

Abstract

This study examines the background of introductory programming concepts and the use of video as an instructional tool. Thirty-five students in an introductory C# class were administered a survey to report data on demographics, usage on video, and opinions about the video. Students were in online and face to face sections of the class. Data were analyzed to determine how students used the videos and to determine if there were differences between the two groups. Multiple aspects analyzed show no difference in use of the online video between face to face and online students.

Keywords: programming, video, instructional tool, video lectures, e-lectures, online learning

1. INTRODUCTION

Programming, regardless of the specific language utilized, how the programming course is taught, whether the language is procedural or object-oriented, has traditionally been one of the more difficult courses for undergraduate students in an information systems (IS) degree program (Fincher 1999; Jenkins, 2002; Lahtinen, Ala-Mutka, & Jarvinen, 2005; Milne & Rowe, 2002; Robins, Rountree, & Rountree, 2003). Couple this fact with the move toward online education and the problem is exacerbated (Butler & Morgan, 2007). The purpose of the following study is to examine the use of video as an instructional tool to illustrate key concepts in an introductory course in C# programming delivered both face-to-face and online and to compare and contrast findings between the two modes of delivery.

2. LITERATURE REVIEW

There has been an ongoing debate regarding the place of programming in the curriculum of undergraduate IS programs (Topi, Valacich, Wright, Kaiser, Nunamaker, Sipior, & de Vreede, 2010), the primary language that should be used to teach programming (Jenkins, 2002; Russell, Russell, Pollacia, & Tastle, 2010), and how to best teach programming (Fincher, 1999; Jenkins, 2002) and a considerable amount of related research. Similarly, there is an abundance of research related to the use of video, in some form or fashion, for teaching and learning in both face-to-face and online modalities across multiple disciplines in both

education and industry. However, a search of the literature yielded no specific research related to teaching programming in an online environment or to providing support for traditional face-to-face courses using video as an instructional tool to assist students in this traditionally difficult subject area.

Programming in the Curriculum

The issue of whether or not programming should be included in the core curriculum and what type of programming it should be is not a new debate (Cain, 1991; Gotwals & Smith, Jr., 1995; Gotwals & Smith, 1993). Recently, IS 2010, Curriculum Guidelines for Undergraduate Degree Programs in Information Systems (Topi et al., 2010) has removed application development from the set of required core courses. The authors, however, assert that "it is important to understand that although application development is not included in the core, it has not been removed from the IS program, and the task force acknowledges that a strong case can be made for inclusion of programming, computational thinking, data structures, and related material in an IS program" (p. 27). The suggestion is to offer application development as an elective and that programs that want to implement a sequence of programming courses can do so.

Programming Language to Use

In regard to what programming language to use, Jenkins (2002) asserts that "there is scant solid evidence that any language is any better or any worse than any other, and the choice continues to be driven largely by the 'flavour of the month' in industry" (p. 55). There appears to be wide agreement among IS educators that the purpose of an introductory programming course, at least, is to teach students to program more so than to teach them a specific language. It is, however, difficult for students to make such a differentiation. Many get caught up in the details of the syntax, while missing the more important higher level concepts. Another variable in this discussion is that languages created to teach programming, but not currently used in industry are avoided in the attempt to recruit students based on the fact that the program teaches languages currently used in the "real world" (Jenkins, 2002).

Russell et al. (2010) conducted a study of the programming languages used in information systems and computer science curricula. Their goal was to determine if a particular language was better suited for the sequence in which the programming course was offered (first-course, second-course, third-course) based upon curriculum type (CIS, MIS/IS, CS, IT). In their study they examined Visual Basic.Net, Java, and C#. Their results indicated that "only for the second programming course did program type seem to influence the programming language used" (p. 10) and that a Windows-based interface was primarily used for the first-course and second-course, while a Web-based interface was primarily used for the third-course.

Teaching and Learning Programming

Jenkins (2002) asserts that "at the moment the way in which programming is taught and learned is fundamentally broken" (p. 53). He goes on to say that "few computing educators of any experience would argue that students find learning to program easy" (p. 53). In light of these statements, Jenkins summarizes several potential reasons that students have such a difficult time learning to program. These include commonly cited reasons such as lack of aptitude or cognitive factors including learning styles and motivation. Rountree et al. (2003) also provides an excellent review of the literature in the general context of cognitive psychology as it relates to learning how to program, specifically: the task, mental models and processes, and novice capabilities and behaviors.

Jenkins (2002) goes on to cite another reason why students find programming difficult which he calls "life skills". These particular skills are not as commonly cited by IS educators, but may include transitioning to college life and such intangibles as being away from home for the first time, struggling to develop new friendships, and having to manage finances and personal and study time for the first time. It is within this time of "transition" that many students encounter their first experience with programming. Jenkins asserts, "this is difficult enough material to master when a student is well settled, but departments' insistence on teaching this during a period of transition can only increase the difficulty" (p. 55).

Jenkins (2002) also suggests that adding to the difficulty of programming is the fact that it consists of multiple skills and multiple processes. This "hierarchy" of skills begins with lower level skills such as the basics of syntax and

progresses to higher level skills such as semantics, structure, and style. Related to processes, the programming student must be able to translate specifications into an algorithm, consider if these specifications resemble something from past experience that can drawn upon, and finally must convert the algorithm to actual code. It is essential, therefore, that a student master all three processes. As Jenkins put it, "there is little point in lecturing to students on syntax when they have no idea of where and how to apply it" (p. 55).

IS educators are all too familiar with the student who attends every class meetings, appears to follow the lectures, seems to grasp the program examples, but is "incapable of writing their own program. They have not mastered all the processes; they can code, but they cannot produce an algorithm" (Jenkins, p. 55). Lahtinen et al. (2005) echo these sentiments by stating, "the biggest problem of novice programmers does not seem to be understanding of basic concepts but rather learning to apply them" (p. 17).

Fincher (1999) states that the approach to learning to program prior to the emergence of Computer Science (CS) and Computer Information Systems (CIS) as distinct disciplines was geared toward learning the "languages and techniques of programming for a specific purpose" (p. 12a4-1). As these disciplines have matured, however, she suggests that "programming is not taught as a process separate from purpose. We no longer teach programming in order to get the computer to *do* something, but as a transferable skill in its own right" (p. 12a4-1). She then goes on to summarize and evaluate four approaches for teaching programming: (1) the "syntax-free" approach, (2) the "literacy" approach, (3) the "problem-solving" approach, and (4) computation as interaction. Fincher suggested that the differences within each approach lies in how it defines what comprises programming. In terms of commonality, she asserts, "all the approaches have in common the idea that coding is separate from programming" (p. 12a4-4). Fincher concludes by stating that the "debate about what we should be teaching undergraduate computer scientists is not particularly new" (p. 12a4-4); however, "what is new is the questioning of what we are aiming to do in the teaching of programming" (p. 12a4-4). Simply put, it is not "what" is taught, but rather "how" and "why" it is taught.

Video Lectures or E-Lectures

A database search of the terms video-based learning, video-based training, and video-based instruction results in myriad ways video has been used in both educational and professional settings ranging from accounting (Martin, Evans, & Foster, 1995), ethics (Sedaghat, Mintz, Wright, 2011), drug education (Dusenbury, Hansen, & Giles, 2003), learning and motivation (Choi & Johnson, 2005), promotion of student-centered learning (Gainsburg, 2009), and acquisition of technical skills such as suturing and knot-typing for medical students (Dubrowski & Xeroulis, 2005) and block-laying and concreting for distance learners (Donkor, 2011), just to name a few. Suffice it to say, the use of video in some form or fashion is not a new phenomena to the teaching and learning discipline.

The use of video, in some form, within business-related education is not without representation in the research literature. For example, Mintu-Wimsatt (2001) conducted a study between two MBA classes, one delivered in a traditional face-to-face mode and the other delivered using interactive video instruction. Their study indicated that students in the face-to-face course consistently rated the course higher than those in the distance learning course. Ellis and Okpala (2004) evaluated the use of digital technology and software use among business education teachers, specifically the use of digital video cameras and multimedia editing software to edit text, sound, video, computer graphics and animation. They found that younger educators had more of an affinity for incorporating digital video technology into their courses than older educators and that there were some "differences in the use and comfort level of these instructional tools among business education teachers of different ethnic groups" (p. 56).

With the rise in online education the use of and study of various information and communication technologies including video, in its various forms, continues to grow (Katz, 2000). With the explosion of available video on the Internet via sites such as YouTube® (Jones & Cuthrell, 2011), through video content providers such NBC Learn®, and various other video streaming technologies (Hartsell & Yuen, 2006), educators have a wealth of content-rich and in many cases, professionally-edited video to provide to

students in both face-to-face and online modes of delivery.

Additionally, through screen recording and video editing software packages such as Adobe Captivate® and Camtasia®, educators know have powerful tools readily available to make their own quality videos. Although the availability of technologies which enable the use of video in the classroom continues to grow, it is still relevant to note the importance of instructional design factors related to the creation of learning environments which implement these technologies (Fanning, 2008).

Terms such as "video lectures" (Brecht & Ogilby, 2008; Geri, 2011; Lents & Cifuentes, 2009) and "E-lectures" (Jadin, Gruber, & Batinic, 2009) are becoming quite common-place in the literature. Jadin et al. (2009) defines an e-lecture as "a media based lecture including an audio or video recording, synchronized slides, table of contents, and optional complementary information (e.g., external links)" (p. 282). Based upon availability and affordability of video-based technologies and the rise in online education, Geri (2011) suggests that "in the coming years, the use of video lectures as a means for distance learning, as well as for supporting traditional in-class learning is expected to increase" (p. 225).

In her article entitled, "If We Build It, Will They Come? Adoption of Online Video-Based Distance Learning" Geri (2011) suggest that they will indeed come, noting that "video lectures offer students a rich learning experience, which resembles traditional in-class learning" and may possess the potential to "increase both student retention and achievements in distance and blended learning environments" (p. 225). Although the study found that "the majority of students prefer attending traditional face-to-face class meetings. Nevertheless, the availability of videos may improve the achievements of all the students enrolled in a course" (p. 231).

Brecht and Ogilby (2008) conducted a study to evaluate the feasibility and effectiveness of a comprehensive teaching strategy based upon video lectures. The authors suggests that video lectures serve two major strategic purposes: (1) they provide additional teaching time to students who may not fully understand material presented in the classroom lecture and textbook, and (2) they allow classroom coverage of more complex and challenging subject materials since basic concepts can be provided via the video lectures and watched outside of class. The study indicated several interesting findings: video lectures helped students raise their course grades, there was a 71.9% reduction in failing grades among students for whom the videos were available compared to students for whom the videos were not available, the creation of videos means that the lectures for the entire semester are available for preparing for the final exam.

Not only does the debate continue about inclusion of programming in the curriculum, what language or languages to teach, and how to teach them, but with the rise in online education and availability and affordability of video-based technology, a new debate arises: that of teaching programming, a traditionally difficult subject, in an online environment and the potential use of video as an instructional tool to support both online and face-to-face courses. As such, we present the methodology for our study, results, and discussion and conclusion in the following sections.

3. METHODOLOGY

Videos

Demonstration videos were created to correspond to each chapter in the programming text. Adobe Captivate® was used to create videos that captured the desktop of the instructor as he completed each programming exercise. Each video focused on key concepts from the chapter and showed students how to program concepts in C# while the instructor narrated the video and added additional explanation to the programming concept. Videos were approximately 20 minutes in length. Videos were then placed in Blackboard for students to view at their convenience.

Participants

In order to examine potential differences in delivery format, students in two sections of introduction to programming classes were given a survey to determine the way they used the videos during the course and their preferences in relation to the videos. Of the two sections, one section was taught entirely online and the other section was taught face to face and was supplemented with material online. Most students in the classes were in the 18-40 age range and most were male. All students were majoring in computer information systems.

Fifteen of the students were in the face to face section and the other twenty were in the online section for a total of thirty-five participants.

Data Collection

A fifteen question survey was administered to both sections of students. Online students were encouraged to take the survey online. Upon completion, students were provided with a code to turn in for extra credit being offered to participants. Students in the face to face section completed the survey on paper and turned the survey in to the instructor during class. They also were offered extra credit for their participation.

In addition to demographic information, students were asked to report their preferences and opinions on the video. Students were also asked to report how much time they spent using the videos and the textbook as well as their opinions on various aspects of the videos such as length, topic coverage, and usefulness. (Appendix A).

4. RESULTS

Hours watching videos and reading the course textbook

Students self-reported the amount of time per week they spent on average watching the online video demonstrations. For all students, the average time spent watching videos was 2.11 hours per week. For online students, the average time spent watching video was 2.46 hours per week and for face to face students, the average time watching videos was 1.58. In order to determine if a difference exists between online and face to face students, an independent t-test was conducted and found no significant difference between the hours spent watching videos by the two different groups of students; $t(31)=1.676$, $p=.070$.

Overall, students spent more time per week watching videos than they did reading their textbook. The mean time spent reading was 1.81 hours compared to the 2.11 hours spent watching videos.

Usefulness of videos

On the question regarding usefulness of the video, 94% of all students rated the usefulness of the videos at 7 or above on a scale from 1 to 10. The two students that reported lower scores of 1 and 2 were in the online and face to face class respectively. A Mann-Whitney U test was conducted between the online and face to face students. The test showed that there was no statistically significant difference between the reported usefulness of the videos between the two groups; $U=118.00$, $p > .05$.

Preference of textbook vs. online videos

Overall, 82.9% of all students surveyed, rated their preference for the videos over the textbook at a level of 7 or above. By group, the preference of textbook versus video was examined using a Mann-Whitney U test. No significant difference was found in the preference of one instructional media over the other; $U=129.00$, $p=.458$.

Knew Instructor

When asked if the videos helped students know their instructor better, 80% of students in both sections indicated at a level of 7 or higher that they did feel as if they knew their instructor through the videos. In order to determine if a difference exists between online and face to face students, a Mann-Whitney test was conducted and revealed there was no significant difference between the responses of the two groups; $U=131.00$, $p=.511$.

5. DISCUSSION AND CONCLUSION

The transition to teaching programming online included many concerns about how to interact with students and to emulate the demonstration aspects of a face to face class in the online environment. To mitigate this problem, the instructor recorded demonstration videos specifically for use in the online sections of the class, never intending to use them in the face to face class. Students who were in the face to face class had access to the demo portion of the class through class lecture and it seemed unnecessary or redundant to include the videos in the online component of the face to face class. In addition, there were concerns that posting demonstration videos online would encourage students to miss class and watch the videos online instead.

However, the results show that the videos were as important to the face to face students as the online students. Students in both sections watched the videos in similar amounts and rated them equally useful between groups.

Surprisingly, the students who saw the instructor for three hours a week were just as likely to report that the videos helped them "know" the instructor as much as online students, many of which had never met the instructor face to face.

When looking at the overall impact of the video on both students, it is evident that the videos were a valuable addition to both sections of the class. Granted, the creation of videos were time consuming for the faculty member, but the high levels of reported usefulness and the fact that many students depended on the video more than their textbook to understand the concepts in the class, seem to indicate that the time was well spent.

Should the instructor spend the time required to create their own video or use video that accompanies the textbook or found on the internet? Although this was not the focus of this study, the fact that students in the online section were able to make some connection with the instructor is demonstrated in the responses to the question covering that subject. In addition to that, many of the students commented on how important it was that the videos contained all of the concepts necessary to complete the lab assignments and that students found them more useful towards the end of the class when they felt that the book did not adequately cover some subjects.

This exploratory study demonstrated the usefulness of including video in the online as well as face to face sections of introductory programming classes. Future areas of study in this area include the use of video in other CIS topics areas, differences among student learning types, and the difference of prepackaged versus instructor created video. In addition to data on student's self-reporting responses, more quantifiable data from usage statistics as well as some correlation to student success in the course would be useful.

6. REFERENCES

Butler, M., & Morgan, M. (2007). Learning challenges faced by novice programming students studying high level and low feedback concepts. In *Proceedings of the 24th ascilite Conference*, December 2-5, 99-107.

Brecht, H. D., & Ogilby, S. M. (2008). Enabling a comprehensive teaching strategy: Video lectures. *Journal of Information Technology Education: Innovations in Practice*, 7, IIP 71-IIP 86.

Cain, W. P. (1991). Object-oriented programming and the CIS curriculum. *Journal of Information Systems Education*, 3(1), 2-7.

Choi, H. J., & Johnson, S. D. (2005). The effect of context-based video instruction on learning and motivation in online courses. *The American Journal of Distance Education*, 19(4), 215-227.

Donkor, F. (2011). Assessment of learner acceptance and satisfaction with video-based instructional materials for teaching practical skills at a distance. *The International Review of Research in Open and Distance Learning*, 12(5), 74-92.

Dubrowski, A., & Xeroulis, G. (2005). Computer-based video instructions for acquisition of technical skills. *Journal of Visual Communication in Medicine*, 28(4), 150-155.

Dusenbury, L. A., Hansen, W. B., & Giles, S. M. (2003). Teacher training in norm setting approaches to drug education: A pilot study comparing standard and video-enhanced methods. *Journal of Drug Education*, 33(3), 325-336.

Ellis, R. S., & Okpala, C. O. (2004). Evaluation of digital technology and software use among business education teachers. *Journal of Instructional Psychology*, 31(1), 53-59.

Fanning, E. (2008). Instructional design factors as they relate to the creation of a virtual learning environment. *Journal of Interactive Instruction Development*, 21(2), 24-42.

Fincher, S. (1999). What are we doing when we teach programming? In *Proceedings of the 29th ASEE/IEEE Frontiers in Education Conference*, November 10-13, 12a4-1-12a4-5.

Gainsburg, J. (2009). Creating effective video to promote student-centered teaching. *Teacher Education Quarterly*, 36(2), 163-178.

Geri, N. (2011). If we build it, will they come? Adoption of online video-based distance learning. *Interdisciplinary Journal of E-Learning and Learning Objects*, 7, 225-234.

Gotwals, J. K., & Smith, Jr., C. R. (1995). Restructuring programming instruction in the computer information systems curriculum: One department's approach. *Journal of Information Systems Education*, 7(2), 67-72.

Gotwals, J. K., & Smith, M. W. (1993). Bringing object-oriented programming into the undergraduate computer information systems curriculum. *Journal of Information Systems Education*, 5(3), 2-8.

Hartsell, T., & Yuen, S. C. (2006). Video streaming in online learning. *AACE Journal*, 14(1), 31-43.

Jadin, T., Gruber, A., & Batinic, B. (2009). Learning with E-lectures: The meaning of learning strategies. *Journal of Educational Technology & Society*, 12(3), 282-288.

Jenkins, T. (2002). On the difficulty of learning to program. In *Proceedings of the 3rd Annual LTSN-ICS Conference*, August 27-29, 53-58.

Jones, T., & Cuthrell, K. (2011). YouTube: Educational potentials and pitfalls. *Computers in the Schools*, 28(1), 75-85.

Katz, Y. J. (2000). The comparative suitability of three ICT distance learning methodologies for college level instruction. *Educational Media International*, 37(1), 25-30.

Lahtinen, E., Ala-Mutka, K., & Jarvinen, H. (2005). A study of the difficulties of novice programmers. In *Proceedings of the 10th ITiCSE*, June 27-29, 14-18.

Lents, N. H., & Cifuentes, O. E. (2009). Web-based learning enhancements: Video lectures through voice-over PowerPoint in a majors-level biology course. *Journal of College Science Teaching*, 38(2), 38-46.

Martin, E., Evans, P., & Foster, E. (1995). The use of videos in the teaching of accounting. *Accounting Education*, 4(1), 77-86.

Milne, I., & Rowe, G. (2002). Difficulties in learning and teaching programming – Views of students and tutors. *Education and Information Technologies*, 7(1), 55-66.

Mintu-Wimsatt, A. (2001). Traditional vs. technology-mediated learning: A comparison of students' course evaluations. *Marketing Education Review*, 11(2), 63-73.

Robins, A., Rountree, J., & Rountree, N. (2003). Learning and teaching programming: A review and discussion. *Computer Science Education*, 13(2), 137-172.

Russell, J., Russell, B., Pollacia, L. F., & Tastle, W. (2010). A study of programming languages used in information systems and in computer science curricula. *Information Systems Education Journal*, 8(56), 1-15.

Sedaghat, A. M., Mintz, S. M., & Wright, G. M. (2011). Using video-based instruction to integrate ethics into the curriculum. *American Journal of Business Education*, 4(9), 57-76.

Topi, H., Valacich, J. S., Wright, R. T., Kaiser, K., Nunamaker, Jr., J. F., Sipior, J. C., & de Vreede, G. (2010). IS 2010: Curriculum guidelines for undergraduate degree programs in information systems. *Communications of the Association for Information Systems*, 26(18), 360-428.