INFORMATION SYSTEMS EDUCATION JOURNAL

Volume 20, No. 4
September 2022
ISSN: 1545-679X

In this issue:

iscap
Information Systems & Computing Academic Professionals

The **Information Systems Education Journal** (ISEDJ) is a double-blind peer-reviewed academic journal published by **ISCAP** (Information Systems and Computing Academic Professionals). Publishing frequency is six times per year. The first year of publication was 2003.

ISEDJ is published online (https://isedj.org). Our sister publication, the Proceedings of EDSIGCON (https://proc.iscap.info) 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 EDSIGCON conference. At that point papers are divided into award papers (top 15%), other journal papers (top 25%), 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 under 40%.

Information Systems Education Journal is pleased to be listed in the 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. Special thanks to members of ISCAP/EDSIG who perform the editorial and review processes for ISEDJ.

INFORMATION SYSTEMS EDUCATION JOURNAL

Editors

Use of Artificial Intelligence to Grade Student Discussion Boards: An Exploratory Study

Stephen M. Rutner
stephen.rutner@gcsu.edu
Department of Management, Marketing and Logistics
Georgia College and State University
Milledgeville, Georgia 31061 USA

Rebecca A. Scott
scottra@uncw.edu
Congdon School of Supply Chain, Business Analytics & Information Systems
UNC Wilmington
Wilmington, North Carolina 28403 USA

Abstract

There appears to be an increasing acceptance of Artificial Intelligence (AI) across society. As people become more comfortable with AI's use in advertising, basic services and other areas of day-to-day life, the question arises will students also be willing to accept AI in learning situations. Furthermore, what are the impacts on both the student learning and acceptance as well as the effect on the instructor or professor. This paper presents the initial findings of the use of AI in grading students' discussion boards. It presents an initial model of student expectations, discusses potential benefits and drawbacks of AI and presents initial findings from a limited number of classes using AI grading.

Keywords: Artificial intelligence, Discussion boards, Pedagogy, Asynchronous learning, Online learning

1. INTRODUCTION

During the first year of COVID-19 pandemic, many traditional pedagogical tools and methods were stressed as classes were often shifted from face-to-face (F2F) to asynchronous, online (Kafka, 2020). During the early phases of the pandemic in 2020, many students went home for spring break only to not return to the physical classroom until fall semester 2021. This required rapid redesign of learning methods to continue courses and not disrupt students' paths toward graduation (Sanders, 2020).

These rapid changes often forced faculty to incorporate new learning methods to meet the asynchronous nature of these classes. For example, the traditional classroom discussion was not possible unless an online audio-visual conferencing platform was used (e.g., Zoom, MS Teams, WebEx, etc.). If an online platform was either unavailable or not used, the discussion portion of the class would suffer without an alternative. Most faculty are aware and have often used traditional learning management systems (LMS) (e.g., Blackboard, Desire2Learn, etc.) discussion boards as a means to an end in online classes. Furthermore, even by 2010, approximately 85% of universities were using some form of LMS (Chen et al, 2010). Therefore, it was a natural alternative to classroom discussion while adopting to the COVID environment. However, the likely stresses of moving multiple classes from F2F to asynchronous meant that faculty's time was pressed. Many faculty members were overwhelmed early in spring 2020 semester trying to convert content, include all learning activities, operate in a new environment, and

maintain academic standards. These challenges highlighted the opportunities for companies to both reduce manual grading and increase student learning through various new or modified teaching tools.

The use of discussion boards represents one opportunity to improve from traditional uses to an enhanced version. In spite of large amounts of literature supporting the benefits of discussion boards, many faculty members are reluctant to use discussion boards for a variety of reasons. First, they are often concerned that the conversation will not be as "rich or inactive" as F2F or in-class conversations (Smidt et al, 2014). Another issue might be that discussion boards are often not voluntary (i.e., a required number of posts) which will impact the learning (Frey and Wojnar, 2004; Gill 2006). Finally, there is a concern on the difficulty of balancing the interaction between the faculty member and students to enhance learning without dominating the discussion (Dennen, 2005). Each of these valid concerns are in addition to the increased amount of faculty time to read all discussion posts and accurately assess them.

Given these challenges and facing the COVID environment, many faculty members were forced to adopt discussion boards into their classes without significant planning, testing or time beginning in the spring of 2020 to substitute for F2F discussions. Even though there were significant benefits to discussion boards, the challenges forced instructors to seek better processes for their benefit and outcomes to ensure improved student learning.

The purpose of this article is to highlight a specific pedagogical tool that appears to improve learning while simultaneously reducing faculty workload by using AI to help in evaluating student responses on a discussion board. Following this section, the literature review will highlight both use of discussion boards and the specific use of AI in grading students. This will help to develop a theoretical model and research propositions for further testing. Next, an early set of student responses will be presented. Finally, the conclusions and impact of this initial study will help develop the future examination of this subject.

2. LITERATURE REVIEW

The Literature Review is divided into two broad subsections. The first is to review the well-established research of the value of discussion boards in academia and highlight one of the key challenges of evaluating student responses. The second subsection is to identify the less developed, but growing, body of works on the application in AI in academia with a focus on the few recent articles involving discussion boards. The goal of these subsections is to identify the gaps in the literature that require further examination.

Discussion Boards and Evaluation

As mentioned briefly in the Introduction, there is a significant amount of literature about the benefits and disadvantages of using discussion boards in various academic settings. It would be beyond any paper to cover all of that research. Therefore, a brief synopsis of those is included. A detailed review of the more relevant literature revolves around the subject of discussion board evaluation and/or grading.

Since this article previously identified some of the challenges of discussion boards, it was reasonable to also present some of the benefits of using them in various educational situations. Hinton and Bradshaw (2004) did some initial examination of the perceptions of Autonomous Online Discussion (AOD). They found that it was difficult to evaluate the effectiveness. However, they did identify AOD as a "Core element" of online learning and course design. Furthermore, Hew et al (2008) further confirmed that AOD was becoming an "Increasingly common means to facilitate dialogue between instructors and students." They also provided an in-depth history of the overall online literature with a specific focus of challenges and studies applying each potential solution which is discussed later.

The benefits of AOD are numerous and have been thoroughly examined over the last twenty years. First, the unique nature of AOD allows students some flexibility on the timing of posts and time to reflect before replying (Murphy and Coleman, 2004). Another benefit identified by researchers is the actual act of writing, as opposed to verbal response, often helps students to increase learning (Newman et al, 1997; Vonderwell, 2003). Tracy et al (2020) also identified that when performed properly, AOD can increase student engagement and improve learning. Finally, one recent study compared the use of AOD with Zoom and found that students using AODs had increased performance in the class. This implies that properly applied AOD may actually work better than traditional discussion format in the classroom or in a online, real time learning environment (Ackerman and Gross, 2021). All of these studies highlighted some of the key benefits to using AOD as part of an online learning experience. Furthermore, the purpose of

this subsection was not to state the shift to online courses due to COVID was a better overall learning experience, but rather, to identify the positive aspects of AOD. Each of the articles highlight a positive aspect that can be used regardless of F2F or asynchronous learning courses.

Unfortunately, there are a number of negative aspects to using AOD. Using Hew et al's (2008) synthesis of the overall literature, they identified three specific areas or dilemmas that faculty face using AOD: use of grades, number of posting guidelines, and instructor-facilitation. While all three of these areas are of interest to most modern educators, the first is key to this research (Hew et al, 2008). While there are a host of other issues, the key element of student evaluation remains a challenge even post COVID. For example, Dennen (2005) found that if there are not clear expectations given by the faculty member, students' interests and efforts will wane. In other words, the students are not willing to put forth efforts if it did not result in better individual grades. Furthermore, Dennen (2005) found the students benefited when post guidelines were specified (i.e., format, style, length, etc.). Also, faculty grading was a key component to student participation in AOD. The greater the weight of the grade, the more involvement by the student (Cifuentes et al, 1997). Finally, Murphy and Coleman (2004) also found that when students were required to post, the responses often devolved to "Me too" or "I agree" types of general comments. The net effect was that AOD grading created benefits and challenges to the overall learning.

However, the Murphy and Coleman (2004) articles raised a significant point that applies to the faculty member. The increased number of posts requires that every comment must be read, reviewed, contemplated and assigned a grade of some sort. This amount of time to incorporate a systematic process to fairly assigned grades to an AOD can be significant. Furthermore, it can feel somewhat arbitrary to the students. Therefore, one finding to many faculty members that have not used AOD prior to COVID may have been the significant increase in time to move from a F2F discussion evaluation of student comments to an AOD evaluation of much larger amounts of material. A fair amount of literature has been developed about the grading of discussion boards. Pecka et al (2014) states that "Rubrics are often used to facilitate and evaluate student's discussion board postings." In addition to the use of rubrics, they found that the use of AOD help to increase higher order learning in general for the students. Finally, one of their key findings was the inclusion of rubrics further increased the level of higher order learning with AOD. Phillippi et al (2015) also applied national and international competencies within their field to grade discussion. From those competencies, they developed a rubric to apply to each discussion post. The result was clearer guidelines for students and faculty to follow improving the use of AOD. Finally, Hew et al (2008) also stated that the use of rubrics for specific categories of contribution could help students' efforts. The overall result is that there are numerous studies and examples of how to standardize grading through the use of rubrics and the potential benefit for both faculty members and students.

Artificial Intelligence or Auto Grading

While the literature addresses the rubric process, the main benefit is to normalize the grades for the students, but it does not significantly reduce the workload on the faculty member. The challenges of grading an open-ended student work can be time consuming (Tsai, 2012). Furthermore, some faculty are likely to avoid giving open-ended assignments due to the time required to grade them (Tsai, 2012). A possible solution to this is the use of automatic or AI grading. But, some faculty were also reluctant to use any form of automation due to their belief that computers were not sophisticated enough to replace human judgement in grading (Bridgeman and Quinlan, 2009). Yeh et al (2007) also found that automated grading systems did not do an adequate job of dealing with higher level and/or critical thinking. This is an interesting finding and may be due to the level of computer sophistication or the lack of common use of AI in society in 2009. However, the initial literature search for AI or automated grading even in 2021 created an interesting result. The top 100 papers gathered by the library search engine, Galileo, had less than ten papers that involved academic applications of grading. Rather, the medical use of AI of grading various symptoms, diagnosis or treatments accounted for over 75% of the results. The implication that widespread use of AI may be much more advanced in the medical community versus academia. Furthermore, the majority of the academic literature trends toward specific computer tools, languages, engineering approaches and applications to improve the process rather than the impact on students and faculty members.

Regardless of the amount of AI usage in academic literature, automatic grading offers a number of potential benefits to both faculty members and students. Tsai et al (2012) did find that while not

perfect, AI grading did offer the following potential benefits: consistency between students, rapid grading, never gets tired, and provides immediate feedback. To address some of the specific shortcomings of AI grading, Kyrilov and Noelle (2014) identified a theoretical framework to improve AI grading using the case-based reasoning (CBR) approach. Figure 1 – CBR Methodology presents the learning process for computer grading. The goal of their process was to develop the AI's ability to improve its grading. Finally, they stated that CBR was not widely adopted within the educational community, but CBR had the ability to assist instructors with grading of open-ended student works. It should also be noted that they foresaw the use of CBR in the medical community nearly ten years ago.

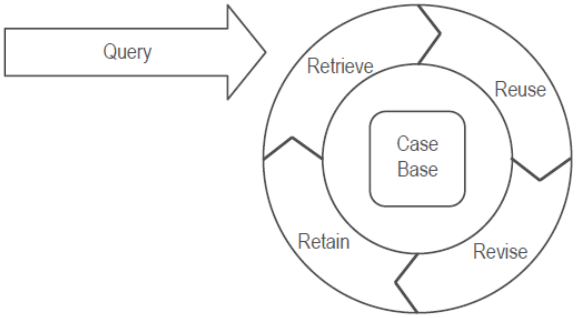


Figure 1 – CBR Learning Framework

Not surprisingly, in the nearly ten years since Kyrlov and Noelle's work, advances have been made in grading open-ended responses by AI. Liu et al (2021) identified the tedious nature of grading these types of answers and applied an automated grading method using multiway attention networks. Their experiments demonstrated superior results compared to six other grading methods. The overall results highlight the ever-increasing power and accuracy of the AI grading systems available to faculty members.

Delgado et al (2020) further identified the advantages of modern AI embedded within a LMS (Pearson MyEnglishLab) to provide specific and tailored feedback to students. In their paper, they demonstrated how the AI's comments were specific and designed to help students identify and improve weak areas of their answers.

As the use of AI grading progresses, current studies are exploring the use beyond simple responses in AOD. Rather, can a different form of input into the AOD be analyzed by the AI. Ghoneim and Elghotmy (2020) studied the use of AI boards for input into the grading system. While their study differed from traditional use of AOD, it did highlight the potential for creative uses for AI. Furthermore, they were one the few studies that specifically stated that the use of AI could be "Fun" for the student if creatively applied.

It is clear that the literature presents a solid overview of the challenges and benefits of the use of AOD. Additionally, there appears to be a growing use of AI in various aspects of the educational community. The increasing sophistication of AI grading has helped to alleviate some of the drudgery and inconsistency of AOD. However, most of the literature was focused on the pros/cons, methods, technical aspects, applications and outcomes of using AI. Very little focused on the reaction from students as well as their learning.

3. THEORETICAL MODEL AND RESEARCH PROPOSITIONS

Based on the previous research of the concept of AI grading, there are numerous potential impacts on student discussion quality, quantity, and learning. The traditional interaction between faculty members and students in an AOD are limited by the asynchronous nature of the process. Figure 2 represents a typical student and instructor interaction process.

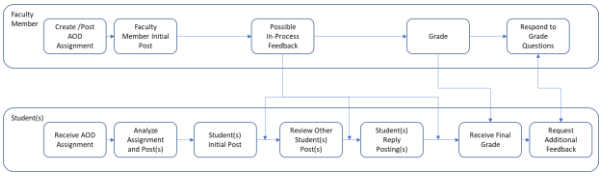


Figure 2 – Traditional AOD Interaction between Faculty and Student(s)
Note: a full size figure in the appendix

The figure highlights the typical pedagogical process on the part of the faculty member. Once the instructor chooses to incorporate an AOD, he or she creates some sort of assignment followed by an initial post containing instructions or questions to beginning the discussion. The instructor then would typically read some posts and may provide feedback at various times through the process. Finally, he or she would grade the students' individual posts and assign a grade. This would be followed by the likely questions from various students concerning grading. Most of the process is linear and involves limited interaction with the student. A key constraint is the faculty member's time to provide timely feedback to the students. Also, the

students' post must be published to the board before the faculty member can provide feedback. These limitations force the student to either accept their initial posts without change or to create more posts that need to be evaluated yet again by the faculty member. This creates even more work and further limits faculty time to evaluate posts.

From the student side, the figure demonstrates the process from their view. It is also linear from receiving the assignment, to making initial post(s), reviewing other students' submissions, possibly receiving feedback, and then, making a final post(s). This is also followed by receiving their grade for the assignment which may trigger a question to faculty member about that grade. A key point is the limited interaction between student and instructor. There may be feedback, but it always lags from the initial post. Often, it may be days until the professor is able to catch up to the numerous posts in the discussion board. Therefore, a student is often left with little to no feedback during the traditional process.

Based upon Kyrilov and Noelle's framework (2014, Figure 3), rapidly received feedback could improve the students' posts, the level of discussion and overall quality of the AOD. Figure 3 presents an adapted version of their model to integrate into the traditional AOD interaction model (Figure 2).

The adapted process assumes that immediate feedback is available to the student through the use of AI grading. The student prepares an initial draft of his or her post. The AI grading would provide either instantaneous or immediate feedback during the draft process. The student then likely revises and improves the post a number of times until he or she is ready to submit it as their submitted post. The net result is likely a vastly improved overall product that has encouraged and motivated the student to think more deeply about the subject and increase overall learning. This occurs with all students' posts nearly simultaneously with little to no faculty interaction.

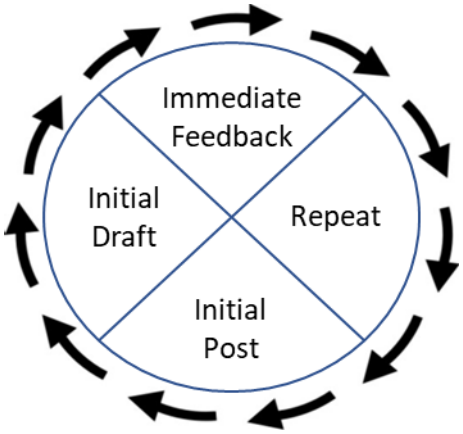


Figure 3 – Adapted Immediate Feedback Model

By integrating the adapted immediate feedback model into the traditional AOD interaction model, an improved AI grading model is displayed in Figure 4 – Incorporating AI Grading into AOD. This model presents the changes in the interaction between the instructor and students by including the AI feedback into the process. First, it demonstrates the timelier feedback from the AI grading. Furthermore, the adapted immediate feedback model interacts with both the faculty member and students' tracks. By providing immediate feedback, the AI acts as a surrogate for the faculty member. It also relieves some of the pressure on the faculty member to try and provide timely feedback. The AI grading becomes a linking feature between the students and the faculty member.

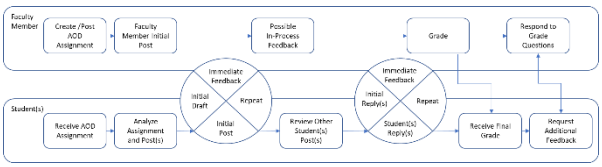


Figure 4 – AI Feedback Modified AOD Process Model
Note: a full size figure in the appendix

Based upon the adapted model using AI feedback or grading, there are a number of research questions that are designed to fill the gaps in the literature. Each of the propositions identifies key issues beyond the software mechanics of AI grading, but rather focuses more on the potential impacts and benefits for both the students and faculty members.

P1: Students benefit from immediate or real-time AI generated feedback.
P2: AI grading and feedback is adequate to replace faculty member inputs during the discussion board posting cycle.
P3: AI grading and feedback encourage students to think more deeply about the topic.
P4: Students will prefer the AI grade to the instructor's grading process.

These first four propositions focus on the potential pedagogical benefits of using AI grading and/or feedback. The assumption is that student learning benefits from any type of immediate feedback. The challenge is that in a real-world setting it is unlikely that faculty members are able to provide real-time or near instantaneous feedback. Furthermore, with the ever-increasing AI sophistication, the current state of AI feedback and grading is adequate to replace instructor comments at least during the discussion board process. However, this is not to imply that AI grading is fully able to provide final grades at this point. Finally, near simultaneous feedback encourages the students to review, revise and resubmit their initial and follow on posts which should encourage deeper thoughts on the topic and an increased learning level for the material.

P5: Artificial constraints in the AI system reduce the students' perceived benefits of using AI graded AOD (e.g. word limits, requirements to post a question vs statement to begin, lack of discussion board structure, etc.)
P6: Immediate feedback will reduce stress on the students throughout the posting process.
P7: An outside vendor (i.e., not university integrated LMS) will create issues for the students – cost, technical issues, ease of use.
P8: An outside vendor's desire to attract customers will create hidden benefits to the student.

The second group of propositions focus more on the mechanics of an AI grading/feedback system. The AI system is likely to have some limitations due to the programming. These may include, but are not limited to, word counts, required formatting or use of questions, various discussion board structure, etc. Furthermore, the large LMS that universities are using do not incorporate AI at this point. This necessitates additional steps, time, effort, and cost to the students and faculty member to employ the AI grading system. Therefore, both faculty member and students have to weigh the tradeoffs of using the system. Also, since a third party vendor is providing the AI solution, there is an implied belief that the company will constantly work to improve the product due to competition in the marketplace which may reduce disadvantages to the students and faculty members that exist at the time of this study. The net result of the second group of propositions is that the improved AI product should benefit the students and faculty member to include the pedagogical propositions (P1-P4).

4. METHODOLOGY

To conduct an initial examination, an AI system was chosen and applied with a student sample. Georgia College and State University used Packback across three traditional asynchronous, online, graduate classes during the spring 2021 semester. The classes were all part of a single Master program. Two different faculty members were the instructors of record. Also, the three classes were three different courses across two differing cohorts of students. All three classes had been taught before using traditional discussion boards; so, the switch to an AI grading/feedback board was a minimal change to each of the existing courses. In other words, the test classes were not part of the reaction to COVID nor involved other significant pedagogical changes. Finally, all of the students were in their second or fifth semester of the five semester program and had used a traditional discussion board as a part of the integrated LMS in a previous class(es). The faculty members believed this group of students would provide a fairly wide cross section of views and experiences. Also, with the students' experiences with traditional discussion boards, they would be excellent judges of the benefits and disadvantages of use the Packback AI system throughout the semester. Finally, since this was an exploratory study, a simple 29 question survey was offered to the students for a small amount of extra credit at the very end of the semester. The majority of questions were five point Likert scale responses about the Packback system. The responses were anonymous, but the students' identification numbers were collected in a separate file to apply credit for completing the survey.

5. PACKBACK

Packback is an online discussion board platform. It was chosen based upon an initial recommendation from faculty that were using it with undergraduate students at another university. On the Packback website home page, they state that use of their AOD product will

"Inspire self-motivated, critical thinkers through inquiry-driven discussion." They even provide comments that their system will improve the learning and grading outcomes for students, create a more rigorous discussion and reduce the workload on faculty members (Packback, 2021).

A goal of the Packback system is to improve both students' discussion and easy faculty workloads. These are two of the critical issues identified in previous studies as advantages. However, the question arises of how does Packback work and how effective is it AI grading system.

One key difference between a traditional discussion board and Packback's system is the use of an AI grading process. The first of the two major parts of the AI grading in Packback was when the students are drafting their post. Packback provides a number of helpful items to encourage them to be more complete with their answers. Figure 5 – Student New Post Screen provides an example of what a student would see while drafting a post. The Instant Feedback column on the right side of the students' screen helps to guide the students' responses. A key item is the student is assigned a "Curiosity Score" during this process. While it is in draft mode, the score is displayed as a range. For example, the example post below is low with a 31-70 potential score. In two of the classes, an 80 was required to have the post count as a valid post. Also, the system helps the students to not focus only on the curiosity score, but encourages them to fix grammatical errors, add links to relevant material, include videos/pictures/charts and checks for plagiarism both inside the Packback program and outside. Finally, as soon as the students finishes the post, he or she will receive their curiosity score.

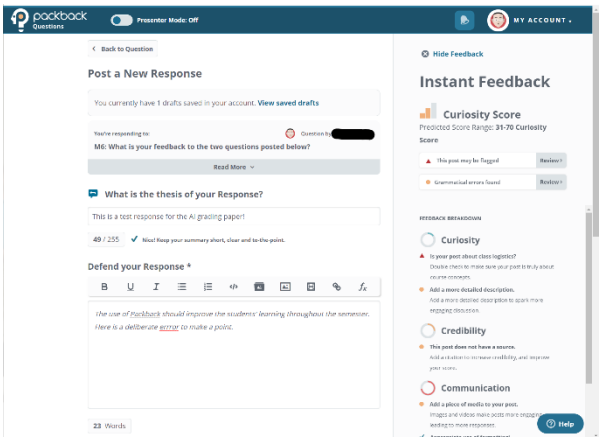


Figure 5 – Student Post Screen
Note: a full size figure in the appendix

This score is provided entirely by Packback and does not involve the faculty member at this point. The score is derived by a Packback algorithm based on a combination of the students' presentation, creditability and effort/depth of the individual post. Without having an entire discussion of the AI process, the score can be summarized as applying an algorithm that correlates high activity, highly curiosity of highly driven member posts. The scores are valuated against other students' posts not only within their class's discussion board, but compared to all other students using Packback. Finally, the algorithm checks for credibility of the post based on relevant and reliable sources that are used to defend the students' main points. This process helps to address some the common concerns about AI grading reliability.

To continue with the example, one of the faculty members in the test classes required a minimum curiosity score of 80 for the post to count as one of the three required postings for the weekly discussion. The score itself was not used as the sole grade for the students' discussion board results throughout these test classes. However, due to the nature of graduate students, the faculty members observed some "friendly competition" among the students to continuously improve their discussion posts' curiosity scores. Figure 6 – Student Post on Discussion Board shows what students viewed after posting their work.

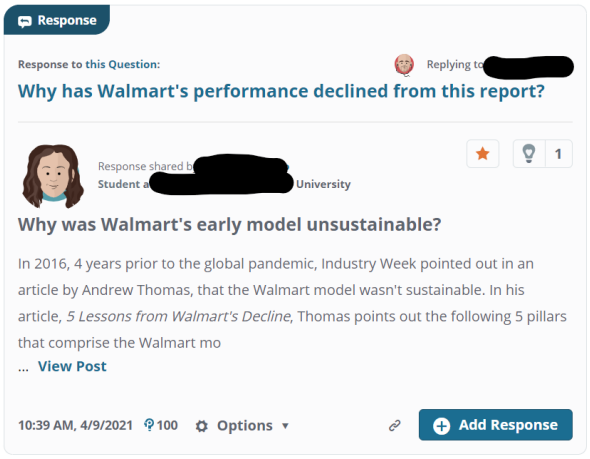


Figure 6 – Student Post on Discussion Board
Note: a full size figure in the appendix

This example was taken from an actual reply from one of the classes. It was chosen as an example for a number of reasons. First, since it was fairly long (four full paragraphs), Packback abbreviated

it and had a "View Post" to see the complete post. This allowed the shorter version to be screenshot more easily and demonstrate a number of key points in one figure. First, you can see the student was replying to another student's post. Also, on the bottom row, the student's final curiosity score was a 100. A key point about the system was the ability for students to edit and re-edit their posts. Assume that the example draft post example ended up scoring a 70. The student could then go back, re-edit it and repost immediately. The new score may be an 85 or 90. If satisfied, the student could leave it as his or her post, or if unsatisfied, he/she could re-edit again in an attempt to increase the score. The resulting iterative cycle created many very highly scored posts. The real benefit was not the high curiosity scores, but rather, students reviewing and revising their work to create better posts which helped in the learning process.

Finally, looking at the top right corner there is a star and a lightbulb. A star was if the faculty member featured this as a significant post in the discussion, and the lightbulb, or "sparks," represent posts that the faculty member or other students "sparked" their curiosity. These little items added a different type of feedback and provided a useful tool to ensure especially good posts were read by the entire class.

There were some significant drawbacks to using Packback. First, there is an additional cost to the students to purchase use. The pricing model continues to change, but it was approximately $20 per class during the test semester. Also, Packback is not fully integrated into the various LMS. Therefore, faculty have to transfer grades between the systems. Packback did provide a very good tool to download scores into Excel spreadsheets with numerous options. A unique challenge with Packback was the inability to subdivide the course discussions into modules. The entire semester had to be performed on the same discussion board (there were a number of tricks to minimize this: post naming conventions, feature postings, etc.). Finally, Packback was another system that students and faculty members had to learn and operate beyond the university's LMS.

The overall result was that Packback is not a revolutionary new system. However, it is clearly an evolutionary step in applying AI to the grading and feedback portions of discussion boards. The faculty were encouraged enough by the anecdotal successes during the semester to use it again in the fall 2021 semester with the same program's students.

6. FINDINGS

The initial survey resulted in 72 useable responses. Table 1 presents the demographic results for gender, class, etc. It should be noted that 100% of the students were in the Master of Logistics and Supply Chain Management program in this study. The demographics are fairly representative of a group of graduate students in the field. It leans a little towards the male side of respondents. Two of the three classes were more represented, but that also aligns some to the class sizes. Since the students are graduate business majors, it is also reasonable that the PCs were much more common that Macs in the sample. Next, the grade distribution is reasonable given both the graduate level and split between first- and second-year students and the likelihood of higher achieving students being a little more likely to provide feedback. Finally, the response by 72 students out of a total population of 95 resulted in a 75.8% response rate. It should be noted that a small number of students could have been in two courses simultaneously but were limited to responding in only one class.

Variable	Responses		
Gender	Male = 59.51%	Female = 40.85%	
Year Group	1st Year = 46.48%	2nd Year = 49.30%	Other = 4.23%
Course	Inventory = 37.50%	Strategy = 47.22%	Purchasing = 15.28%
Computer	PC = 85.92%	Mac = 14.08%	
GPA	4.00 = 49.30%	3.50-3.99 = 22.54%	3.00-3.49 = 19.72%
GPA (cont.)	Below 3.00 = 2.82%	Not Provided = 7.04%	

Table 1 – Summary Demographics

The use of Likert scale survey questions attempted to evaluate the effectiveness of the Packback AI towards the students. One of the key differences between this work and previous studies was to collect student feedback about the use of AI grading and/or feedback. A series of specific questions asked questions based upon the research propositions. For example, questions regarding the value of the immediate scoring and other areas addressing the pedological impact were included in the survey. These questions were aimed at the first four propositions. Also, there were numerous questions about the specific process to include strengths and weakness of the system to examine the second group of research propositions. Finally, there were some duplicate questions to check student response consistency. A summary of the key results is included here; however, providing all of the questions here would be redundant and too lengthy.

To begin with the pedagogical impacts of AI grading, the first key question was "Did the students like the ability to receive immediate feedback?" The response was an overwhelming yes. Over 97% of the respondents answered that they either strongly agreed or somewhat agreed (70 of 72). Only two students were neutral or opposed. When asked specifically about AI grading portion, the students were still very positive. Figure 7 – I Liked the AI Grading presents the students' responses to this question (5-Strongly Agree to 1-Strongly Disagree). Although the result was not as strong as the immediate feedback question, 83.3% of the respondents had a positive view and only 9.7% were opposed. The combination of the immediate feedback and AI grading were supported by the vast majority of the students across all classes, both genders and regardless of GPA.

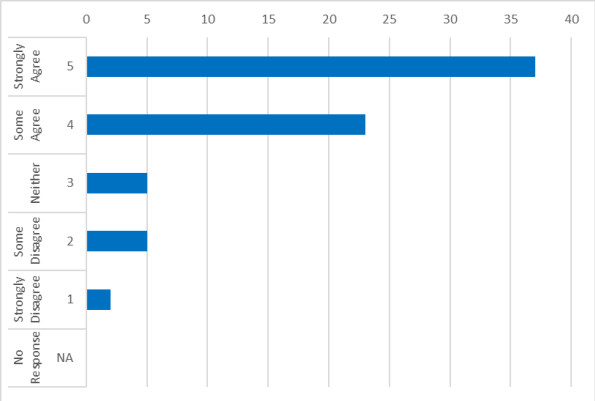


Figure 7 – I Liked the AI Grading

To continue to examine the pedagogical impacts, the students were asked to evaluate their view of AI grading compared to the faculty members' grading system. Here, there was a cross section of answers. The students did not have a strong opinion on which, if either, was better. Figure 8 highlights this finding. It is interesting that the students were not willing to completely trust the AI system. However, clearly some students preferred the AI compared to the faculty members' grading processes. There are many possible causes for this finding and they could be a subject to an entire paper in itself. However, some of the main comments included a lack of clear understanding of how the AI system worked and how the professors would incorporate the grading into their classes. It should also be noted that the faculty members used the results from the AI grading differently in their individual classes. Finally, some of the variation is clearly due to the belief that a minimum score on the AI or curiosity score would earn the student full credit for the assignment which was not the case.

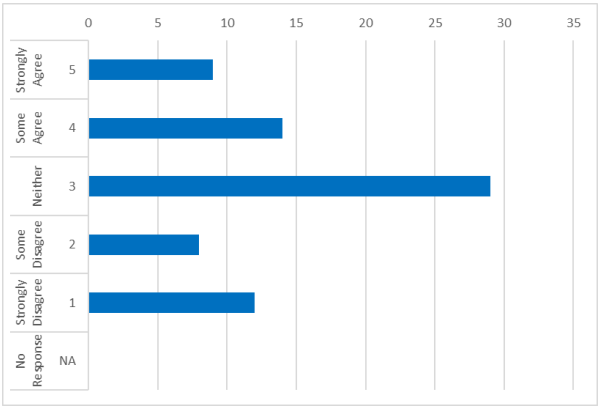


Figure 8 – I Prefer Instructor verse AI Grading

One of the key goals of the use of AI was to encourage deeper thought and learning in the AOD. Figure 9 demonstrates that 55.6% of the students reported a positive impact. It should be noted that the wording of the question did not include that the use of the AI could have had a negative impact. Therefore, the fact that a majority of the students responded that it increased their learning experience by using the AI system. This is a tremendous benefit to the overall class. Furthermore, there are a number of second order effects that may have not been obvious to the students. First, if over half were improving their posts and learning, then the remaining students were reading more well-developed submissions and by default would have an increased learning experience. Also, even if a student did not feel his or her learning was better, the level of competition within the course likely encouraged them to improve their work and hence their individual learning. Finally, the impact on learning was likely the most significant on the students in the middle of the grade distribution. The very high and low achieving students may not have gained as much due to their already being on the extreme ends of the spectrum. These are additional areas for future research. However, the initial finding was strong enough for the faculty to continue to use AI.

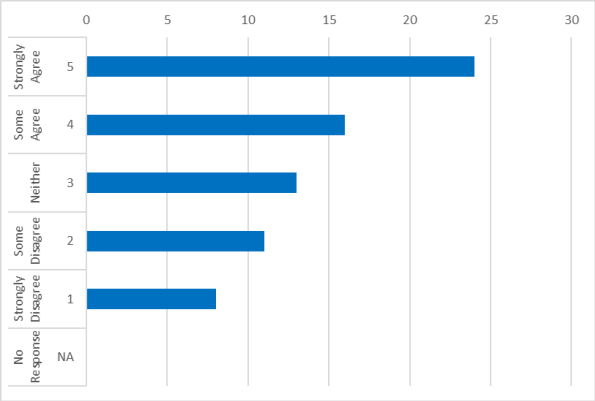


Figure 9 – The AI Encouraged Me to have Deeper Thoughts

In terms of the next group of research propositions, there were a number of questions about the mechanics. A summary table condenses these findings due to space limitations. Table 2 – Process Results for Using Packback highlights most of the key findings. It should be noted that since the specific Packback program was used, the findings may or may not apply to other AI AOD programs. The table is organized with a shortened version of each question followed by the students' responses. The responses are organized by positive, then neutral, and finally negative based on the questions. For example, the first question was "Is the use of Packback more difficult since it was not incorporated into the LMS?" The majority of the students did not think it was more difficult; so, the "No" finding is a positive for this question.

Question	Responses (not all add 100% due to non responses)		
More difficult since not in LMS?	No – 56.66%	Neutral – 18.06%	Yes – 15.28%
I gamed the system for grade?	No – 52.50%	Neutral – 18.06%	Yes – 19.44%
I had technical problems?	No – 94.45%	Neutral – 1..39%	Yes – 2.78%
Use of AI's features was fun?	Yes – 62.50%	Neutral – 29.17%	No – 8.3%
The cost was worth the benefits?	Yes – 58.33%	Neutral – 23.61%	No – 18.05%
It was easier to use that the LMS?	Yes – 61.11%	Neutral – 18.06%	No – 20.83%
Overall satisfied with AI?	Yes – 83.31%	Neutral – 5.56%	No – 11.11%

Table 2 – Process Results for Using Packback

For all of the specific, mechanical types of questions, the majority of the students had a positive response. There were virtually no technical problems with the AOD. The two students that did have issues both were using VPNs to block their identities which when turned off, the Packback website worked fine. A concern with any AI grading system is the students will game the process for a better score. For example, Packback allows you to put a link into your response which will help your curiosity score. However, the link could be very much off topic and the student still gets the points. Therefore, it was interesting to see that almost 20% of the students did game their posts at some time during the semester. This both highlights that AI is not perfect at this point and faculty member involvement is still needed. Next, the students enjoyed using the AI system. While not an extremely important point, a positive experience using the system will likely encourage additional use when compared to a negative experience. One of the faculty members' key concerns was cost. Students already pay for a LMS and have premium pricing in the program. The majority of students did think Packback was worth the additional cost. However, the written comments did state that since the costs was outside the university, some students' employers would not reimburse it which led to their dislike. Finally, the summary question of overall satisfaction was very high at 83% of the students. These findings coupled with the pedagogical results highlight that the AI grading and feedback had a successful proof of principle test in the spring semester.

7. CONCLUSIONS, RECOMMENDATIONS, AND FUTURE OPPORTUNITIES

As previously stated, this was an initial test of the Packback. AI grading and feedback systems were considered a qualified success based on both the faulty members' and students' feedback. The majority of the propositions were supported with summary data. The students reported they learned more, applied more effort and were satisfied with the AI system. Faculty members were also pleased in general with the clear improvement with student work but were not as positive due some of the technical items due to the stand-alone nature of the AOD, content organization abilities, and cost. However, as stated before, the overall positive aspects were more than enough to adopt the Packback AOD again in the upcoming fall semester.

Based on the use of Packback for the first time, there were a few clear learning points that will be

applied before the next iteration. These items are shared as recommendations for any faculty member planning use either Packback or another AI grading system. First, both the syllabus and faculty member should clearly articulate exactly how the AI scores will be incorporated into the overall grading scheme. The key is not whether the grade is all AI based, a hybrid or all faculty member derived, but rather which will be used. That will help to clarify the students' expectations. The authors recommend a hybrid that is given to the students at the beginning of the class (i.e., 50-50% faculty-AI scores based upon …) Next, the benefits and challenges of using the AI should be stated at the beginning of the semester. For example, one of the challenges of Packback is the lack of modules or submodules to separate different discussions. This will likely be addressed in future updates of Packback. In the spring classes we developed a numbering system that aligned LMS module numbers and specific posts were to include the number in the title. Again, it was a simple item, but helped to provide clarity. Finally, faculty member expectations should be restrained. There was a minor reduction in workload; however, each student post still needed to be read and evaluated. The primary benefits were improved student posts and more timing flexibility of when to review the postings. The asynchronous portion of the board and that the AI will fill in for the instructor should be communicated at the beginning. The instructor should clearly indicate that he or she will be reading all the posts to ensure that students are not trying to game the system with unnecessary photos, videos, links, etc.

Also, due to the introductory nature of this paper, there are a number of findings that should be more rigorously tested. For example, the sample size was not large enough to do specific demographic tests beyond a cursory evaluation. Another area for future examination is the impact on undergraduate students and use in a F2F class. It is likely that the findings would remain the same; however, the differing nature of these groups and class settings might have significant impacts on the results. Another key point is that the AI software continues to evolve. A future examination of the ability of students to game the system when less faculty review is conducted would help instructors to moderate their time and effort in grading.

The last point about improving nature of the software is a key closing point. As more companies enter the field and traditional LMS recognize the benefits of AI grading, it is likely the quality and options for AI grading both inside AOD and in other areas will improve dramatically over the next ten years. Faculty members should begin to realize the potential pedagogical and workload benefits. Just as PowerPoint changed classrooms 20+ years ago, and real-time media is reshaping them today, AI will change the learning experience over the next few years. The same question arises of how should a faculty member apply this new technology to maximize it benefits while minimizing its weakness for both students and instructors.

8. REFERENCES

Ackerman, David S. and Barbara L. Gross (2021). "Synchronous Online Discussion Board as a Primary Mode of Delivering Marketing Education: Responding to the Covid-19 Pandemic and Beyond." Marketing Education Review, March, 1–10.

Bridgeman, Brent & Thomas Quinlan (2009). "Differences between Humans and Computers for Evaluating and Teaching writing." In Proceedings of 2009 International Conference on ELT Technological Enterprises, pp, 3-24.

Chen, Pu-Shih Daniel. Amber D. Lambert and Kevin R. Guidry (2010) "The Impact of Web-Based Learning Technology on College Student Engagement." Computers & Education, 54, 1222-1232.

Cifuentes, Lauren, Karen L. Murphy, Rhoda Segur, and Sailaja Kodali, (1997). "Design Considerations for Computer Conferences." Journal of Research on Computing in Education, v. 30(2), pp. 177–201.

Day, J., H. Lou,. and C. Van Slyke. (2004). "Instructors' Experiences with Using Groupware to Support Collaborative Project-Based Learning." International Journal of Distance Education Technologies. 2(3). 11-25.

Delgado, Heloisa Orsi Koch, Aline de Azevedo Fay, Maria Jose Sebastiany and Asafe Davi Cortina Silva (2020). "Artificail Intelligence Adaptive Learning Tools: The Teaching of English in Focus" Brazilian English Language Teaching Journal, 11(2), 1-19.

Dennen, Vanessa Paz (2005). "From message posting to learning dialogues: Factors affecting learner participation in asynchronous discussion." Distance Education, 26, 127–148.

Douglas, Tracy, Allison James, Louise Earwaker, Carey Mather and Sandra Murry (2020). "Online Discussion Boards: Improving Practice and Student Engagement by Harnessing Facilitator Perceptions." Journal of University Teaching & Learning Practice, 17(3), 1–14.

Frey, Barbara A., and Linda C. Wojnar. (2004). "Successful Synchronous and Asynchronous Discussions: Plan, Implement and Evaluate". Educause Information Resources Library. Accessed at http://www.educause.edu/ir/library/pdf/MAC0426.pdf on 9 Jun 2021.

Ghoneim, Nahed Mahammed Hahmoud and Heba Elsayed Abdelsalam Elghotmy (2020). "Using an Artificial Intelligence Based Program to Enhance Primary stage Pupils' EFL Listening Skills," Sohag Universit Journal of Education, 83(1), 1-32.

Gill, Grandon (2006). "Asynchronous Discussion Groups: A Use-based Taxonomy with Examples." Journal of Information Systems Education, v. 17, n. 4, p. 373–383.

Hew, Khe Foon, Wing Sum Cheung, and Connie Siew Ling Ng (2010). "Student Contribution in Asynchronous Online Discussion: A Review of the Research and Empirical Exploration." Instructional Science, v. 38, n. 6, p. 571–606.

Hinton, Leone and Julie Bradshaw (2004). "Benefits of an Online Discussion List in a Traditional Distance Education Course," Turkish Online Journal of Distance Education, vol. 5, no.3, pp. 8-16.

Liu, Tiaoqiao, Wenbiao Ding, Zhiwei Wang, Jiliang Tang, Gale Yan Huang and Zitao Liu (2019). "Automatic Short Answer Grading via Multiway Attention Networks" Working Paper, https://arxiv.org/abs/1909.10166 accessed June 2021.

Kafka, Alexander C. (2020). "Shock, Fear, and Fatalism: As Coronavirus prompts College to Close, Students grapple with Uncertainty." The Chronical of Higher Education, Mar. 12, https://www.chronicle.com/article/shock-fear-and-fatalism-as-coronavirus-prompts-colleges-to-close-students-grapple-with-uncertainty/ accessed Jun 2021.

Murphy, Elizabeth and Elizabeth Coleman (2004). "Graduate Students' Experiences of Challenges in Online Asynchronous Discussions." Canadian Journal of Learning and Technology, *30*(2), spring.

Newman, D. R., Chris Johnson, Brian Webb and Clive Cochrane, (1997). "Evaluating the Quality of Learning in Computer Supported Co-operative Learning." Journal of the American Society for Information Science, 48. 484-495.

Packback. (2021). https://www.packback.co/ accessed January 2021.

Pecka, Shannon, Kendra Schmid and Bunny Pozehl (2014). "Psychometric Testing of the Pecka Grading Rubric for Evaluating Higher-Order Thinking in Distance Learning." AANA Journal, 82(6), 449–456.

Phillippi, Julia C., Mavis N. Schorn, and Tonia Moore-Davis, (2015). "The APGAR rubric for scoring online discussion boards." Nurse Education in Practice, v. 15, n. 3, p. 239–242.

Ratcliff, Jennifer J., Korrine I. Minister and Chelsea Monheim (2014). "Engaging students in an online format during the COVID-19 pandemic: A jury voir dire activity." Scholarship of Teaching and Learning in Psychology, February.

Sandars, John, Roaqueel Correia, Mary Dankbaar, Peter de Jong, Poh Sun Goh, Inga Hege, Ken Masteers, So-Young Oh, Rakesh Patel, Kalyani Premkumar, Alexandra Webb and Martin Pusic (2020). "Twelve tips for Rapidly Migrating to Online Learning during the COVID-19 Pandemic," MedEdPublish, April 29, https://doi.org/10.15694/mep.2020.000082.1 accessed June 2021.

Smidt E, McDyre B, Bunk J, Li R, Gatenby T. (2014). "Faculty Attitudes about Distance Education." IAFOR Journal of Education, 1(2) 181-209.

Sull, Errol C., (2014). "Discussion Boards, Grades, and Student Involvement." Distance Learning. 11(3), 55-57.

Tsai, Min-hsiu (2012) "The Consistency Between Human Raters and an Automated Essay Scoring System in Grading High School Students' English Writing," Action in Teacher Education, (34)4, 328-335, DOI: 10.1080/01626620.2012.717033

Vonderwell, Selma (2003) "An Examination of Asynchronous Communciatin Experiences and Perspectives of Students in an Online Course: A Case Study," The Internet and Higher Education, 6(n), (first quarter), 77-90.

Walker Kristen, Mary T. Curren, Tina Kiesler, H. Bruce Lammers and Jamie Goldenson, (2013). "Scholarly Networking Among

Business Students: Structured Discussion Board Activity and Academic Outcomes." Journal of Education for Business, 88(5), 249–252.

Yeh, Y., Liou, H. C., & Yu, Y. T. (2007). "The Influence of Automatic Essay Evaluation and Bilingual Concordance on EFL Students' Writing." English Teaching and Learning, *31*(1), 117-160.

Appendix:

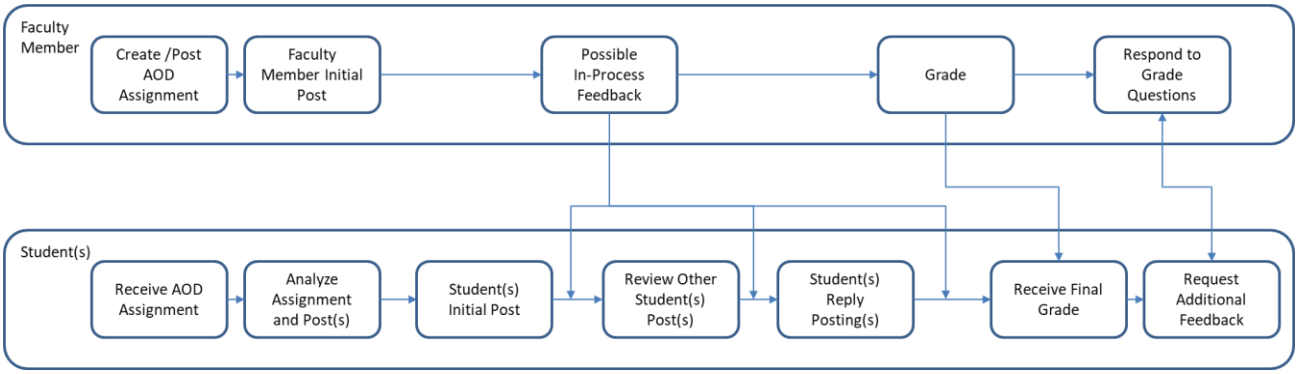


Figure 2 – Traditional AOD Interaction between Faculty and Student(s)

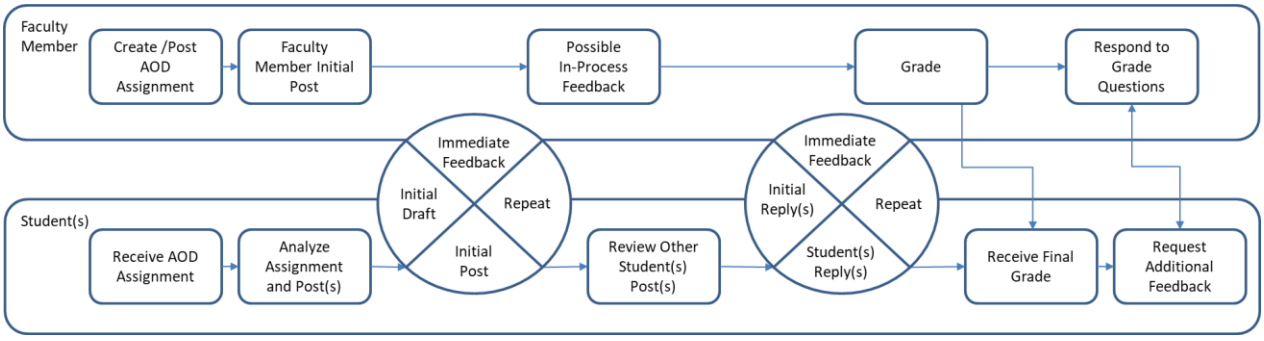


Figure 4 – AI Feedback Modified AOD Process Model

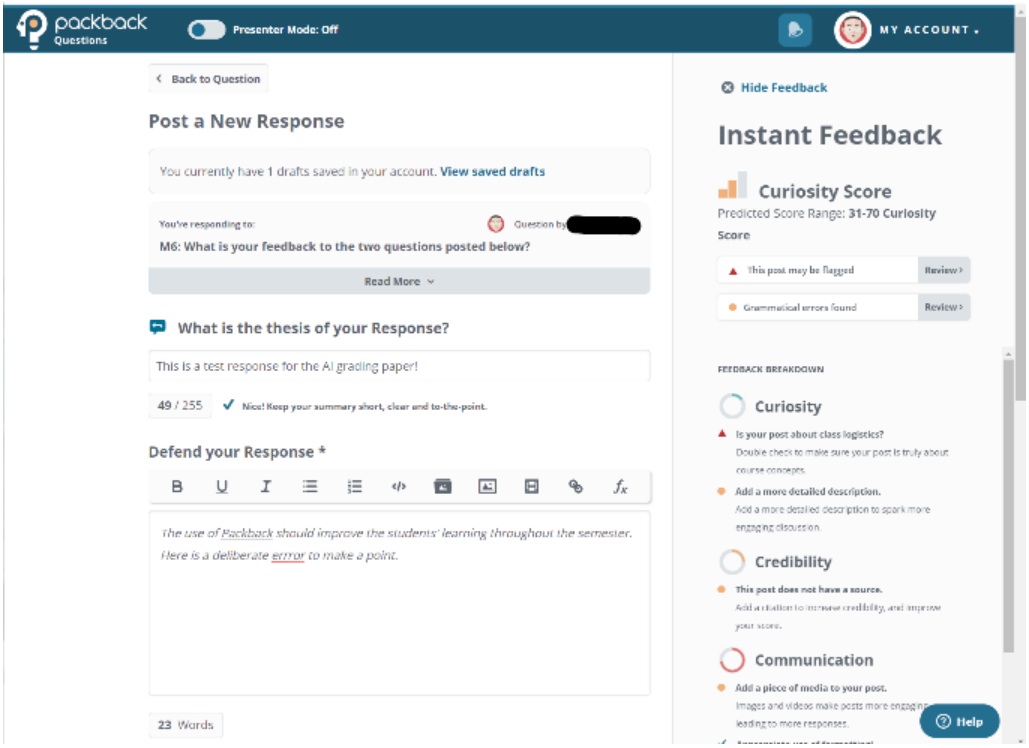


Figure 5 – Student Post Screen

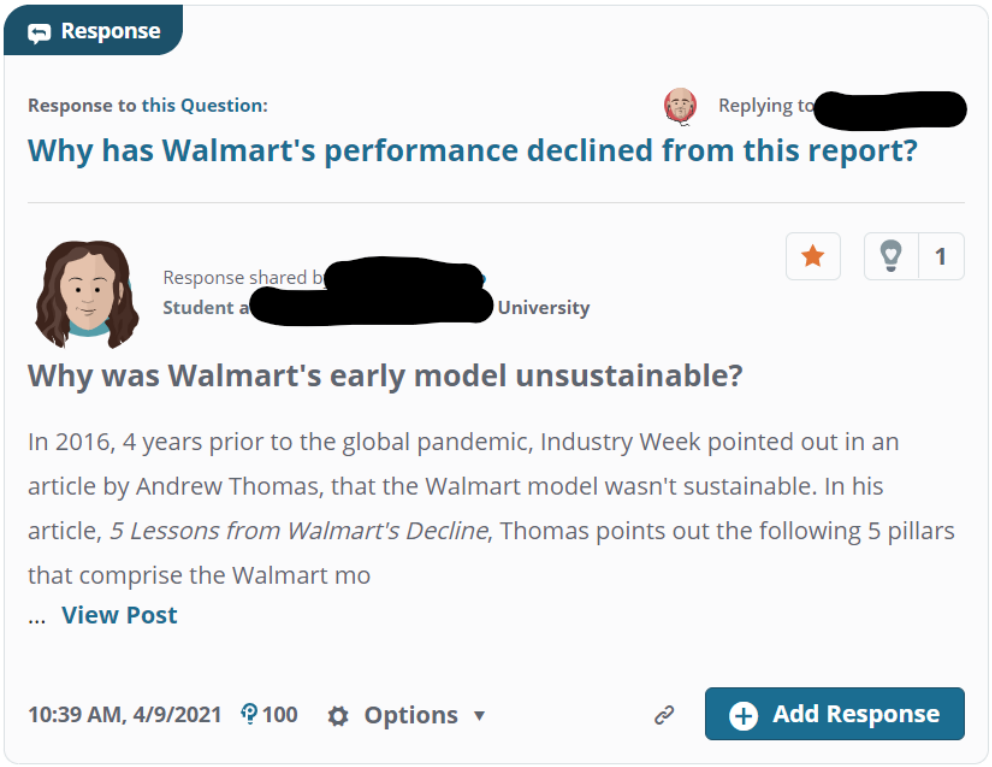


Figure 6 – Student Post on Discussion Board

The Role of Task Value and Online Learning Strategies in an Introductory Computer Programming Course

Pratibha Menon
menon@calu.edu
Department of Computer Science, Information Systems, and Engineering Technology
California University of Pennsylvania
California, PA, 15419, USA

Abstract

The autonomy and flexibility that online learning contents provide students in a traditional face-to-face course require them to pick up newer strategies for regulating their learning process. This study focuses on identifying how students' self-reported traits of self-regulated learning may relate to the task value of the learning contents of an introductory programming course. This study explores the distribution of self-regulated learning and task value components reported by students. A moderately positive correlation is seen between the task value and perceived self-regulated learning traits of students. The findings of this study demonstrate how some of the online learning components and facilitation methods that students value the most could be incorporated into a traditional face-to-face course to promote self-regulated learning skills.
Keywords: Task-Value, Self-Regulated Learning, Computer-Programming, Self-Evaluation, Interest, Help-Seeking

1. INTRODUCTION

Although students today display increasing familiarity with online tools and communication technologies, many are unfamiliar with online learning methods (Gillett-Swan, 2017). The flexibility and autonomy that online learning afford to learners in an online/blended-online environment also necessitate a commitment to effectively completing course-related tasks. To maintain a high responsibility for the learning tasks, students need to perceive a high task value, which is how the course meets the learner's interests and future goals. Task value, which is a motivational construct, increases engagement and promotes course completion and academic success (Jung & Lee, 2018; Vanslambrouck, 2018; Zhang & Liu, 2019). Studies report that students with a high task value employ more profound cognitive and metacognitive strategies (McWhaw & Abrami, 2001). The flexibility and autonomy of online learning also require students to develop critical self-regulated learning (SRL) skills and strategies (Azevedo, 2007; Barnard et al., 2009; Lee & Choi, 2011; Rasheed et al., 2020). This paper explores the distribution of task value and the self-regulated learning skills reported by students who complete an undergraduate computer programming course with significant online learning content.

The concept of task value, which derives from expectancy-value theory, is operationalized by measuring the learners' perspective of the task's interest, usefulness, and importance (Eccles & Wigfield, 2002). In their model, Eccles and Wigfield define task values with respect to the qualities of different tasks and how those qualities influence the individual's desire to do the task. Interest in a task refers to the intrinsic value of enjoyment or inherent motivation for the task. The term usefulness stands for the student's perception that the task will be helpful to meet

some future goals. The term importance stands for the attainment value or the value of doing well on the task.

Self-regulated learning indicates the ability of learners to regulate their motivation, metacognition, cognition, and behavior to meet their learning goals. Self-regulated learning takes place through an active, constructive process. Learners plan and set goals before learning, monitor their progress, and then self-evaluate their performance after learning (Pintrich, 1999; Zimmerman, 2008). Studies have also shown that SRL skills regulate students' cognition and motivational factors such as task-value (Butler & Cartier, 2005; Pintrich, 2004). Prior studies have investigated the role of SRL in computer programming courses for a campus-based delivery format (Ramirez et al., 2018; Kumar et al., 2005; Castellanos et al., 2017).

This study intends to explore the distribution of students' perception of the online learning contents' task value and investigate how task value is associated with perceptions of self-regulated learning. This study takes place in a blended online class of an undergraduate level introductory computer programming course at a public university.

The motivation for this study is to explore student perceptions after having redesigned an introductory programming course to meet the learning needs of students during the COVID-19 pandemic. The redesign is characterized by blending online learning with instructor support, either via in-person or zoom sessions during regular class hours due to the COVID-19 pandemic. During the two semesters under this study, the pandemic situation inhibited students from attending in-person sessions. Redesigning the course became crucial to motivating students and supporting their self-regulated learning skills throughout the uncertain conditions caused by the pandemic. This study will help instructors identify how students value the course and its learning tasks and what kind of support they might require in improving their self-regulated learning skills.

2. THE STUDY

This study investigates the distribution of task value perceived by students who attended an undergraduate-level introductory programming course with online learning contents. While online learning activities afford autonomy and flexibility, they also require students to employ self-regulated learning skills. Hence, this study investigates if there exists an association between students' perceptions of their self-regulated learning skills and the task value of the course. This study attempts to answer the following questions:

- How do students perceive the task value of the introductory programming course and its online learning contents?
- How do students who attend an introductory programming course perceive their self-regulated learning?
- What kind of association exists between the task value and the perceived self-regulated learning traits reported by students in an introductory programming course that contains online learning components?

Context

This study takes place during two semesters of an undergraduate introductory Java-programming course. The course contains a significant online component that includes a series of short video lectures, detailed code demonstrations of programming solutions, self-assessment quizzes, graded quizzes, graded online assignments, structured feedback, and a discussion forum. In addition, students access all the learning materials for the course from the Learning Management System (LMS).

Students regularly complete auto-graded practice quizzes that follow every lecture video. The lecture videos that introduce key concepts are short and do not exceed fifteen minutes. The practice exercise also includes a series of coding assignments that require students to design and implement programs in Java using Eclipse- an integrated development environment (IDE) used throughout the course. The coding assignments are more significant projects for which students obtain feedback from the instructor to improve their solutions. Students communicate to the instructor via the online discussion board, emails, and office hours set up through Zoom. The assignment submission drop box in the LMS affords ways to provide written and video feedback for the submissions.

Apart from the asynchronous online components, the course supported a bi-weekly instructional session during regular class hours during both semesters. During the first semester under the study, due to the COVID-19 pandemic situation, the instructional sessions were conducted using Zoom. It became possible to achieve in-person sessions during the following semester through in-person classes. However, the pandemic situation still made it impossible for a few

students to attend all the in-person sessions due to their need to quarantine. In addition, since all the learning materials and recordings were available online, students could flexibly learn from home and get caught up on the course materials if they could not attend any of the face-to-face sessions.

Each face-to-face session covered a quick recap of the concepts covered in the video lectures, followed by problem-solving sessions that discussed several types of problems typical to the programming topic. In addition, the instructor used the face-to-face sessions to address common errors encountered by students. The assignments for a topic contained multiple questions, and they could be completed by students flexibly and submitted before a hard deadline. Students were allowed to submit their assignments for the instructor's feedback. They were allowed to use the feedback to correct or improve their solution and resubmit the problem before the hard deadline. The video lectures and the associated online quizzes made it possible for students to complete a self-assessment of their knowledge and re-watch the videos if needed to clarify any misconceptions. Students must do prior planning, independently write the programs, obtain feedback, or help if needed, and reiterate the solutions to meet the programming assignments' problem-solving requirements fully. The videos, quizzes, and assignments contained mechanisms that support students' skills to apply self-regulated learning cycles.

The students in this class, who are also the study participants, are regular campus students. The introductory Java Programming course is a required pre-requisite for several higher-level Computer Information Systems courses. However, this is also a general education course that enrolls students who are non-majors.

The LMS has features that allow the instructor to set up deadlines and control access to submission drop boxes, quizzes, and assignments. The calendar system in LMS also provides submission reminders on the course page. In addition, students visit the course pages several times a week to keep track of the tasks and due dates. Finally, the online discussion boards allow students to interact with the rest of the class.

Survey Instruments

Seventeen students from the first semester and fourteen students from the following semester participated in an anonymous, end-of-the-course survey. In the instructor's absence, the survey was administered to all students concurrently during class time using an online survey tool. Table 1 summarizes the three main parts of the study implemented. The end of the course survey includes an SRL section, a task value section, and a ranking of course-related tasks section.

Survey type	Scale	Instru-ment	Measurement
SRL	1 – 5 Likert Scale	OSLQ with six sub-scales	Goal Setting, Environment Restructuring, Task Strategies, Time Management, Help Seeking, Self-Evaluation capabilities of students
Task Value	1 - 7 Likert Scale	MSLQ - task value sub-scale (only)	Interest, Usefulness, Importance of the course.
Ranking Questions	1 - 5 Rank Order	Developed by the author A of set of three ranking questions, each with 5 choices.	Ranking of reasons that makes the course important, interesting, and useful to students

Table 1. Summary of survey instruments

The survey instrument includes an Online Strategies for Learning Questionnaire (OSLQ), a pre-validated 24-item questionnaire that evaluates students' SRL skills in an online learning environment (Barnard et al., 2009). The OSLQ consists of goal setting, environment restructuring, task strategies, time management, help-seeking, and self-evaluation subscales. The subscales of OSLQ map into Zimmerman's SRL model (Zimmerman, 1998), consisting of forethought, performance, and self-reflection phases. The goal setting and environment restructuring phase correspond to the forethought phase of Zimmerman's SRL model. The subscales of task strategies, time management, and help-seeking correspond to the performance phase of the SRL model. As the name suggests, the self-evaluation subscale measures the self-evaluation phase of Zimmerman's SRL model. The OSLQ asks students to rate their responses on a scale of 1 ('not-at-all-like-me") to 5 ("very-much-like-me"). Appendix A shows the subscales and items in the OSLQ used in this study. The internal reliability of

OSLQ, as indicated by Cronbach's alpha value, is between 0.85 and 0.92 for the subscales (Barnard, Paton & Lan, 2008).

In addition to the items from the OSLQ questionnaire, the survey instrument contains six questions that measure the perceived task value of course contents. These questions are part of a Motivated Strategies for Learning Questionnaire (MSLQ) (Pintrich et al., 1993). Appendix B shows the task value subscale of the MSLQ. In addition to the task value scale, the survey also includes a series of ranking questions created by the author to discover how various critical features of the course content shape students' perceived importance, usefulness, and interest. Students rank the answer choices on a scale of 1 (most important/useful/interesting) to 7 (least important/useful/interesting). This survey component helped the instructor pinpoint the course design's important aspects of the perceived task value.

The task value components of the MSLQ survey can only be used to collect a general perception of students about the importance, interest, and usefulness of a course. To learn how students value various course features, the author of this paper created a set of ranking questions to know what some of the course-related features were useful, important, and interesting to students. Appendix C shows three ranking questions, each with five choices. Students respond to these questions by ordering the choices on a scale of 1(high value) to 5 (low value). The electronic survey tool ensured that no choice in a set had the same rank.

3. RESULTS

Before analyzing the survey results, the author of this paper inspected the data to find and remove any incomplete attempts. The online survey tool had features that could control the ranking questions' irregularities, such as using the same rank value for two different answer choices. Altogether, 31 students had fully completed the survey.

Measuring SRL

Cronbach's alpha value provides the reliability of the 24-item survey instrument on online learning strategies. In the current study involving OSLQ and consisting of 31 (complete) responses, the Cronbach's coefficient value of the subscales is sufficient with an alpha value between 0.88- 0.66 for the sub-scales, as shown in Table 2. A high alpha value indicates a high internal consistency of the items in each subscale and the overall SRL scale. Shapiro-Wilk test performed on data from each subscale revealed that only three out of the six subscales indicated a normal distribution. Therefore, median values are used as the descriptor for each subscale.

SRL Sub-Scale	Cron bach-Alpha	Median Score
Goal Setting (GS)	0.72	3.9
Environment Restructuring (ER)	0.88	3.4
Task Strategy (TS)	0.78	3.6
Time Management (TM)	0.86	3.1
Help Seeking (HS)	0.66	3.4
Self Evaluation (SE)	0.68	3.5
SRL total	0.85	3.5

Table 2. Aggregate scores of the online SRL subscales

Appendix A shows the items for every subscale of the OSLQ questionnaire. As described in the previous section, students' responses to the SRL questionnaire are scored on a scale of 1 (”not-at-all-like-me") to 5 ("very-much-like-me"). Table 2 indicates the median value of the weighted average scores of each of the subscales of the SRL questionnaire. All the components appear symmetrically distributed from the histogram shown in Figure 1, except for environment restructuring skewed to the left. The scores used to construct the bar chart uses the median value of the survey responses for each of the components of the SRL survey.

Task Values

The survey measures the task value of the course on a scale of 1 (that stands for most important/useful/interesting) to 7 (least important/useful/interesting), using the task value subscale of the MSLQ questionnaire. Appendix B shows the items and the task value subscale. Shapiro-Wilk test was used to check the normal distribution of the responses for the task value subscales that were found to be non-parametric.

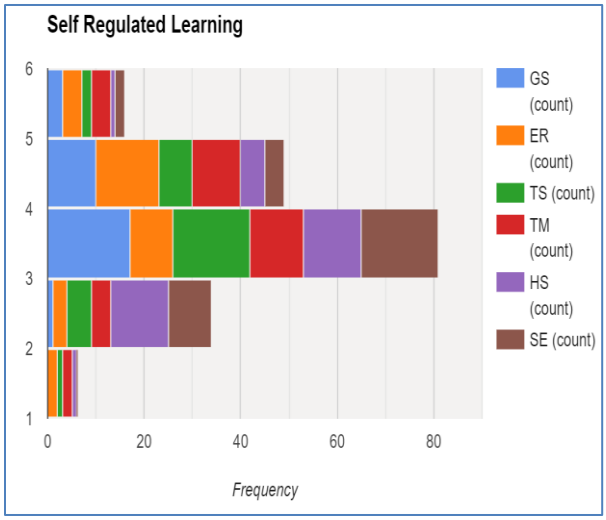


Figure 1. SRL Components Histogram

Table 3 shows the median scores for importance, utility, and interest. The median overall task value score is 6.5, which indicates a high value. Through the task value survey responses, students express how much they felt the course to be important, useful, and interesting by answering a series of ranking questions on a scale of 1(most important/useful/interesting) – 7 (least important/useful/interesting).

Task Value Sub-Scale	Cron bach-Alpha	Median Score
Importance	0.91	7
Utility	0.76	6.7
Interest	0.85	6.5
Task Value Total	0.91	6.5

Table 3. Aggregate Task Value Scores

The components of the task value scores are depicted in a bar chart in Figure 2. The bar chart has seven bins corresponding to the task value scale of 1 to 7. The scores are used to construct the histogram using the median value of the task value survey responses from each student for each task value component. As a result, the bar chart displays a skewed distribution of the three task value components.

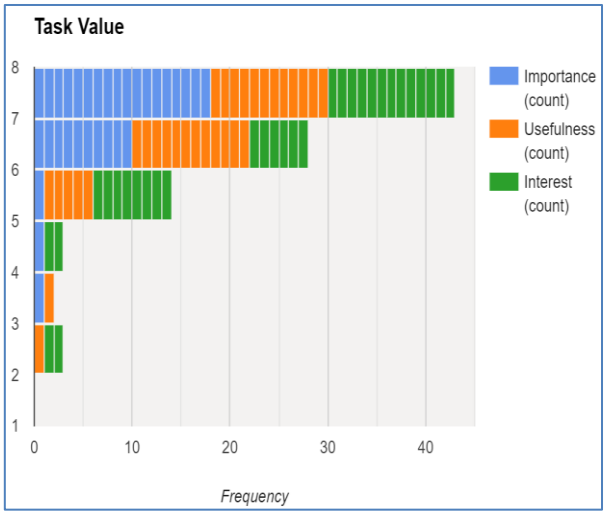


Figure 2. Task Value Components Histogram

Correlation between Task Value and SRL

Data collected from the survey reveals a moderately positive correlation between some SRL and the task value components. Table 4 shows the correlation matrix with the values of Spearman's rank correlation coefficients, r, and the significance value p. For correlating the aggregate SRL and task value scores, values of 'SRL Total Score' and 'Task Value Total Score' are calculated for every student by using the weighted average of all SRL and Task Value responses, respectively.

The correlation matrix in Table 4 indicates a medium positive correlation between task value components and the SRL components of task strategy, help-seeking, and self-evaluation.

	Task Value Total Score	Task Value-Interest	Task Value-Useful	Task Value-Importance
SRL Total Score	**r=0.41 p=0.02**	***r=0.39 p =0.02***	**r=0.51 p =0.03**	**r=0.39 p = 0.02**
Goal Setting	r=0.33 p=0.06	*r =0.28 p =0.1*	**r= 0.44 p =0.01**	**r=0.37 p=0.03**
Env. Restruct	*r=0.13 p= 0.4*	*r=0.1 p =0.5*	*r=0.16 p=0.3*	r=0.17 p=0.35
Task Strategy	**r=0.46 p =0.008**	**r= 0.38 p =0.03**	**r= 0.46 p= 0.009**	**r=0.52 p=0.002**
Time Management	*r=0.29 p=0.1*	*r=0.16 p=0.3*	**r=0.38 p=0.03**	**r=0.48, p=0.005**
Help Seeking	**r=0.49 p=0.004**	**r=0.4, p=0.02**	**r=0.45 p=0.009**	**r=0.48, p=0.05**
Self Evaluation	**r=0.52 p=0.002**	**r= 0.49 p=0.005**	**r=0.5 p <0.001**	**r=0.58 p < 0.001**

Table 4. SRL – Task Value Correlations

All the grey-colored cells in the correlation matrix in Table 4 show the significant correlations whose p-values are less than 0.05. There was no significant correlation between task value and the SRL components of environment restructuring and time management. Perception of goal setting and time management is moderately correlated with the usefulness and interest in the course. A medium positive correlation between the SRL and task value indicates that students who perceive high task value may not always report high SRL. Conversely, students who have low scores of SRL may still perceive a high task value. students

Cluster analysis is used to visualize how students could be separated into distinct clusters based on the association between their task value and SRL scores. A K-Means cluster analysis using the standardized aggregate scores of SRL and task values shows the presence of three discernable clusters, as shown in Figure 3. The three-cluster model, created using the K-Means cluster analysis method, is shown in Figure 3. This three-cluster model explains 66.08% of the variance, a within-group error-sum-of-squares (SSE) of 20.35, and a between-group SSE of 39.7. Among the three clusters, Cluster 0 is the largest one that has 13 students who report moderately high task value and medium to low SRL skills. Cluster 1 is the next large cluster with 9 students, indicating high perceptions of task value and higher SRL score. Finally, Cluster 2 has 9 students who report low perceptions of task value and lower scores of SRL. A sizeable number of students under Cluster 0 supports the lack of a strong correlation between task value and SRL scores, as indicated in Table 6.

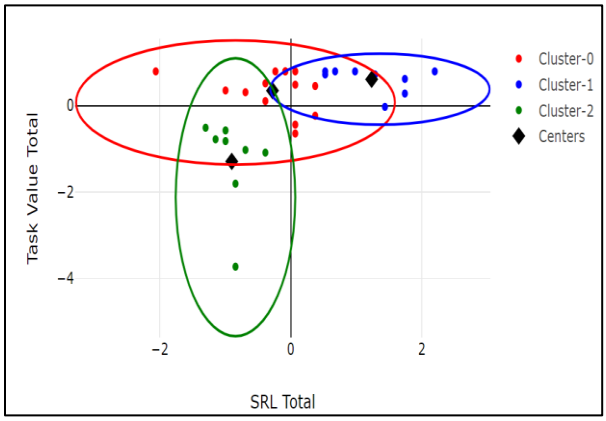


Figure 3. Cluster Analysis SRL and Task Value

Ranking the Factors that Make the Course Valuable

Tables 5, 6, and 7 show the ranking of various features of the course to learn the order in which these features contribute to the perceptions of task value. The ranking questions require the survey respondents to order the choices such that a response of 1 stands for the most ranked-choice and a response of 5 stands for the least ranked one. However, the responses to the ranking questions, as shown in Tables 5, 6, and 7, elicit the course design features that students find most, or least important, interesting, and useful. For each ranking question, students were asked to order the choices on a scale of 1 to 5. The median score values in Tables 5, 6, and 7 show the ranking scores' median values based on student responses. The column called 'Final Rank' is by ordering the calculated median scores of each choice.

Please rank each of the choices on a scale of 1 - 5 on why this course is <u>important</u> to you. On the 1-5 scale, 1 stand for (most important) and 5 stands for (least important).		
	Median Score of Ranks	Final Rank
It is a required course	4	3
It allows me to learn how to write computer programs	2	1
It improves the chances of getting a job	2	1
It allows me to understand how a programming solution is constructed using logical elements	3	2
It makes me successful in future classes	4	3

Table 5: Ranking question on why the course is important.

When asked why the course is important to students, the median scores of the item with the highest and the lowest ranks varied by 2 points on the Likert scale. The most important feature of the course was that it helps students learn how to write computer programs and helps them get jobs. Even though the introductory programming course is a pre-requisite for many other upper-level courses in the CIS program, it is also a general education course that enrolls non-majors.

Non-majors may not require introductory programming for their future studies. However, many non-majors attend this class to view programming as an essential job skill. Therefore, future course design iteration could leverage this motivational aspect and show how professionals in different careers use computer programming in day-to-day jobs.

Table 6 shows the ranking for items that ask students about the usefulness of the course. Again, the variation between the minimum and maximum scores was 2 points on the Likert scale. Furthermore, it seems as if students focus on the usefulness of the course based on problem-solving and programming skills they have picked up.

Please rank each of the choices on a scale of 1 - 5 on why this course is <u>useful</u> to you. On the 1-5 scale, 1 stands for (most useful), and 5 stands for (least useful).		
	Median Score of Ranks	Final Rank
I learnt how to write and test Java programs using Eclipse (IDE)	3	2
I improved my problem-solving skills by writing programs	2	1
I can apply what was learnt in this class in future courses	3	2
I have developed skills to evaluate the correctness of my programs	4	3
I have gained skills on how to translate the requirements in the problem to develop programming solutions	3	2

Table 6. Ranking questions on why the course is useful.

Student perceptions on the utility of the course could have been influenced by the fact that heavy emphasis was placed on problem-solving methods. For example, the code-demonstrations video breaks down a typical programming process into identifying the givens in the problems, looking for similar problems from the worked-out examples, identifying the required outputs, and configuring the programming constructs. In addition, the class discussions usually centered around problem-solving methods. The assignment feedback also contained pointers on how students could improve their self-evaluation skills and programming skills. Therefore, a sizable portion of the online videos and class discussions afforded instruction on developing problem-solving and programming skills, supporting student SRL.

Students would spend most of their learning time writing and iterating their programming solution using Eclipse IDE equipped with 'intelli-sense' to guide students through their code. However, most novice students struggle with self-evaluating their code's correctness and require help and feedback from the instructor. That could have been why students gave lower rank to the usefulness of the course in helping them learn how to evaluate the correctness of their program by themselves.

Table 7 shows the ranking for items that ask students why the course is interesting. In Table 5, ranks 1 and 2 pertain to the design of the course. The hands-on, active learning and facilitation methods involved learning components that made the class enjoyable.

Please rank each of the choices on a scale of 1 - 5 on why this course is <u>interesting</u> to you. On the 1-5 scale, 1 stands for (most interesting) and 5 stands for (least interesting).		
	Median Score of Ranks	Final Rank
I liked the way the course was facilitated (e.g.: code demos, videos, self-assessment quizzes, and multiple submissions for assignment)	2	1
I found it interesting to learn programming by completing the module assignment questions.	3	3
I enjoyed the hands-on learning process	2.5	2
I enjoy learning how the basic programming constructs such as the decision structures, loops, and objects are used in real-world applications	3	3
I enjoyed working with Eclipse	4	4

Table 7. Ranking questions on why the course is interesting

The lower final ranks pertain to the steps required to complete a coding assignment, including using

Eclipse IDE to complete programs. Future course design could investigate more user-friendly IDEs suitable for beginner learners.

4. DISCUSSION

This study investigates students' perceptions of the task value of an introductory programming course. This study also presents the distribution of various SRL skills reported by the students. While the task value items of the survey pertain to the course-specific features, SRL questions pertain to a student's learning skill that may not pertain specifically to the programming course. Their past and current learning experiences may influence a student's self-reported SRL skills. The distribution of the SRL components shows a mostly symmetrical distribution, but the reported task value components are all are skewed onto the higher task values in the 1 (high task value)-7 (low task value) scale.

The study results cannot ascertain if perceptions of task value would have contributed to the reported SRL scores or vice versa. However, the study discovers a moderately positive correlation between various components of task value and the SRL. In addition, the presence of student clusters has implications on how the course design could have catered to the learning needs of students from each group.

The ranking questions for the three components of the task value, as depicted in Tables 5, 6, and 7, show that the course design and facilitation of an active learning process play an essential role in generating interest in learning. The course facilitation methods and active learning that students found enjoyable require them to apply higher levels of SRL while working on the various online components. For example, completing the assignment is a multi-week process that requires cycles of planning, learning the materials, writing the programming solutions, obtaining feedback, and fixing the solution. Every assignment package also requires students to plan out their learning process by deciding when and how to complete the self-assessment quizzes, watch or re-watch the videos and get help. The instant feedback from the quizzes allows students to revisit the concepts presented in the videos. The debugging methods demonstrated through the videos show how to check the code before submitting the program. Multiple submissions of an assignment allow students to obtain feedback and guidance from the instructor and a chance to improve the final version. The feedback and the grades from the formative assessments could help students strategize their learning for the coming weeks. Therefore, a significant part of instruction and feedback provided ways to involve students in self-regulated learning.

The clusters in Figure 3 reveal an association between low-task value and lower perceived SRL. Therefore, students in this cluster could be motivated by making the course more exciting and engaging that they feel motivated to develop their SRL skills by engaging in the course. On the other hand, students from the central cluster that reported medium to high task value and low SRL skills could benefit from encouraging feedback that explains how completing the task has improved or could improve their SRL skills. Therefore, another approach will be to adapt the learning contents that will enhance task value and perceptions of SRL among students depending on their current perceptions.

This study took place in two small classes, and only 31 students participated in the survey. A small sample size limits the generalizability of the findings. Despite its limitations, this study has developed a method to study the relationship between online learning strategies and task value using well-known and validated survey instruments that can apply across multiple contexts and online course delivery methods. Future studies could compare the self-reported SRL skills with the observed SRL by collecting course-related data logs. What students reports, however biased they might be, are nevertheless significant as it could determine the perceptions of self-efficacy, which is a crucial motivating factor for students to continue to engage in future programming courses.

Both self-efficacy and SRL are perceptions that could be influenced by factors unrelated to the course alone. However, task value is an important motivating factor that shows the situational interest of the student. Knowing what students value in a course can help instructors figure out ways to improve student SRL. Such a student-centric approach to course design will help students take advantage of the flexibility and autonomy provided by online learning components to achieve academic success.

5. CONCLUSIONS

This study elicits student perceptions of their SRL skills and task value for an undergraduate introductory programming course with significant online learning content. Course-specific factors that influence the perceived task value regarding importance, utility, and interest could guide the future redesign of the online learning experience.

Student responses rank a well-facilitated, hands-on learning experience essential for making the course interesting. Students found the course to be necessary for their long-term career prospects. Additionally, due to the focus on problem-solving methods, students felt the course helped improve their problem-solving skills.

Correlation studies and cluster analysis indicated an association between task value and perceived SRL reported by students. The clusters did suggest that students who express high task value also tend to perceive higher SRL skills. It is also observed that students who perceive a low SRL for themselves tend to report a lower task value. Most of the students do not fall under either extreme, reporting a medium to low SRL with medium to high task values. Students in the low and medium score clusters that form a sizable portion of the class could benefit from instruction, activities, and feedback to improve their actual SRL and how they perceive their SRL skills. Future studies could use student data to infer self-regulated learning behavior and compare the findings with the SRL that students report through surveys.

Online learning continues to challenge ways in which students employ self-regulated learning. Students' lack of face-to-face interaction and increased autonomy and responsibility requires them to develop newer SRL strategies in an online learning environment. Therefore, developing strategies to improve the course content's task value could motivate students to build better online learning strategies.

6. ACKNOWLEDGEMENTS

The author would like to thank and acknowledge the PASSHE-FPDC grant for funding the instructor during summer 2019. The grant funding helped the author acquire the required professional development on the Self-Regulated Learning, and Learning Design Study applied in this study.

7. REFERENCES

Azevedo, R. (2007). Understanding the complex nature of self-regulatory processes in learning with computer-based learning environments: An introduction. Metacognition and Learning, 2(2-3), 57–65.

Barnard, L., Lan, W. Y., To, Y. M., Paton, V. O., & Lai, S. L. (2009). Measuring self-regulation in online and blended learning environments. *The Internet and Higher Education*, *12*(1), 1–6.

Barnard, L., Paton, V. O., & Lan, W. Y. (2008). Online self-regulatory learning behaviors as a mediator in the relationship between online course perceptions with achievement. The International Review of Research in Open and Distance Learning, 9(2).

Butler, D. L., & Cartier, S. C. (2005, April). Multiple complementary methods for understanding self-regulated learning as situated in context. Paper presented at the 2005 Annual Meeting of the American Educational Research Association, Montréal, Canada.

Castellanos, F. H. Restrepo-Calle, F., González. A, & Echeverry, J. R. (2017). "Understanding the relationships between self-regulated learning and students' source code in a computer programming course," 2017 IEEE Frontiers in Education Conference (FIE), Indianapolis, IN, 2017, pp. 1-9.

Eccles, J. S., & Wigfield, A. (2002). Motivational Beliefs, Values, and Goals. *Annual Review of Psychology*, *53*(1), 109–132.

Gillett-Swan, J.K. (2017). The challenges of online learning: Supporting and engaging the isolated learner.

Jung, Y., & Lee, J. (2018, July). Learning Engagement and Persistence in Massive Open Online Courses (MOOCS). *Computers & Education*,9–22.

Kumar, V., Winne, P., Hadwin, A., Nesbit, J. et al., (2005) "Effects of self-regulated learning in programming," Advanced Learning Technologies. ICALT 2005. Fifth IEEE International Conference on, pp. 383-387

Lee, Y., & Choi, J. (2010). A review of online course dropout research: implications for practice and future research. Educational Technology Research and Development, 59(5), 593–618.

McWhaw, K., & Abrami, P. C. (2001). Student Goal Orientation and Interest: Effects on Students' Use of Self-Regulated Learning Strategies. *Contemporary Educational Psychology*, *26*(3), 311–329. https://doi.org/10.1006/ceps.2000.1054

Pintrich, P. R., Smith, D. A. F., Garcia, T., & Mckeachie, W. J. (1993). Reliability and Predictive Validity of the Motivated Strategies for Learning Questionnaire (Mslq).

Educational and Psychological Measurement, *53*(3),801–813.

Pintrich, P. R. (1999). The role of motivation in promoting and sustaining self-regulated learning. *International Journal of Educational Research*, *31*(6), 459–470.

Pintrich, P. R. (2004). A Conceptual Framework for Assessing Motivation and Self-Regulated Learning in College Students. *Educational Psychology Review*, *16*(4), 385–407.

Ramirez E. J. J., Rosales-Castro, L. F., Restrepo-Calle, F., & Gonzalez, F. A. (2018). Self-Regulated Learning in a Computer Programming Course. IEEE Revista Iberoamericana de Tecnologias Del Aprendizaje, 13(2), 75–83.

Rasheed, R. A., Kamsin, A., & Abdullah, N. A. (2020). Challenges in the online component of blended learning: A systematic review. *Computers & Education*, *144*, 103701. https://doi.org/10.1016/j.compedu.2019.103701

Vanslambrouck, S., Zhu, C., Lombaerts, K., Philipsen, B., & Tondeur, J. (2018). Students' motivation and subjective task value of participating in online and blended learning environments. *The Internet and Higher Education*, *36*, 33–40.

Zhang, S., & Liu, Q. (2019). Investigating the relationships among teachers' motivational beliefs, motivational regulation, and their learning engagement in online professional learning communities. Computers & Education, 134, 145–155.

Zimmerman, B. J. (1998). Developing self-fulfilling cycles of academic regulation: An analysis of exemplary instructional models. In D. H. Schunk & B. J. Zimmerman (Eds.), Self-regulated learning: From teaching to self-reflective practice (pp. 1–19). Guilford Publications.

Zimmerman, B. J. (2008). Investigating Self-Regulation and Motivation: Historical Background, Methodological Developments, and Future Prospects. *American Educational Research Journal*, *45*(1), 166–183.

Appendix A

This set of questions requires you to self-assess your competency in various aspects of self-regulated learning. Please note that the use of the term "online" in some of the questions are synonymous with "blended-online". Please provide responses to the following questions on a 1 to 5 scale where 1 stands for " Not at all like me" and 5 stands for "Very much like me"					
Goal Setting					
	5- Very much like me	4	3	2	1- Not at all like me
I set standards for my assignments in *online/blended* courses.					
I set short-term (daily or weekly) goals as well as long-term goals (monthly or for the semester).					
I keep a high standard for my learning in my courses					
I set goals to help me manage studying time for my *online/blended-online* courses.					
I don't compromise the quality of my work because it is online					
Environment Restructuring					
I choose the location where I study to avoid too much distraction.					
I find a comfortable place to study.					
I know where I can study most efficiently for *online/blended-online* courses.					
I choose a time with few distractions for studying for my *online/blended online* courses.					
Task Strategies					
I try to take more thorough notes for my *online/blended online* courses because notes are even more important for learning online than in a regular classroom.					
I read aloud instructional materials posted online to fight against distractions.					
I prepare my questions before joining in discussions.					
I work extra problems in my *online/blended-online* courses in addition to the assigned ones to master the course content.					
Time Management					
I allocate extra studying time for my *online/blended-online* courses because I know it is time-demanding					
I try to schedule the same time everyday or every week to study for my *online/blended-online* courses, and I observe the schedule.					
Although we don't have to attend daily (in-person) classes,I still try to distribute my studying time evenly across days.					
Help Seeking					
I find someone who is knowledgeable in course content so that I can consult with him or her when I need help.					
I share my problems with my classmates online so we know what we are struggling with and how to solve our problems.					
If needed, I try to meet my classmates face-to-face.					
I am persistent in getting help from the instructor through e-mail *and regular class-sessions.*					
Self Evalutaion					
I summarize my learning in *online/blended-online* courses to examine my understanding of what I have learned.					
I ask myself a lot of questions about the course material when studying for an *online/blended-online* course.					
I communicate with my classmates to find out how I am doing in my *online/blended-online* classes.					
I communicate with my classmates to find out what I am learning that is different from what they are learning.					

OSLQ questionnaire and responses were used to rate various SRL skills. The term "online" in the original survey was modified to "online/blended-online" to be consistent with the names of the learning modality that is familiar to students. All the changes made to the original survey instruments are shown as italicized words in the items listed above.

APPENDIX B

Please answer this question on a scale of 1-7 where 1 stands for "Strongly disagree" and 7 stands for "Strongly agree".							
	7- Strongly Agree	6	5	4	3	2	1- Strongly Disagree
Importance							
It is important for me to learn the course material in this class.							
Understanding the subject matter of this course is very important to me.							
Utility							
I think I will be able to use what I learn in this course in other courses.							
I think the course material in this class is useful for me to learn.							
Interest							
I am very interested in the content area of this course.							
I like the subject matter of this course.							

Task Value subscale of MSLQ questionnaire and survey results

APPENDIX C

Please rank each of the choices on a scale of 1 - 5 on why this course is <u>important</u> to you. On the 1-5 scale, 1 stand for (most important) and 5 stands for (least important).	
	Rank Order
It is a required course	
It allows me to learn how to write computer programs	
It improves the chances of getting a job	
It allows me to understand how a programming solution is constructed using logical elements	
It makes me successful in future classes	

Please rank each of the choices on a scale of 1 - 5 on why this course is <u>useful</u> to you. On the 1-5 scale, 1 stands for (most useful), and 5 stands for (least useful).	
	Rank Order
I learnt how to write and test Java programs using Eclipse (IDE)	
I improved my problem-solving skills by writing programs	
I can apply what was learnt in this class in future courses	
I have developed skills to evaluate the correctness of my programs	
I have gained skills on how to translate the requirements in the problem to develop programming solutions	

Please rank each of the choices on a scale of 1 - 5 on why this course is <u>interesting</u> to you. On the 1-5 scale, 1 stands for (most interesting) and 5 stands for (least interesting).	
	Rank Order
I liked the way the course was facilitated (e.g.: code demos, videos, self-assessment quizzes, and multiple submissions for assignment)	
I found it interesting to learn programming by completing the module assignment questions.	
I enjoyed the hands-on learning process	
I enjoy learning how the basic programming constructs such as the decision structures, loops, and objects are used in real-world applications	
I enjoyed working with Eclipse	

Ranking Question for why the course (and its tasks) were interesting, important, and useful

Comparison of Information Technology Professionals' Perception of Job Satisfaction and Inclusion by Gender: Insights for Recruitment and Retention of Female Students

Leah Schultz
lschult@tarleton.edu

Mark Adams
madams@tarleton.edu

Marketing and Computer Information Systems
Tarleton State University
Stephenville, Texas 76401, USA

Abstract

This paper examines responses to questions about job satisfaction and inclusion from professionals in the information technology field. Responses from over 10,000 professionals were analyzed to determine if there were differences in response to these questions based on gender of respondent. This information, along with previous research on inclusion of women in higher education and industry, are discussed to determine similarities with previous research. Results from the study are used to suggest ways that educators can use the responses to improve recruitment and retention of females in technology majors and minors.

Keywords: Women in technology, recruitment, retention, job satisfaction, inclusion.

1. INTRODUCTION

Jobs in information technology (IT) continue to grow in the United States and new jobs are predicted to increase by 11%, adding approximately half a million new jobs by 2029 (Bureau of Labor Statistics, 2021). In 2015, only 25% of computing jobs in the United States were occupied by women which declined sharply since the 1990's but has been holding steady in recent years (Ashcraft, McLain, & Eger, 2016). In rapidly growing fields within IT such as information security analysis, jobs are projected to grow at a staggering rate of 31% over the next 10 years with only 16.8% of jobs in the area held by women (Women's Bureau, n.d.). Retention of female employees in IT related fields is dismal with more than 50% of women leaving the field (Hewlett, Luca, Servon, Serbin, Shiller, Sosnovich & Sumberg, 2008). There are many factors leading to this underrepresentation, starting with fewer women choosing information technology careers in college and large numbers leaving the field for various reasons.

In order to increase participation by women in the technology field, it is important to understand why women choose or do not choose to major in information technology related fields in college and why they choose to leave their careers in large numbers. Understanding these factors may help educators recruit and prepare female students for IT careers and aid employers in retaining employees.

The purpose of this study is to look at various aspects of job satisfaction and inclusion for professionals in IT careers and determine if there are differences in these beliefs between genders. This study will compare responses to male counterparts whereas many previous studies on women in the workplace are studied as a standalone population. Comparing the two populations will provide insight into whether certain factors are considered differently across gender or if there are factors of job dissatisfaction that are shared by both males and females.

2. BACKGROUND INFORMATION

Fewer Women Entering the Field

In 2019, more 18-24 year old females were enrolled in colleges and universities than males with 57% compared to 43%, respectively (National Center for Education Statistics, 2021a). This trend has held steady since 2010 and graduation rates show similar differences with 66% of female students earning a baccalaureate degree within 6 years of enrollment compared with 60% of male students (National Center for Education Statistics, 2021c). However, in IT related fields these numbers vary greatly. At the associate's degree level in computer and information sciences, 80% of the degrees were awarded to male students. In 2018 and 2019, 63,703 bachelor's degrees in computer and information science were awarded to male students and 15,894 awarded to female students during the same time period. Even though female populations in four-year schools continue to exceed those of males, degrees awarded in computer related fields are significantly skewed towards males (National Center for Education Statistics, 2021b).

Reasons why females are underrepresented in baccalaureate programs have been studied by many researchers over the years with hopes of mitigating the problem. Serapiglia & Lenox (2010) discussed a complex decision-making process for women pursuing computer information systems degrees. They found that male role models, positive introduction to computers and technology, the opportunity for earnings, and natural talent at problem solving were some factors that lead women to choose CIS as a major. Lack of female role models in IT, media perceptions, parental guidance as well as previous programming experience also potentially play a role in selection of computer related majors (Jung, Clark, Patterson & Pence, 2017). In countries where females are more represented in technology fields like Malaysia, lack of female role models is not a problem as it is for students in the United States and Europe (Othman & Latih, 2006).

Socialization and early education have also been studied as possible reasons women do not pursue careers in STEM fields and computer science, in particular. An increasing lack of confidence and fear of failure in technical fields as female students progress through their education has been cited by many researchers as a deterrent for choosing computer related fields as a career (Jung, et al., 2017; McGee, 2018; Serapiglia & Lenox, 2010). Lack of female role models and stereotypes of "geeks only" mindsets have also been cited as reasons female students do not enroll in technology related degree programs (Serapiglia & Lenox, 2010). Preconceptions deterring women from studying computer science indicate that many believe computer science is a male dominated field and that women who work or study computers have low self-worth, are unattractive, and are different or atypical (Berg, Sharpe, & Aitkin, 2018).

Fewer Women Remaining or Advancing in Field

Once women decide to pursue educational opportunities in computer related fields, they face many barriers in the workplace that lead to problems with retention. In 2008, technology fields had the highest rate of abandonment of the STEM fields with over 50% of women leaving the field and at a higher rate than their male counterparts (DuBow, & Gonzalez, 2020). Research has shown that five areas of concern are responsible for many women leaving the field. Bias, isolation, supervisory issues, promotion opportunity and process, and other external life issues are the often cited reasons for women leaving technology (Hewlett, et.al, 2008).

While many women remain in technology careers, those that leave discuss multiple factors influencing their decision. Mentioned in their report on brain drain in STEM fields, Hewlett and colleagues (2008) found a significant number of women who quit in their thirties due to lack of support at their job and challenges with work/life balance. In addition to these, women also mentioned male-dominant attitudes, isolation, undefined career paths, and reward systems based on risk-taking as barriers confronted in the technology workplace. In focus groups investigating workplace barriers and voluntary turnover, Allen, et al., (2006) found that women left their jobs due to problems with flexibility in scheduling, family issues, stress, and workplace policy issues.

Once in field, many women report feelings of career "stall", where they feel they are no longer getting ahead at work. In 2014, approximately 33% of Asian, Hispanic, and Caucasian women reported feelings of career "stall" and a staggering 48% of African American women felt that their careers had stalled (Hewlett, Sherbin, Dieudonné, Fargnoli, & Fredman, 2014).

Advancement opportunities for women in technology fields have also lagged behind opportunities for men. In 2016, only 20% of the Chief Information Officer positions in Fortune 100 companies were held by women (Shein, 2018). While some attribute this to the relative numbers of women in the field, other researchers have proposed other barriers for women. In her article on the influence of gender and race of advancement in IT, McGee (2018) reviewed a large body of literature that found women's views towards IT culture were not favorable. Common themes throughout the research presented, described the field as "predominately male and white, anti-social, individualistic, competitive, hostile toward women, misogynistic, and sexist." Roldan, Soe, and Yakura (2004) believe that this masculine environment is not conducive to career advancement for women and results in high turnover, in general, for female employees.

Although women receive similar job performance ratings in IT, evidence suggests that the effect of job performance ratings on advancement is skewed in favor of men (Igbaria & Baroudi, 1995). Women also cite lack of respect, ageism, stress, and scheduling flexibility as barriers to advancement in their careers (Allen, et al., 2006). Other researchers have investigated structural barriers around formal and informal networking as well as lack of mentors as another impediment to women advancing in technology careers (Armstrong & Reimenschneider, 2014; Roldan, Soe, & Yakura, 2004). Many of the interventions by major tech companies wishing to increase retention of diverse work forces center on support structures and networks (Barker, Mancha, & Ashcroft., 2014). The importance and improvement of professional networking for women in IT has been studied in depth and has been shown to be an important part in retention and advancement of women in IT (Bapna & Funk, 2021; Ahuja, 2002; Kleinbaum, Stuart, & Tushman, 2013).

Importance of Gender Diversity in Workforce

There are social as well as economic reasons that businesses and society at large should be interested in increasing diversity in the IT workplace. From a social aspect, women should be encouraged to find employment in a field that offers significant opportunity in terms of salary as well as impact of work. In 2020, the average annual salary of $91,250 for computer related careers more than doubled the national average salary of $41,950 (Bureau of Labor Statistics, 2021). Salaries and job security are still attractive incentives for women in the workforce, even though the pay gap in technology related fields is wider than other sectors (Chamberlain, 2016).

In their research review of the importance of diversity in the tech workplace, Barker, Mancha & Ashcraft (2014) found that companies perform better when women hold leadership positions. They found superior team dynamics and increased productivity. Research from the report indicates that diverse teams are more likely to stay on schedule and on budget. Collaboration is improved when teams represent both genders (Bear & Wooley, 2011).

Because innovation is critical in many areas of IT, impact of diverse working groups should be of interest to businesses. When groups are diverse, there is an increased level of experimentation and efficiency (Lehman Brothers Centre for Women in Business, 2007).

From an economic standpoint, companies should consider diversity in terms of coming, costly labor shortages. With the expected demand for computer related talent discussed in the introduction, industry should be grooming and recruiting talented labor from all walks of life and should work at avoiding the costly brain drain on their workforce as women continue to leave the field and retention efforts flounder.

3. METHODOLOGY

Anonymized data were used from an online developer survey conducted by Stack Overflow under their Open Database License program. Stack Overflow is a website dedicated to question and answer communities covering a wide range of information technology topics. The survey was completed by over 65,000 participants worldwide in February 2019. Data that were submitted where participants spent less than three minutes on the survey were omitted. Participants were recruited by Stack Overflow through its various online sites and the survey methodology indicates that there is a potential bias of participation from users who are highly engaged in Stack Overflow online properties (Stack Overflow, 2020).

For the purposes of this study, a subset of responses from participants were filtered to include only respondents in the United States and those that indicated a binary gender (male or female). Participants that indicated non-binary or selected more than one gender were excluded but warrant further study in future analysis. Data were further filtered to only include responses from currently employed respondents. Both part time and full-time respondents were included and student and hobbyist respondents were excluded. The final number of responses collected totaled 10,148 but all respondents did not respond to all of the selected questions.

Because the survey covers a wide range of topics, questions that related to work environment, participation, and job satisfaction were selected for the purposes of this study. Likert questions were combined into one construct for analysis around the idea of job satisfaction and participation. Additional questions that required participants to select one or more items from a list were analyzed separately. Questions used for analysis are presented in the appendix. In a few instances, such as computer major and education level, responses were grouped to create a subset that was of interest to the study.

Data were imported into SPSS for analysis of selected criteria related to job satisfaction, participation, and inclusion.

4. RESULTS

Education and Computer Related Majors

Women responding to the survey were more likely to have a college degree than their male counterparts with 86% of women reporting degree completion compared to 79% of male respondents. However, when looking at field of study in degrees, 67% of male respondents had an information technology related degree compared to female respondents who reported in field degrees at 51%.

When asked about the importance of formal education, such as a degree in computer science, there was no statistical difference in how men and women responded. Most of the responses for the question indicated that both genders indicated it to be somewhat important with very few respondents believing that it was either critically or not important at all.

Job Satisfaction and Participation

A series of Likert scale questions were analyzed to determine differences in attitudes towards job satisfaction and participation. Questions used a range from 1 to 5 with five representing a more positive response to the question. Results indicate that women and men do have statistically different attitudes towards their current job as well as their roles and participation in professional activities, $t(1187)=-9.38$, $p<.001$.

Reasons to Initiate a Job Search

When asked to select from a list of issues that might cause respondents to search for a new job, there were many shared reasons and some significant differences between men and women. Both were as likely to job hunt out of curiosity, a desire to share accomplishments, leadership issues, or for no reason at all. Men were more likely to select compensation issues, $t(1038)=-4.05$, $p<.001$, and desire to work with new technologies, $t(1060)=-4.40$, $p<.001$, as reasons to consider switching jobs.

Women were more likely to job hunt after having a bad day, $t(1033)=4.20$, $p<.001$, relocation $t(1039)=3.09$, $p=.002$, trouble with manager, $t(1024)=4.93$, $p<.001$, trouble with teammates, $t(991)=6.22$, $p<.001$, and issues with work/life balance, $t(1060)=2.04$, $p=.041$. Women were also more likely to select interest in growth opportunities as a reason to initiate a job search, $t(1076)=2.84$, $p=.004$.

Problem Solving in the Workplace

When asked how respondents reacted to getting "stuck on a problem", there were many shared strategies among women and men. Both genders were equally likely to consult online communities like Stack Overflow, watch help videos, or do other distracting activities such as meditate or walk. Men were more likely to indicate they would play games while working through a problem, $t(1391)=-2.47$, $p=.013$. Women were more likely to indicate that they would ask a coworker for help, $t(1347)=3.66$, $p<.001$, focus on other work as a distraction, , $t(1399)=6.29$, $p<.001$, or panic, $t(1229)=8.63$, $p<.001$.

Criteria for Job Search

When asked which three factors were important when deciding between two theoretical jobs with similar pay, benefits, and location, there were some factors that were selected equally by both genders and some that were significant to a respective gender. Selecting factors such as ability to work remotely, professional development, department or team, flex time, and family friendliness were as likely to be selected as one of the three most important factors in a job search by both sexes.

Men were more likely to indicate that the following three factors were most important to consider: financial performance of a company, $t(1421)=-5.51$, $p<.001$, impact of contributions, $t(1304)=-2.79$, $p=.005$, industry, and technologies used, $t(1296)=-9.06$, $p<.001$. Women were more likely to select office environment/company culture, $t(1269)=2.63$, $p<.001$, and company diversity, $t(1486)=10.86$, $p<.001$, as reasons to distinguish between two jobs.

Other factors that were not selected at significant levels between men and women include remote work options, professional development, specific department or team assignment, flexible schedules and family friendliness.

5. DISCUSSION

Overall, results from the survey were similar to findings from previous studies (Allen, et al., 2006; Berg, Sharpe, & Aitkin, 2018), and support the idea that women are less satisfied and participate less in their IT jobs than their male counterparts. Looking at reasons that might cause employees to look for other jobs or criteria used in selecting jobs provides further insight into why some of these differences may exist.

From a general standpoint, the importance of interpersonal relationships appeared to have more importance in women's responses than men. Company culture, diversity, problems in the workplace regarding team members and managers were more likely to be cited as reasons to search for a new job by women, whereas more concrete issues of salary and new technologies tended to be more likely to be cited by men. While both men and women turned to colleagues for help when stuck on a problem, women were more likely to rely on other people for help than men.

What was surprising, however, was that issues that have appeared to carry importance in other research were not present in this study. Issues such as flexible schedules, remote work schedules, and family friendliness were of similar importance to both men and women respondents. Additional data about family status and current work environments weren't available in this study but could provide additional insights about the results. Additionally, this survey was conducted in February of 2020, right before the pandemic shutdowns began, which significantly changed work environments in all industries.

From an education perspective, it is not surprising that women were more likely to have a bachelor's degree than male respondents given the general statistics about female vs. male enrollment in US colleges and universities. Of the respondents that did have a bachelor's degree, men were more likely to have majored in a computer related field with almost half of the respondents coming from other areas of study. This also mirrors the findings reported by the National Center for Education Statistics (2021b).

So, what, as educators, can we take away from this snapshot of men and women in the workplace? Can we do something to increase diversity recruitment and retention of female technologists? There are a few areas from previous research that are supported by this research as well.

Recruitment of Female Students in IT Careers

As shown by many other studies, there are many reasons women do not choose IT as a major and there are as many studies trying to change that. Perhaps the answer does not lie in recruiting women to the IT field but in bringing IT to their chosen fields. Almost 50% of women tech professionals responding to the survey had college degrees in areas other than IT. They eventually discover, either through interest or necessity or other life circumstances, that they can be interested and successful in the IT field. Departments that are looking to increase gender diversity in their classrooms might consider minors or certificate programs in conjunction with other majors in the university. This exposes more students to information technology education and potentially opens the door of opportunity for more women to choose careers in IT.

If circumstances and resources allow, creating cohort groups in introductory computing classes that require students from all majors to explore how technology is used specifically in their chosen field of study could be helpful. Not only will this benefit students in that field but potentially will spark an interest in students that might not have considered a career in information technology. Doing so may allow many women to receive the technical training and confidence in their skills that might transition them to an IT career, sooner than later. Presenting different problems across multiple disciplines could help disperse the "geeky" stereotypes of computer nerds writing code only to solve science problems and broaden horizons to include how technology plays an important role in all fields from business to the arts and humanities.

Retention of Female Students in IT Programs

Because issues like panic in the face of a problem, as well as dissatisfaction or issues with interpersonal relationships at work were significant concerns of female respondents, a strong support system for women in IT should be instituted at the educational as well as professional level. Participation in student groups and professional memberships should be strongly encouraged for all students, and women in particular. Participation in these activities will allow them to practice their networking skills, both formal and informal, that can help mitigate advancement issues they may encounter in the workplace.

In addition, there are case studies supporting the formulation of women specific professional organizations to make selection of IT as a career more appealing (Wang, Goldgof, & Christensen, 2019; Heistand-Tupper, Leitherer, Sorkin, & Gore 2010). Encouraging female participation in student or professional groups for women allows students to find mentors and build networks that will help them in their future career. If there aren't that many or any female faculty in a department, consider teaming up with other STEM related departments on campus or encourage participation in local or national organizations intended to promote women in computer related fields. Also consider encouraging female students to participate more in online communities like Stackflow or other industry specific communities at an early stage to normalize this behavior and open networking and help opportunities that may benefit them in the workplace.

Interpersonal Skills

While interpersonal skills rarely appear in the curriculum, there are ways to consider preparing our students, both male and female, how to be successful in the workplace and this carries on beyond pure technical skills. Many programs emphasize concepts such as group work, written ability, and public speaking. When considering these soft skills, also consider some of the issues presented in this study. When assigning group projects, try to create diverse groups when possible. If assigning roles within the group, give female students a chance to lead her male colleagues and give male students the chance to interact in meaningful ways with their female colleagues. If recruitment efforts in higher education are successful and women choose computer related majors with more frequency, the ability to interact and become familiar with colleagues of all genders, races, and backgrounds should mitigate some of the issues currently faced by women in the technology field.

Future Studies

While this study is just a snapshot of some opinions held by professionals in the workplace, it gives us a glimpse of some of the issues that affect a woman's satisfaction and success in her career. A broader, more nuanced look at some of these issues would provide even more information on how the industry is changing and can reveal other steps that can be made to improve women's place in IT.

This study did not examine any aspect beyond gender. Other factors such as women of color help increase the diversity of our field, yet they are even rarer participants than the scope of this study. While women are chronically underrepresented in this field, other groups such as African-American women make a small fraction of the technology workforce and tend to leave at greater rates. These issues, along with gender should also be studied to increase participation and diversity.

Another aspect that limits this study is the concept of gender, in general. This study looked at binary genders only for ease and clarity of results. Respondents that indicated non-binary or multiple gender affiliations were not included. With the increase and acceptance of gender fluidity of younger generations, the research done in this field to this point could see major shifts as younger people reconceptualize the concept of gender overall.

6. CONCLUSION

As mentioned before, this study gives a snapshot of women's feelings about their jobs in IT fields, but it is just that, a snapshot. It gives us ideas of where problems might lurk and where opportunity exists. This study supports findings from many previous areas of research but also provides some surprising divergence. Issues that many times are considered "women's issues" such as family support and flexible work schedules, really are employee concerns across all genders. With a growing demand for technical talent, higher education and the tech industry should focus on encouraging and including women to increase participation in areas of technical expertise. Researchers should continue to identify strategies to recruit women to the field and retention strategies to retain and advance women to make IT a richer, more inclusive industry.

7. REFERENCES

Allen, M.W., Armstrong, D.J., Riemenschneider, C.K., & Reid, M.F. (2006). Making sense of the barriers women face in the information technology work force: standpoint theory, self-disclosure, and causal maps. Sex Roles, 54, 831-844.

Ashcraft, C., McLain, B. & Eger, E. (2016). Women in tech: the facts 2016 Update. National Center for Women and Information Technology. https://wpassets.ncwit.org/wp-content/uploads/2021/05/13193304/ncwit_women-in-it_2016-full-report_final-web06012016.pdf

Bapna, S., & Funk, R. J. (2021). Interventions for Improving Professional Networking for Women: Experimental Evidence from the It Sector. MIS Quarterly, 45(2), 593–636.

Barker, L., Mancha, C. & Ashcroft, C. (2014). What is the impact of gender diversity on technology performance? Research summary. National Center for Women & Information Technology. https://www.ncwit.org/businesscase

Bear, J. B., & Woolley, A. W. (2011). The role of gender in team collaboration and performance. Interdisciplinary Science Reviews, 36(2), 146–153.

Berg, T., Sharpe, A., Aitkin, E. (2018). Females in computing: understanding stereotypes through collaborative picturing. Computers & Education, 126(2018), 105-114.

Bureau of Labor Statistics (2021). Occupational Outlook Handbook, Computer and Information Technology Occupations. U.S. Department of Labor. https://www.bls.gov/ooh/computer-and-information-technology/home.htm

Chamberlain, A. (2016). The widest gender pay gaps in IT. Glassdoor Economic Research. https://www.glassdoor.com/research/gender-pay-gap-in-tech/

DuBow, W. & Gonzalez, J.J. (2020) NCWIT Scorecard: The Status of Women in Technology. Boulder, CO: NCWIT. https://ncwit.org/resource/scorecard/

Heistand-Tupper, D., Leitherer, B., Sorkin, S., & Gore, M.E. (2010). Strategies for increasing IT enrollment: recruiting, retaining, and encouraging the transfer of women and underrepresented groups to four-year colleges. Information Systems Education Journal, 8(54), 3-21. http://www.isedj.org/8/54/

Hewlett, S.A., Luca, C.B., Servon, L.J., Serbin, L., Shiller, P., Sosnovich, E. & Sumberg, K. (2008). The Athena Factor: Reversing the Brain Drain in Science, Engineering, and Technology. Center for Work-Life. Harvard Business Review.

Hewlett, S.A., Sherbin, L., Dieudonné, F., Fargnoli, C., & Fredman, C. (2014). Athena Factor 2.0: Accelerating female talent in science, engineering, & technology. New York: Center for Talent Innovation. https://coqual.org/reports/athena-factor-2-0-accelerating-female-talent-in-science-engineering-technology/

Igbaria, M., & Baroudi, J.J. (1995). The impact of job performance evaluations on career advancement prospects: an examination of gender differences in the IS workplace. MIS Quarterly, 19(1), 107-123.

Jung, L., Clark, U., Patterson, L., & Pence, T. (2017). Closing the gender gap in the technology major. Information Systems Education Journal, 15(1), 26-41. http://www.isedj.org

Kleinbaum, A.M,, Stuart, T.E., & Tushman, M.L. (2013). Discretion within constraint: Homophily and structure in a formal organization. Organization Science 24(5), 1316-1336.

McGee, K. (2017). The influence of gender, and race/ethnicity on advancement in information technology (IT). Information and Organization, 28(1), 1-36.

National Center for Education Statistics (2021a). College enrollment rates. https://nces.ed.gov/programs/coe/indicator/cpb.

National Center for Education Statistics (2021b). Undergraduate degree fields. https://nces.ed.gov/programs/coe/indicator/cta.

National Center for Education Statistics (2021c). Undergraduate retention and graduation rates. https://nces.ed.gov/programs/coe/indicator/ctr.

Othman, M., & Latih, R. (2006). Women in computer science: no shortage here! Communications of the ACM, 49(3), 111-114.

Serapiglia, C.P., & Lenox, T.L. (2010). Factors affecting women's decisions to pursue an IS

degree: a case study. Information Systems Education Journal, 8(12). http://isedj.org/8/12/

Shein, E. (2018). Broadening the path for women in STEM. Communications of the ACM, 61(8), 19-21.

Stack Overflow. (2020). 2020 Developer Survey. https://insights.stackoverflow.com/survey/2020

Roldan, M., Soe, L., & Yakura, E. (2004). Perceptions of chilly IT organizational contexts and their effect on the retention and promotion of women in IT. SIGMIS CPR '04.

Women's Bureau (n.d).Occupations with the largest projected increase in jobs by share of women in the occupation. U.S. Department of Labor. https://www.dol.gov/agencies/wb/data/high-demand-occupations.

Appendix A

Survey Questions (Stack Overflow, 2020)

1. Which of the following best describes the highest level of formal education that you've completed?
   I never completed any formal education; Primary/elementary school; Secondary school (e.g. American high school, German Realschule or Gymnasium, etc.); Some college/university study without earning a degree; Associate degree (A.A., A.S., etc.); Bachelor's degree (B.A., B.S., B.Eng., etc.); Master's degree (M.A., M.S., M.Eng., MBA, etc.); Professional degree (JD, MD, etc.) Other doctoral degree (Ph.D., Ed.D., etc.); Other (please specify):
2. What was your primary field of study?
   Computer science, computer engineering, or software engineering; *Web development or web design; *Information systems, information technology, or system administration; Mathematics or statistics; Another engineering discipline (such as civil, electrical, mechanical, etc.); A business discipline (such as accounting, finance, marketing, etc.); A health science (such as nursing, pharmacy, radiology, etc.); A humanities discipline (such as literature, history, philosophy, etc.); A natural science (such as biology, chemistry, physics, etc.); A social science (such as anthropology, psychology, political science, etc.); Fine arts or performing arts (such as graphic design, music, studio art, etc.); I never declared a major
3. How important is a formal education, such as a university degree in computer science, to your career?
4. Which of the following best describes your current employment status?
   Employed full-time; Employed part-time; Independent contractor, freelancer, or self-employed; Not employed, but looking for work; Not employed, and not looking for work; Student; Retired;I prefer not to say
5. Imagine that you are deciding between two job offers with the same compensation, benefits, and location. Of the following factors, which 3 are MOST important to you?
   Remote work options, Office environment or company culture, Financial performance or funding status of the company or organization, Opportunities for professional development, Diversity of the company or organization, How widely used or impactful my work output would be, Industry that I'd be working in, Specific department or team I'd be working on, Flex time or a flexible schedule, Languages, frameworks, and other technologies I'd be working with, Family friendliness
6. In general, what drives you to look for a new job? Select all that apply.
   Curious about other opportunities, Better compensation, Having a bad day (or week or month) at work, Wanting to share accomplishments with a wider network, Looking to relocate, Just because, Trouble with my direct manager, Better work/life balance, Wanting to work with new technologies, Growth or leadership opportunities, Trouble with my teammates, Trouble with leadership at my company
7. What do you do when you get stuck on a problem? Select all that apply
   Visit Stack Overflow, Call a coworker or friend, Play games, Meditate, Go for a walk or other physical activity, Watch help / tutorial videos, Do other work and come back later, Panic, Visit another developer community (please name)

Likert Scale Questions with Response Ranges from 1-5 with 5 being positive response

8. How satisfied are you with your current job? (If you work multiple jobs, answer for the one you spend the most hours on.)
9. How often do you work overtime or beyond the formal time expectation of your job?
10. What level of influence do you, personally, have over new technology purchases at your organization?
11. Do you consider yourself a member of the Stack Overflow community?
12. How frequently would you say you participate in Q&A on Stack Overflow? By participate we mean ask, answer, vote for, or comment on questions.
13. Compared to last year, how welcome do you feel on Stack Overflow?

Appendix B

Statistical Results

Descriptive Statistics of Satisfaction Construct

Gender	N	Mean	Std. Deviation	Std. Error Mean
Male	8033	2.7682	.49902	.01640
Female	926	2.6043	.53842	.00601

Independent Samples Test of Satisfaction Construct (Equal Variances not Assumed)

t-test for Equality of Means							
t	df	Sig. One-Sided	Sig. Two-Sided	Mean Difference	Std. Error Difference	95% Confidence Interval if the Difference Lower	95% Confidence Interval if the Difference Upper
9.384	1187.456	<.001	<.001	-.1737	.02141	-.21522	-.13111

Question: Imagine that you are deciding between two job offers with the same compensation, benefits, and location. Of the following factors, which 3 are MOST important to you?

Option	Gender	N	Mean	SD	Diff(F-M)	t	df	p
Remote work options	F	1014	0.38	0.4867443	0.0146986	0.91	1265	0.363
	M	8410	0.37	0.4828106				
Office environment or company culture	F	1014	0.52	0.4999314	0.043673	2.63	1269	0.008*
	M	8410	0.47	0.4993573				
Financial performance or funding status of the company or organization	F	1014	0.08	0.2697086	-0.0508311	-5.51	1421	<.001*
	M	8410	0.13	0.3360223				
Opportunities for professional development	F	1014	0.33	0.4705801	-0.0061294	-0.39	1272	0.695
	M	8410	0.34	0.4725417				
Diversity of the company or organization	F	1014	0.28	0.4473046	0.2125194	14.86	1086	<.001*
	M	8410	0.06	0.2440799				
How widely used or impactful my work output would be	F	1014	0.20	0.4010827	-0.037461	-2.79	1304	0.005*
	M	8410	0.24	0.4262803				
Industry that I'd be working in	F	1014	0.14	0.3471992	-0.0250022	-2.15	1308	0.031*
	M	8410	0.17	0.3712402				
Specific department or team I'd be working on	F	1014	0.15	0.36003	-0.0073065	-0.61	1279	0.542
	M	8410	0.16	0.3667822				
Flex time or a flexible schedule	F	1014	0.43	0.4957269	0.0227129	1.38	1265	0.168
	M	8410	0.41	0.4919038				
Languages, frameworks, and other technologies I'd be working with	F	1014	0.35	0.4766114	-0.1442638	-9.06	1296	<.001*
	M	8410	0.49	0.4999718				
Family friendliness	F	1014	0.11	0.3160458	-0.0169438	-1.60	1304	0.109
	M	8410	0.13	0.3356287				

Question: In general, what drives you to look for a new job? Select all that apply.

Option	Gender	N	Mean	SD	Diff (M-F)	t	df	p
Curious about other opportunities	F	860	0.60	0.4904191	0.0047673	0.27	1065	0.787
	M	7521	0.59	0.4911038				
Better compensation	F	860	0.67	0.4709888	-0.0682654	-4.05	1038	<.001*
	M	7521	0.74	0.4403616				
Having a bad day (or week or month) at work	F	860	0.31	0.4629533	0.0695397	4.20	1033	<.001*
	M	7521	0.24	0.4276736				
Wanting to share accomplishments with a wider network	F	860	0.06	0.2341433	-0.0123298	-1.45	1107	0.147
	M	7521	0.07	0.2559534				
Looking to relocate	F	860	0.29	0.4527234	0.0500068	3.09	1039	0.002*
	M	7521	0.24	0.4253957				
Just because	F	860	0.13	0.3405503	-0.0085474	-0.70	1076	0.486
	M	7521	0.14	0.3493483				
Trouble with my direct manager	F	860	0.29	0.4554139	0.0800197	4.93	1024	<.001*
	M	7521	0.21	0.4094574				
Better work/life balance	F	860	0.45	0.4976649	0.036525	2.04	1060	0.041*
	M	7521	0.41	0.4922835				
Wanting to work with new technologies	F	860	0.49	0.5002247	-0.079872	-4.44	1060	<.001*
	M	7521	0.57	0.4948606				
Growth or leadership opportunities	F	860	0.65	0.4783308	0.0489847	2.84	1076	0.004*
	M	7521	0.60	0.4904289				
Trouble with my teammates	F	860	0.22	0.41433	0.0910612	6.22	991	<.001*
	M	7521	0.13	0.3348968				
Trouble with leadership at my company	F	860	0.38	0.4851463	0.0314105	1.80	1056	0.071
	M	7521	.35	0.4758853				

Question: What do you do when you get stuck on a problem? Select all that apply.

Option	Gender	N	Mean	SD	Diff (M-F)	t	df	p
Visit Stack Overflow	F	1077	0.90	0.2942281	0.0049818	0.52	1357	0.600
	M	9064	0.90	0.3008386				
Call a coworker or friend	F	1077	0.55	0.4977574	0.0587218	3.66	1347	<.001*
	M	9064	0.49	0.4999457				
Play games	F	1077	0.13	0.34149	-0.0274365	-2.47	1391	0.013*
	M	9064	0.16	0.3685351				
Meditate	F	1077	0.11	0.3147985	-0.0120348	-1.18	1370	0.238
	M	9064	0.12	0.3289774				
Go for a walk or other physical activity	F	1077	0.60	0.4909058	0.0236157	1.49	1349	0.136
	M	9064	0.57	0.4947454				
Watch help / tutorial videos	F	1077	0.53	0.4990179	0.0306265	1.90	1345	0.057
	M	9064	0.50	0.50001				
Do other work and come back later	F	1077	0.75	0.4325414	0.0885393	6.29	1399	<.001*
	M	9064	0.66	0.4728414				
Panic	F	1077	0.23	0.4229732	0.1150054	8.63	1229	<.001*
	M	9064	0.12	0.3226844				
Visit another developer community (please name)	F	1077	0.13	0.3323153	0.0222387	2.09	1301	0.036*
	M	9064	0.10	0.3053266				

Redesigning Assessment in the Computer Science Program at a Regional University

Ernst Bekkering
bekkerin@nsuok.edu

Patrick Harrington
harringp@nsuok.edu

Mathematics and Computer Science
Northeastern State University
Tahlequah, OK 74464

Abstract

This paper describes the development of the assessment program for the Computer Science program at a regional university in the Southwest. Assessment has become a major component in higher education. It is both a requirement for funding and accreditation, a signal of the quality of education provided, and a tool for improvement. Starting with an assessment program consisting of a single measurement at the end of the program, we have developed an assessment program that monitors student learning continuously throughout the curriculum. The new program allows for faster and more accurate identification of potential for improvement. The process of assessment program conversion and results from the first year are discussed in this paper.

Keywords: learning outcome assessment, learning outcomes, measurement, outcome-based education, program outcomes, accreditation.

1. INTRODUCTION

At the national level, outcomes assessment has gained attention through the "No Child Left Behind" (Boehner, 2002) and "Race to the Top" (U.S. Department of Education, 2016) programs. This movement has expanded to higher education (American Association of University Professors, 2015), where regional accrediting agencies, school accreditation agencies, and program accreditation agencies alike have instituted assessment as one of the core accreditation requirements. So far, only university-wide accreditation is a requirement to receive federal funding, whereas school accreditation and program accreditation are still voluntary. University accreditation may be the minimum standard, but assessment is a required component of all levels of accreditation.

Despite diverting attention from teaching and research, assessment is an opportunity for faculty to develop new models of assessment and analyze and use its results in a way that results for implementing meaningful change. This compares to formal definitions of assessment, which clearly include "closing the loop." The following definitions of assessment clearly include action at the end of the cycle. Osters (2003) defines assessment as "a systematic and ongoing process of collecting, interpreting, and acting on information relating to the goals and outcomes developed to support the mission and purpose of an institution." Likewise, Suskie (2018) describes assessment as "a cyclical, outcome-oriented process where faculty develop clearly written, measurable, and achievable program outcomes which are then systematically measured and evaluated with the results driving student learning and program improvement." It is the

latter activities that make assessment meaningful, but also the most difficult (Blaich & Wise, 2011; Fulcher et al., 2014).

In the United States, students in four-year programs typically spend the first two years in General Education (Gen Ed) courses, and complete major/minor specific courses in the junior and senior years. Assessment is important at both levels, and they can intersect. Outcome assessments for general education are embedded in Gen Ed courses, and occasionally in major/minor courses. For instance, communication can be measured in COMM 1113 Fundamentals of Oral Communication, but also in the capstone presentation of CS 4233 Professional Development in Computer Science (CS capstone). University-wide committees are responsible for assessment at the General Education level, whereas department-level committees conduct the program-specific assessment.

This paper is organized as follows. In the next section, we discuss relevant literature. In section 3, we discuss the development of our assessment program. Results of the first year's data collection are discussed in section 4. We end with conclusions and recommendations.

2. LITERATURE REVIEW

Assessment takes place at multiple levels. In this paper we describe assessment in one specific academic program, Computer Science. There are broader implications for our work that extend beyond assessment in one department.

Our university's assessment plan, last updated in 2017, is grounded in the university mission and vision. It aligns to the institutional priorities and strategic goals as articulated in the strategic plan. At this level, Degrees of Excellence (DOE) serve as foundation for assessment at the institutional level. They include intellectual skills, integrative knowledge, specialized knowledge in the major, integration of knowledge in the major with degree expectations, and citizenship. The degrees build upon the Lumina Foundation's Degree Qualification Profile (Lumina Foundation, 2021) and the Essential Learning Outcomes of the Association of American Colleges and Universities (Association of American Colleges & Universities, 2020).

Types of accreditation

Accreditation takes place in six regional accreditation agencies at the university, school, and program levels. These agencies include accrediting bodies such as the regional NEASC and HLC, school-level ACBSP, and ABET for programs and majors.

The United States has six regional accreditation agencies (Figure 1). The Higher Learning Commission is a regional accreditation agency is one them (New England Commission of Higher Education, 2021). It accredits granting institutions of higher education based in the 19-state North Central region. Members are evaluated against five criteria (Higher Learning Commission, 2021):

1. Mission
2. Integrity: ethical and responsible conduct
3. Teaching and learning: quality, resources, and support
4. Teaching and learning: evaluation and improvement
5. Resources, planning, and institutional effectiveness.

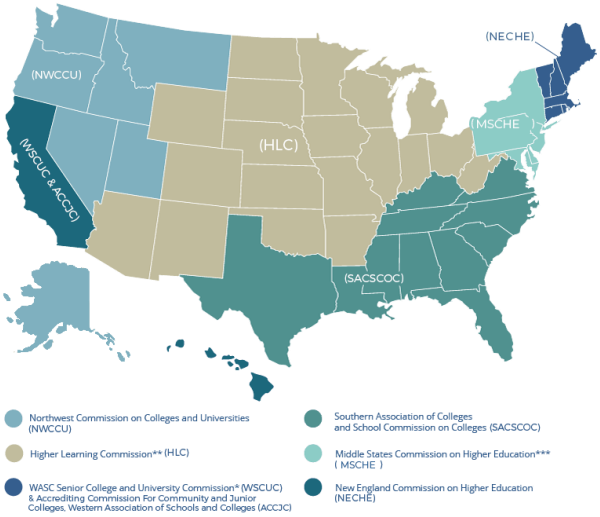


Figure 1: Regional Accreditation Agencies. Source:chea.org

Program level assessment falls under the fourth criterium.

Program assessment is also required by the state's Regents for Higher Education. Regents expect assessment in four categories:

- Entry level to determine academic preparation and course placement
- General education assessment
- Academic program learning outcomes assessment
- Student engagement and satisfaction.

Notice that again, assessment is required at the program level but not at the course level within the programs.

Benefits of accreditation

Since accreditation demands sometimes significant effort and expense, it must be done in a way to maximize benefit. Most literature mentions indirect benefits, like stakeholder confidence, which cannot be directly measured. However, accreditation makes a program not only more attractive to potential students but can help with certifying excellence at a measurable level. At a more ordinary level, a direct benefit is that accreditation enables obtaining federal funds like student aid (Eaton, 2015). Financial Aid in all forms continues to climb from year to year (Educationdata.org, 2021), a trend higher education experts hope will remain (Washington Post, 2021). The Federal Government in return, pressured accreditation agencies to adopt an outcomes-based approach (Banta, 2001).

These benefits do not come without a price. Institutions face financial costs to the accrediting agency, burdens on staff and faculty, and potential threats to academic freedom (uPlanner, 2016). Faculty are frequently concerned about the impact of assessment on their workload (Hogan et al., 2002). At the program level, faculty may not have much support for accreditation either. Hilton et al. (2004) found the understanding of potential benefits of accreditation to be quite low among program administrators. Finally, accreditation can cause a loss of flexibility for program improvement. For example, ACBSP accreditation in another college, our College of Business, requires a common set of business courses for all BS degrees for a total of 42 hours, which limits the number of courses Information Systems (IS) students can take within their major. Overall though, the benefits of accreditation outweigh the limitations (Babb & Abdullat, 2014) and accreditation requirements are applicable to all programs dedicated to continuously delivering a quality curriculum, whether they are accredited or not (Challa et al., 2004).

Course vs. program outcomes assessment

In higher education, we can distinguish three levels of learning outcomes: course outcomes, program outcomes, and institutional outcomes. Occasionally, assessment at the school level is mentioned (Al-Mubaid et al., 2011). Course level outcomes are formal statements that describe what students are expected to learn in a course. They refer to specific knowledge, practical skills, areas of professional development, attitudes, higher-order thinking skills, etc. that faculty members expect students to develop, learn, or master during a course (Suskie, 2018). They can also be referred to as "learning outcomes", "student learning outcomes", or "learning outcomes statements."

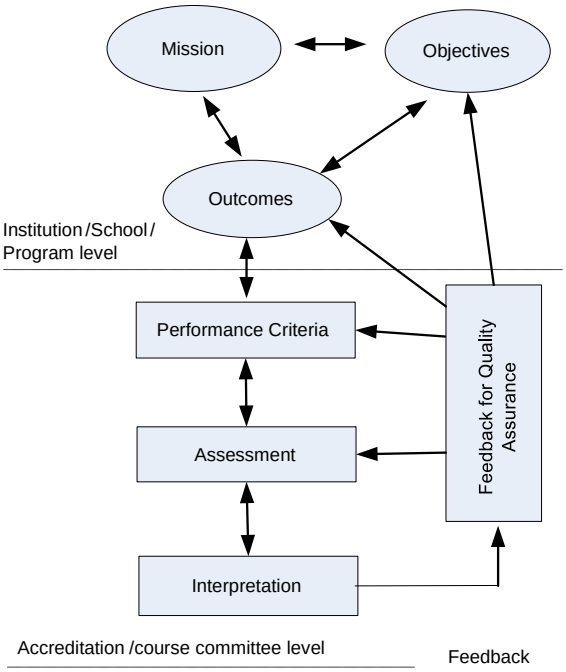


Figure 2: Assessment Model

The relationship between learning outcomes and the assessment process is discussed by Al-Mubaid et al. (2011). It consists of three components: guidance components (mission, objectives, and outcomes), evaluation components (performance criteria, performance of assessment processes, and interpretation of results), and feedback (Figure 2).

Whereas course level outcomes are narrow in scope (students achieve the outcomes as they complete a single course), program level outcomes are broad in scope (students achieve them as they complete the program at time of graduation). Lending et al. (2019) mention commitment to continuous assessment as one of the five essential ingredients of high-quality majors. For both course and program level assessment, it is important to use verbs that describe accurately what students will be able to do. Traditionally, higher education uses action verbs to indicate the level of performance (Anderson et al., 2001; Bloom, 1956).

The Institutional Effectiveness Plan (Figure 3) shows the relationship between the different types of outcomes.

Figure 3: Outcome alignments

Outcomes measurement
Useful outcomes must be specific and measurable (Spady and Marshall 1994).

Program learning outcomes specify what students will know or be able to do at completion of the program. They need to meet the S.P.A.M criteria:

- Specific – start with a verb and target a single key competency
- Purposeful – relevant to students and the program
- Attainable – can be completed within a reasonable time
- Measurable – either direct or indirect measurement.

Course grades are not good measure of meeting goals because letter grades or scores do not closely correlate with strengths and weaknesses of individual learning outcomes. Rubrics, on the other hand, can combine the individual elements of multiple outcomes to a single score.

Types of measurement
Depending on the learning outcome, measurements can take multiple forms. We will briefly cover measures relevant to our program.

Measurements can use student work directly or indirectly. Direct measures use actual student work to produce evidence of student performance in the learning outcomes. Examples include performance assessments, capstone projects, senior theses, exhibits or performances, and standardized exams. Indirect measures use secondary information that do not rely on actual student work. Examples include satisfaction surveys, exit interviews, and focus groups. DFIW rates (proportion of D, F, Incomplete, Withdrawn grades) can serve to identify potential barriers to student success (Suskie, 2018).

Direct measures can be embedded in single courses or used across courses. The advantage of embedded assessment is the high level of student motivation and the reduced burden on faculty to collect and analyze data. Course grades and test scores are not appropriate measurements, because grades give specific students feedback about performance, and assessment measures performance for all students as a group. Specific test questions or parts of larger work products can be used if they directly affect a Program Learning Outcome (PLO).

Capstone projects are good examples of cross-course measures. Projects combine the skills learned in multiple courses. A working program may be evidence of programming skills, and the documentation may be evidence of communication skills.

Indirect measures are harder to use and potentially more problematic. Employer surveys are only useful if respondents have first-hand knowledge of student work. Focus groups represent a limited number of opinions. Exit interviews with graduating students may not produce honest responses while still studying, and collection of data is harder after graduation. Perhaps the greatest use is not an assessment of how the program meets current students' needs, but to provide guidance about the desired level of performance after graduation.

Summative vs. formative assessment
Finally, the timing of assessment is relevant. Assessment can take place at the end of a single course or major program, but also during. Formative assessment takes place during the learning activity and benefits current students because there is still time for corrections. At the course level, an example might be a draft of a paper. At the program level, assignments in programming classes identify strengths and weaknesses in programming skills. Summative assessment takes place at the end when current students can no longer benefit but future students can. It evaluates courses and programs as a whole. For instance, capstone projects can be used to measure a wide variety of program objectives (Murray et al., 2008). In the examples above, the finished paper and capstone project can be used for summative assessment (Suskie, 2018, p. 157). For the health of a program, formative and summative assessments are equally important.

3. METHODOLOGY

This section describes the development of the current assessment plan for the Computer Science program.

Old assessment program
At the time of the last Higher Learning Commission accreditation visit in 2011, the CS program assessment was based on an exit exam from Educational Testing Services (2021) in paper and pencil form. Over time, we considered that the typical questions on the exam did not always reflect our program and attempted to devise a homegrown exit exam with typical questions from constituent courses in the program. Faculty members selected representative multiple-choice questions from core courses in the major, and after some discussion the final selection of questions was made.

Both tests suffered from two shortcomings, that of timing and relevance. Since they were administered one time only at the end of the last semester, had no consequences for non-participation and poor performance, and reflected material from courses taken in the past, the performance on these tests was low. Moreover, it was impossible to draw any conclusions that might lead to actionable results. Changes in the curriculum were the result of discussions among faculty, and not based on measurement of student learning.

New program
In 2017, the new university assessment plan was initiated. The plan defines steps in the learning outcomes assessment process as follows:
1. Articulate program mission and vision
2. Establish program learning outcomes
3. Develop outcome measures
4. Align program PLOs, outcome measures, and curriculum
5. Engage learners
6. Gather and analyze data
7. Share and report data
8. Close the loop by making evidence-based decisions

With this framework in mind, the process of preparing the assessment function for the next Higher Learning Commission accreditation visit started.

Computer Science program mission
First, we discussed our mission and vision statements in department meetings and made some slight revisions. The new mission statement became "The Department of Mathematics and Computer Science at (our university) will offer mathematics and computer science courses of the highest quality to both majors and non-majors through the faculty's dedication to excellence in teaching, commitment to scholarship, and determination to bring new knowledge into the classroom so that each student will be prepared for advancement in his or her own field of study."

Assessment plan template
The assessment plan template went through minor revisions at the administrative level. In the Spring of 2019, the current assessment plan template was introduced by the Executive Director of Planning and Assessment. At the program level, it includes the mission statement, PLOs, a curriculum map that connects the PLOs with program courses and experiences, an assessment map that connects the PLOs with assessment measures and their performance targets, and a sample plan to increase consistency between programs.

Program Learning Outcomes
Prior to the start of the Fall 2019 semester, faculty and staff attended a keynote presentation by an experienced assessment speaker, and spent the rest of the day collaborating and discussing program goals and assessment instruments. Program Learning Outcomes were reviewed and revised to make them more specific, specifically to make them more measurable. **Table 1** compares the old and new program outcomes. In general, we replaced the verb "demonstrate" with a Bloom's action verb appropriate for the performance level we wished to pursue. We also included algorithms as one of the major components of computer science, to be measured at two different levels considering their importance.

Outcome measures
In developing our new assessment program, we moved away from one-time measurement to measuring student learning throughout the program. Program level outcomes must be addressed throughout the curriculum, and students need to have at least two courses where they can work towards these goals (Suskie, 2018, p. 55). The new assessment template reflected this distributed measurement. Over the course of the curriculum, students progress to the PLO in different stages. When the PLO is introduced (I), students acquire basic disciplinary knowledge and skills. Course activities help them to remember and/or understand. When the PLO is reinforced (R), students integrate skills with increasing complexity. Course activities give them the opportunity to apply and/or analyze. Finally, when the PLO is mastered (M), knowledge and skills are applied to address complex disciplinary questions and problems. Course activities focus on the advanced ability to evaluate and create.

Old Program Outcomes	New Program Outcomes
Demonstrate competency in multiple modern programming languages	Program in two or more computer languages
Demonstrate knowledge and understanding of the theory in core areas of Computer Science	Explain core elements of computer science theory
Demonstrate critical thinking and problem-solving skills through the application of software development process	Create software using software engineering principles
Demonstrate the ability to design and implement a computer system project	Solve problems by applying the software development process
Demonstrate the ability to use current techniques, skills, and tools necessary in the computing practice	Apply current techniques, skills, and tools necessary in the computing practice
Demonstrate the ability to learn new programming languages and acquire new skills independently	
Demonstrate effective written and oral communication skills	Communicate technical ideas both orally and in writing
Demonstrate the ability to collaborate effectively as a member of a team	Collaborate as a member of a team
	Design computer algorithms to solve problems
	Apply computer algorithms to solve problems

Table 1: Old and new program outcomes

Computer Science curriculum

Traditionally, Computer Science programs are rooted in either Electrical Engineering or Mathematics. Our university does not have an EE program, and the mathematics component is deliberately kept small so students can focus on programming skills. This is in line with proficiency in programming and software development, which is especially important in most CS programs (Association of Computing Machinery, 2020). Some of the courses in the program are pulled from other majors where they are more appropriately taught, such as Technical Writing, which is taught in the English department. Table 2 presents a list of required and elective courses.

CS General Education Requirements	
ENGL 1113	English Composition I
ENGL 1213	English Composition II
MATH 1513	College Algebra
MATH 3023	Discrete Mathematics
CS Major Courses (31 hours)	
CS 2014	Computer Science I
CS 2163	Computer Science II
CS 3033	Object Oriented Programming
CS 3173	Basic Computer Architecture
CS 3343	Computer Operating Systems
CS3403	Data Structures
CS 4343	Database Management Systems
ENGL3083	Professional & Technical Writing
CS 4203	Software Engineering
CS 4233	Professional Development in CS
Select CS Electives (11 hours)	
CS 3023	Object Based Visual Programming
CS 3100	Select Topics
CS 3203	Application Development in C++
CS 3643	Programming for Cyber Security
CS 3663	Principles of TCP/IP
CS 4100	Advanced Select Topics
CS 4113	Computer Science Internship
CS 4143	Web Applications
CS 4223	Game Programming
CS 4253	Windows Programming
CS 4553	Parallel Programming

Table 2: CS Courses

Alignment with curriculum

Aligning PLOs, outcomes measures and curriculum makes the connection between what students will learn, how they will learn it, and how students demonstrate that they learned it. For instance, collaboration in a team requires team activities, and students can either rate each other or themselves (self-evaluation). Measuring team collaboration with a multiple-choice or essay test would not be appropriate. Writing software programs is best evaluated using actual programming assignments, and we developed a rubric that can be used to score essential elements of software programs (Appendix E**Error! Reference source not found.**).

In the process of alignment, faculty decide which courses will be used for measuring specific

outcomes. In Information Systems, this process has been described by Veltri et al., (2011). Distribution of outcomes measurement over courses in the program is paramount. In our first draft of the curriculum map, the capstone course was assigned eight of nine PLOs, and three other courses only one. Not only did this provide a disproportionate burden on the faculty teaching the capstone course, but the measurement was done at a point in the program where few changes could be made if the students did not perform as expected. In the revised version, assessment activities were much more evenly distributed. Faculty needed to include no more than three assessment measurements in a course, and the total number of measurements was cut in half. This does not mean that students could not pursue learning outcomes in other courses; those specific outcomes were just not measured for assessment purposes. Appendix C shows the old and new curriculum mapping.

Next came development of the assessment measures. Since we had significant investment in development of the departmental exit exam, we started with its multiple-choice questions. We assigned them to the course from which they came, usually at the introduction level for the appropriate outcome. This resulted in nine outcome/course combination levels covered. We developed four rubrics: programming assignments, self-evaluation of group participation, oral communication, and capstone project evaluation. For the remainder of the measurements, we developed additional multiple-choice questions. In the process, we discovered that all relevant material was covered in outcomes 1 and 3-9, and that outcome 2 (Apply current techniques, skills, and tools necessary in the computing practice) was superfluous. We decided that it could be deleted and removed it from the program outcomes. Appendix A lists all outcomes at different levels for assessment. Appendix D provides a listing of all measurement items in the Computer Science program.

Data collection and analysis

All CS faculty are involved in collecting data in their respective courses and reporting the results to the Assessment Coordinator. All materials, from assessment instruments to data files and reports, past and present, have been stored in a shared folder so all faculty members have access. In addition, the Assessment Coordinator sends copies of the instrument for specific courses, which may contain measurements for more than one learning outcome, to the faculty teaching the course. This is done at the start of the semester and after the mid-semester break to ensure that faculty remember to include the measurements in their teaching and testing materials.

If an instrument does not fit the course material, the instructor and assessment coordinator can make changes that don't substantively change the items. For instance, code fragments in Data Structures can be in C# in one course, and in Python in another depending on the language used.

After the semester, faculty email results to the assessment coordinator. Results are either in aggregate form (summary, averages) or anonymized (names replaced by numbers) to comply with the Family Educational Rights and Privacy Act (FERPA) (U.S. Department of Education, 2020). FERPA requires all student work to be maintained in a secure system with access limited to those involved in assessment, or personally identifiable information should be removed.

According to the assessment handbook, students should also be aware that their work may be used in assessment purposes. One way to do this is by incorporating a statement in select or all course syllabi. In the preparation for this article, we noticed that this was not common practice and reported this to Academic Affairs. As of this writing, it is not sure if a statement may be included in the campus-wide common syllabus template or not. If this is not the case, CS faculty have the option of including it in their individual syllabi.

As discussed before, each learning objective is measured two or three times in the program. Since students take courses in a predictable sequence due to prerequisites and course rotations, we can label them as introduced, reinforced, and mastered. The learning objectives are labeled as introduced in the first course where measured, mastered in the last course where measured, and reinforced if measured in between. This also allows us to set different performance targets based on placement in the sequence. Performance targets were arbitrarily set at 60% for introduced, 70% for reinforced, and 80% for mastered, to allow for improved mastery throughout the program and realistic program goal evaluation. Alternative goals, such as the % of students achieving at least 70% on an item (e.g Aasheim et al., 2007) do not allow for improvement along the program.

Reporting data
The assessment program is cyclical in nature and cycles with the academic year (Figure: **4**). Data are collected and analyzed in Fall and Spring semesters, the assessment report due September 1st is prepared at the end of Spring or in Summer, and the assessment plan is revised as needed at the start of the Fall Semester. Before submission by September 1st, CS faculty review the results, discuss potential changes to assessment instruments and the CS program, and submit the report to Academic Affairs.

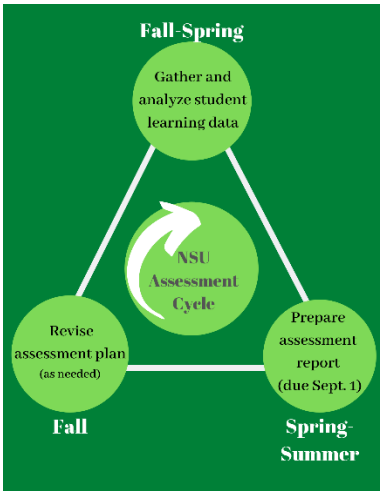


Figure: 4 Assessment cycle

The annual assessment report template, developed by the Office of Academic Affairs in collaboration with the Office of Institutional Effectiveness, is used across campus and includes the outcomes measures, the performance targets, the number of students assessed, the results, analysis of the results, and plans for the future (closing the loop).

4. FIRST YEAR'S RESULTS

Collecting and analyzing data for two semesters provided excellent results with regard to the current performance of our CS program. We will now describe the results for the eight different program outcomes. The specific measurement items are listed in Appendix D. Numbers of students and scores are listed separately for Fall and Spring.

Student communication
Our program goal "Communicate technical ideas both orally and in writing" was measured twice. In the ENGL3083 Technical writing course, students (n=5,13) scored well above the performance target of 60% with 92.5% and 82.5%. In the capstone oral presentation (n=14, 22) students scored initially below the performance target of 80% with 69.4%. After more thorough instruction in presenting skills during the capstone course, the scores improved to 85.5%

Computer programming
The second goal, "Program in two or more computer languages", was measured at three levels. At the introductory level, students (n=15, 8) surpassed the target of 60% with 92.4% and 92.9%. At the reinforced level, the goal was measured in three different courses in the fall (n= 11, 20, 16) and once in the spring surpassing the 70% target with 89.3%, 87.6%, 94.0%, and 96.4%. At the mastered level, with a target of 80%, students (n=18, 30) scored 89.4% and 94.6%.

Designing algorithms
The third goal, "Design computer algorithms to solve problems", was again measured at three levels. At the introductory level, students in the fall (n=17) scored 70.6% with a goal of 60% and in two spring sections (n=17,15) 55.3% and 62.6%. At the reinforced level, students (n= 15,8) fell short of the 70% mark with 33.0% and 40.6%. At the mastered level, data was only collected in the spring (n= 40, score = 92.0%, target 80%).

Applying algorithms
The fourth goal, "Apply computer algorithms to solve problems", was measured at three levels. At the introductory level, students (n = 15, 8) first fell short of the 60% target and then surpassed it with 72.5%. At the reinforced level, data was only collected in the spring (n = 40, score = 74.3%, target 70%). At the mastered level again, data collection in the spring (n= 30) fell short of the 80% target with a score of 60.7%.

Using software engineering principles
The fifth goal, "Create software using software engineering principles", was supposed to be measured at three levels. At the introductory level, students in the fall (n=17) fell short of the 60% target with 52.9% and in two spring sections (n=17,15) results were mixed with 50.6% and 66.7%. No data was collected at the reinforced level in either semester because the instructor in the course neglected to collect data. At the mastered level, the students in the fall (n=37) surpassed the 80% target with 87.8% and fell short in the spring (n=16) with 65.6%.

Collaboration
The sixth goal, "Collaborate as a member of a team" was measured at two levels. At the

introductory level, students (n=15,13) surpassed the 60% target with 92.0% and 93.0%. At the mastered level, using the same instrument and an 80% target, the results were mixed with 75.0% (n=37) and 82.2% (n=16).

Core elements of CS

The seventh goal, "Explain core elements of Computer Science", was measured at three levels. At the introductory level, students in the fall (n=45) surpassed the 60% target with 86.7%. At the reinforced level, two sections (n=23,19) had mixed results for a 70% target with scores of 87.0% and 44.0%. In the spring (n=19), results again fell short with 67.9%. At the mastered level, students (n=23,25) surpassed the 80% target with scores of 87.0% and 92.0%.

Problem solving

The final goal, "Solve problems by applying the software development process", was measured at three levels. At the introductory level, 17 students in the fall surpassed the 60% target with 64.7% and two sections in the spring (n=17,15) surpassed the target with 68.2% and 69.3%. At the reinforced level, students in both semesters (n=37,16) fell short of the 70% target with 65.4% and 65.6%. At the mastered level, one section in the fall (n=9) and three sections in the spring (n=12,5,13) surpassed the 80% target with 83.8%, 85.8%, 86.7%, and 85.3%.

5. CONCLUSIONS AND RECOMMENDATIONS

Redesigning our program assessment for university accreditation was an involved process, but it has already resulted in a better understanding of the program's strengths and weaknesses. Based on analysis of the data above, we conclude that our students have very good programming skills (goal 2), understand the core elements of CS (goal 7), could benefit from more work with algorithms (goals 3 and 4), and need more practice in professional presentations (goal 1). The process of data collection has also improved. In some instances, more data could still be collected. Only one of two instructors in ENGL 3083 submitted data, one CS faculty member may not have understood the process in the first semester and neglected to submit data in the second semester after having been released from the tenure track. Not all instructors require capstone presentations yet.

Assessment is an ongoing effort, and we plan to continue actively working within the assessment cycle to improve the quality of the CS program. One of the key determinants of successful continuing assessment efforts may be generating institutional support for changes in program structure and allocation of resources as identified by the assessment process. Based on this assessment cycle, we can already identify some potential program changes. We could merge CS 3173 Basic Computer Architecture and CS 3343 Computer Operating Systems to a single course to make room for an advanced programming course. We are also considering removing ENGL 3083 Professional and Technical Writing to make room for an advanced computer algorithms course. The writing component could be replaced by requiring submission of professional portfolios on a site like GitHub. Career Services could be included for development of resumes and interview skills. Finally, the data suggest that we need to come to a more consistent approach between faculty teaching the same course, especially in the capstone course.

Now that we have set up a basic framework for assessment, we can use this as a framework to address applying for program accreditation. Based on previous publications, the transition from one type of accreditation (institutional) to another (program) may be relatively straightforward, even if there are some differences (Babb & Abdullat, 2014; Feinstein et al., 2014; Hilton et al., 2004). Our CS program underwent a program review last Fall. Using the assessment results, the recommendations of the program review, and the 2020 ACM Curriculum for Computer Science (Association of Computing Machinery, 2020), we hope to revise the curriculum in preparation for program accreditation. Results of these efforts may be reported at a later date.

6. REFERENCES

Aasheim, C., Gowan, J. A., & Reichgelt, H. (2007). Establishing an assessment process for a computing program. *Information Systems Education Journal*, *5*(1), 23–25.

Al-Mubaid, H., Abeysekera, K., Kim, D., Perkins-Hall, S., & Yue, K.-B. (2011). A Model for Long Term Assessment of Computing and Information Systems Programs. *Information Systems Education Journal*, *9*(3), 59.

American Association of University Professors. (2015). *No Child Left Behind Goes to College*. AAUP. https://www.aaup.org/article/no-child-left-behind-goes-college

Anderson, L. W., Kratwohl, D. R., Airasian, P. W., Cruikshank, K. A., Mayer, R. E., Pintrich, P. R., Raths, J., & Wittrock, M. W. (2001). *A*

Taxonomy for Learning, Teaching, and Assessing—A revision of Bloom's Taxonomy of Educational Objectives. Longman.

Association of American Colleges & Universities. (2020, April 30). *Essential Learning Outcomes* [Text]. Association of American Colleges & Universities. https://www.aacu.org/essential-learning-outcomes

Association of Computing Machinery. (2020). *ACM and IEEE-CS issue Computing Curricula 2020*. https://www.acm.org/media-center/2021/march/computing-curricula-2020

Babb, J. S., & Abdullat, A. (2014). Communicating the Value of Program-Level Accreditation for Information Systems in the College of Business. *Information Systems Education Journal*, *12*(3), 39.

Banta, T. W. (2001). Assessing competence in higher education. In *Assessing student competence in accredited disciplines: Pioneering approaches to assessment in higher education* (pp. 1–12). Stylus Sterling, VA.

Blaich, C., & Wise, K. (2011). From gathering to using assessment results. *National Institute for Learning Outcomes Assessment*.

Bloom, B. S. (1956). Taxonomy of educational objectives. Vol. 1: Cognitive domain. *New York: McKay*, *20*, 24.

Boehner, J. A. (2002, January 8). *H.R.1 - 107th Congress (2001-2002): No Child Left Behind Act of 2001* (2001/2002). https://www.congress.gov/bill/107th-congress/house-bill/1

Challa, C. D., Kasper, G. M., & Redmond, R. (2004). The Accreditation Process for IS Programs in Business Schools. *Journal of Information Systems Education*, *16*(2), 207.

Eaton, J. S. (2015). An Overview of U.S. Accreditation. Revised November 2015. In *Council for Higher Education Accreditation*. Council for Higher Education Accreditation. https://eric.ed.gov/?id=ED569225

Educational Testing Service. (2021). *Major Field Tests: Computer Science*. https://www.ets.org/mft/about/content/computer_science

Educationdata.org. (2021). *Financial Aid Statistics*. EducationData. https://educationdata.org/financial-aid-statistics

Feinstein, D., Longenecker, B., & Shresthat, D. (2014). A Study of Information Systems Programs Accredited by ABET In Relation to IS 2010. *Information Systems Education Journal*, *12*(3), 76.

Fulcher, K. H., Good, M. R., Coleman, C. M., & Smith, K. L. (2014). A Simple Model for Learning Improvement: Weigh Pig, Feed Pig, Weigh Pig. Occasional Paper# 23. *National Institute for Learning Outcomes Assessment*.

Higher Learning Commission. (2021). *Criteria for Accreditation*. https://www.hlcommission.org/Policies/criteria-and-core-components.html

Hilton, T. S., Johnson, D. A., & Kasper, G. M. (2004). ABET Accreditation of MIS Programs in AACSB Schools. *The Proceedings of Information Systems Education Conference (ISECON)*, *21*, 1–16.

Hogan, T. R., Harrison, P. R., & Schulze, K. G. (2002). Developing and maintaining an effective assessment program. *ACM SIGCSE Bulletin*, *34*(4), 52–56.

Lending, D., Mitri, M., & Dillon, T. W. (2019). Ingredients of a high-quality information systems program in a changing IS landscape. *Journal of Information Systems Education*, *30*(4), 266.

Lumina Foundation. (2021). *Degree Qualifications Profile*. DQP. https://www.learningoutcomesassessment.org/dqp/

Murray, M. C., Perez, J., & Guimaraes, M. (2008). A model for using a capstone experience as one method of assessment of an information systems degree program. *Journal of Information Systems Education*, *19*(2), 197.

New England Commission of Higher Education. (2021). *Regional Accreditation Overview*. NECHE. https://www.neche.org/for-the-public/faq/regionally-accredited-overview/

Northeastern State University. (2017). *Institutional Assessment Plan*. https://offices.nsuok.edu/AcademicAssessment/Institutionalassessmentplan.aspx

Osters, S. (2003, February). *Writing measurable Leaning Outcomes Texas A&M 3rd Annual 2003 Assessment Conference, February. Retrieved May 25, 2021*. http://www.gavilan.edu/research/spd/Writin g-Measurable-Learning-Outcomes.pdf

Suskie, L. (2018). *Assessing Student Learning: A Common Sense Guide* (3rd edition). Jossey-Bass.

uPlanner. (2016, January 27). *Common challenges for accreditation in higher education institutions*. https://uplanner.com/en/blog/desafios-para-acreditar-programas-de-educacion-superior/

U.S. Department of Education. (2016, July 19). *Race to the Top Fund* [Program Home Page; Programs]. US Department of Education (ED). https://www2.ed.gov/programs/racetothetop/index.html

U.S. Department of Education. (2020, December 15). *Family Educational Rights and Privacy Act (FERPA)* [Guides]. US Department of Education (ED). https://www2.ed.gov/policy/gen/guid/fpco/ferpa/index.html

Veltri, N. F., Webb, H. W., Matveev, A. G., & Zapatero, E. G. (2011). Curriculum mapping as a tool for continuous improvement of IS curriculum. *Journal of Information Systems Education*, *22*(1), 31.

Washington Post. (2021, April 25). Washington pumped $35 billion into emergency grants for college students. Here's how it's going. *Washington Post*. https://www.washingtonpost.com/education/2021/04/24/college-emergency-aid-financial-assistance/

APPENDIX A
Listing of Institutional, Program, Course Learning Outcomes

Degrees of Excellence Institutional Learning Outcomes

1. Intellectual skills emphasizing analytic inquiry, information literacy, engaging diverse perspectives, quantitative fluency, and communication fluency.
2. Integrative knowledge emphasizing the ability to produce, independently or collaboratively, an investigative, creative, or practical work that draws on specific theories, evidence, tools, and methods from diverse perspectives.
3. Specialized knowledge in the major emphasizing student competency in the program outcomes of the major field(s) of study.
4. Capstone Experience in the Baccalaureate Degree emphasizing the integration of the major with baccalaureate degree expectations reflecting the intersection of academic and post-baccalaureate settings.
5. Citizenship emphasizing leadership and engagement, experiential learning, cultural foundations, and personal and career development.

General Education Learning Outcomes

1. Communicate effectively through writing, listening, speaking, and reading;
2. Recognize and analyze works in the humanities (literature, art, music, philosophy, and religion) as expressions used to communicate perspectives on the human condition;
3. Identify and evaluate political, historical, and social forces that shape the past, present, and future;
4. Become globally-aware citizens through an understanding and appreciation of human and cultural diversity;
5. Understand physical and biological phenomena and their importance for the welfare of society;
6. Apply methods of scientific inquiry;
7. Use quantitative symbolic systems to solve problems and interpret data;
8. Understand and apply concepts and activities that promote good health and life skills;
9. Use critical thinking to analyze and solve problems.

Computer Science Program Learning Outcomes

1. Communicate technical ideas both orally and in writing.
2. Apply current techniques, skills, and tools necessary in the computing practice (deleted).
3. Program in two or more computer languages.
4. Design computer algorithms to solve problems.
5. Apply computer algorithms to solve problems.
6. Create software using software engineering principles.
7. Collaborate as a member of a team.
8. Explain core elements of computer science theory.
9. Solve problems by applying the software development process.

APPENDIX B
Higher Learning Commission Criterion for Evaluation and Improvement of Teaching and Learning

Link: https://www.hlcommission.org/Policies/criteria-and-core-components.html

Criterion 4. Teaching and Learning: Evaluation and Improvement

The institution demonstrates responsibility for the quality of its educational programs, learning environments, and support services, and it evaluates their effectiveness for student learning through processes designed to promote continuous improvement.

Core Components

4.A. The institution ensures the quality of its educational offerings.

1. The institution maintains a practice of regular program reviews and acts upon the findings.
2. The institution evaluates all the credit that it transcripts, including what it awards for experiential learning or other forms of prior learning, or relies on the evaluation of responsible third parties.
3. The institution has policies that ensure the quality of the credit it accepts in transfer.
4. The institution maintains and exercises authority over the prerequisites for courses, rigor of courses, expectations for student learning, access to learning resources, and faculty qualifications for all its programs, including dual credit programs. It ensures that its dual credit courses or programs for high school students are equivalent in learning outcomes and levels of achievement to its higher education curriculum.
5. The institution maintains specialized accreditation for its programs as appropriate to its educational purposes.
6. The institution evaluates the success of its graduates. The institution ensures that the credentials it represents as preparation for advanced study or employment accomplish these purposes. For all programs, the institution looks to indicators it deems appropriate to its mission.

4.B. The institution engages in ongoing assessment of student learning as part of its commitment to the educational outcomes of its students.

1. The institution has effective processes for assessment of student learning and for achievement of learning goals in academic and cocurricular offerings.
2. The institution uses the information gained from assessment to improve student learning.
3. The institution's processes and methodologies to assess student learning reflect good practice, including the substantial participation of faculty, instructional and other relevant staff members.

4.C. The institution pursues educational improvement through goals and strategies that improve retention, persistence and completion rates in its degree and certificate programs.

1. The institution has defined goals for student retention, persistence and completion that are ambitious, attainable, and appropriate to its mission, student populations and educational offerings.
2. The institution collects and analyzes information on student retention, persistence, and completion of its programs.
3. The institution uses information on student retention, persistence, and completion of programs to make improvements as warranted by the data.
4. The institution's processes and methodologies for collecting and analyzing information on student retention, persistence and completion of programs reflect good practice. (Institutions are not required to use IPEDS definitions in their determination of persistence or completion rates. Institutions are encouraged to choose measures that are suitable to their student populations, but institutions are accountable for the validity of their measures.)

APPENDIX C
Initial Draft and Final Version of Course Mapping

Degree Program Courses or Experiences	Communicate technical ideas both orally and in writing.	Apply current techniques, skills, and tools necessary in the computing practice.	Program in two or more computer languages.	Design computer algorithms to solve problems.	Apply computer algorithms to solve problems.	Create software using software engineering principles.	Collaborate as a member of a team.	Explain core elements of computer science theory.	Solve problems by applying the software development process.	
CS 2014 Computer Science I		introduced	introduced		introduced					3
CS 2163 Computer Science II		reinforced	reinforced	introduced	reinforced					4
CS 3033 Obj Oriented Programming		reinforced	reinforced	reinforced	reinforced	introduced		introduced		6
CS 3173 Basic Computer Architecture								introduced		1
CS 3343 Computer Operating Systems								reinforced		1
CS 3403 Data Structures		reinforced	reinforced	mastered	mastered	introduced		reinforced		6
CS 4203 Software Engineering	reinforced	mastered				reinforced	introduced	reinforced	introduced	6
CS 4233 Professional Development in CS	mastered	mastered	mastered	mastered	mastered	mastered	reinforced		reinforced	8
CS 4343 Database Management Systems	reinforced	mastered	reinforced	reinforced	mastered					5
ENGL 3083 Professional & Technical Writing	introduced									1
MATH 3023 Discrete Mathematics			reinforced	reinforced						2
	4	7	7	6	6	4	2	5	2	
									items	43

Degree Program Courses or Experiences	Communicate technical ideas both orally and in writing.	Apply current techniques, skills, and tools necessary in the computing practice.	Program in two or more computer languages.	Design computer algorithms to solve problems.	Apply computer algorithms to solve problems.	Create software using software engineering principles.	Collaborate as a member of a team.	Explain core elements of computer science theory.	Solve problems by applying the software development process.	
CS 2014 Computer Science I				introduced		introduced			introduced	3
CS 2163 Computer Science II			introduced	reinforced	introduced					3
CS 3033 Obj Oriented Programming			mastered	mastered	reinforced					3
CS 3173 Basic Computer Architecture							introduced	reinforced		2
CS 3343 Computer Operating Systems								mastered		1
CS 3403 Data Structures					mastered					1
CS 4203 Software Engineering						mastered	mastered		reinforced	3
CS 4233 Professional Development in CS	mastered								mastered	2
CS 4343 Database Management Systems						reinforced				1
ENGL 3083 Professional & Technical Writing	introduced									1
MATH 3023 Discrete Mathematics								introduced		1
any other programming elective (optional)			reinforced							1
	2	0	3	3	3	3	2	3	3	
									items	22

APPENDIX D
Measurement Items

CS 2014: Computer Science I

Design computer algorithms to solve problems (introduced)

State the output for each of the following program segments. Assume that there are no syntax errors and that the appropriate header statements are there to make it compile correctly.

1.
```
for(int i=1; i< 3; i++)
    {
        cout << i << " ";
    }
```
a) 1, 2, 3
b) 1 3
c) 3
d) 1 2

2.
```
int myarray[3];
for(int i=0; i< 3; i++)
myarray[i] = i;
for(int i=0; i< 3; i++)
cout << myarray[i] << endl;
```
a) 0, 1, 2
b) 0 1 2
c) 0 1
   2
d) 0
   1
   2

3.
```
int i=1, total=0;
while(i < 4)
{
total=total+i;
i++;
}
cout << total << " " << i << endl;
```

i	total
1	0
2	1
3	3
4	6

a) 6 3
b) 6 4
c) 10
d) 4 6

4.
```
void dostuff(char a[],  char b[]);

void main(void)   {
  char s1[] = "PAPOA";
  char s2[] = "WEHLO";

  dostuff(s1, s2);
  cout << s1;
 }

void dostuff(char a[],  char b[]) {
  a[0] = 'Y';
```

```
  a[2] = b[2];
  a[4] = b[4];
 }
```

a) PAHAO
b) YAHOA
c) PAPOA
d) YAHOO

5.
```
int x=5, y=-1,  &rx=x,  &ry=y, z;
z = rx + ry;
    cout << x << ", " << ry << ", " << z;
```

a) NULL
b) 5 -1 4
c) 5 -1 NULL
d) 5, -1, 4

Create software using software engineering principles (introduced)

1.Which of the following is not a nested loop?

a)
```
for(i=0;i<10;i++)
    for(j=1;j<i+2;j++)
        cout << i*j << endl;
```

b)
```
for(i=0;i<10;i++)
        cout<<"i="<<i;
   for(j=1;j<i+2;j++) cout<<"j="<<j;
```

c)
```
for(i=0;i<10;i++)
        while(j%2!=0) { cout<<j<<"\t"; j++; }
```

d)
```
while(j<x)
{
    while(x<z)
        x+=j;
}
```

2.How many times will the loop print "NSU"?
```
#include <iostream>
   int x = 5;
  while (x = 5)
   {
      cout << "NSU\n" << endl;
      ++x;
   }
```
a) infinite loop
b) 4
c) 5
d) 0

3. Convert the While loop to a Do While loop
```
while (number <1 || number >10)
{
 a)      Do
     {
```

```
        cout << "enter a number between 1 and 10: ";
    cin >> number;
    }while (number <1 || number >10);
```

b)
```
Do;
{
        cout << "enter a number between 1 and 10: ";
cin >> number;
}while (number <1 || number >10)
```

c)
```
Do
{
        cout << "enter a number between 1 and 10: ";
cin >> number;
}while (number <1 && number >10);
```

d)
```
Do
{
        cout << "enter a number between 1 and 10: ";
cin >> number;
}while (number >1 || number >10);
```

4. Convert the While loop to a For loop

```
while(counter <= 5)
{
   cout << (startNumber + counter) << " squared is "
 << pow((startNumber + counter),2) << endl;
   counter++;
}
```

a)
```
for(int counter = 1; counter <=5; counter++)
{
cout << (startNumber + counter) << " squared is "
<< pow((startNumber + counter),2) << endl;

}
```

b)
```
for(int counter = 1, counter <=5, counter++)
{
cout << (startNumber + counter) << " squared is "
<< pow((startNumber + counter),2) << endl;

}
```

c)
```
{
cout << (startNumber + counter) << " squared is "
<< pow((startNumber + counter),2) << endl;

} for(int counter = 1; counter <=5; counter++)
```

d)
```
{
cout << (startNumber + counter) << " squared is "
<< pow((startNumber + counter),2) << endl;
} for(int counter = 1, counter <=5, counter++)
```

5. Which code will create a random roll of the die?

a)
```
#include <cstdlib>
#include <ctime>
```

```
……..
const int MIN_VALUE = 1;
const int MAX_VALUE = 6;
unsigned seed = time(0);
srand(seed);
cout << "Rolling the dice...\n";
int   die = (rand() % (MAX_VALUE - MIN_VALUE + 1)) + MIN_VALUE;
cout << die << endl;
```

b)
```
#include <cstdlib>
#include <ctime>
……..
const int MIN_VALUE = 1;
const int MAX_VALUE = 6;
unsigned seed = time(0);
srand(seed);
cout << "Rolling the dice...\n";
int   die = (rand() % (MAX_VALUE - MIN_VALUE) + MIN_VALUE;
cout << die << endl;
```

c)
```
#include <cstdlib>
#include <ctime>
……..
const int MIN_VALUE = 1;
const int MAX_VALUE = 6;
cout << "Rolling the dice...\n";
int   die = (rand() % (MAX_VALUE - MIN_VALUE+1));
cout << die << endl;
```

d)
```
#include <cstdlib>
#include <ctime>
……..
const int MIN_VALUE = 1;
const int MAX_VALUE = 6;
unsigned seed = time(0);
random(seed);
cout << "Rolling the dice...\n";
int   die = (rand() % (MAX_VALUE - MIN_VALUE + 1)) + MIN_VALUE;
cout << die << endl;
```

Solve problems by applying the software development process (introduced)

1. Put the steps of software development in the proper order:

a. Develop algorithms in pseudocode
b. Gathering requirements
c. Testing
d. Write programming code

a) b - a - d - c
b) a - b - c - d
c) c - d - a - b
d) d - a - c - b

2. How can we prevent the input problem in this program?

```
#include <iostream>
using namespace std;
int main()
```

```
{
  int year;
  char ch;
  cout << "Enter the year you were born: ";
  cin >> year;
  cout << "Enter your middle initial: ";
  ch = cin.get();
  cout << ch << " was born in " << year << "\n";
  return 0;
}
/* output:
Enter the year you were born: 1988
Enter your middle initial:
 was born in 1988
Press any key to continue . . .
*/
```

a) add cin.ignore() after the input line for the year
b) change ch = cin.get() to cin >> ch
c) change cin >> year to year = cin.get()
d) change int year to char[4] year

3. Put the steps to create an executable program in the proper order:
   a. run the compiler to convert source code to machine instructions
   b. run the preprocessor to convert source file directives to source code program statements
   c. run the linker to connect hardware-specific code to machine instructions
   d. create a source code file with a text editor

   a) d - b - a - c
   b) a - b - c - d
   c) d - a - c - b
   d) b - c - d - a

4. Which loop will create the following output: 20 17 14 11 8 5

   a)
```
for (int x = 20; x >= 5; x -= 3)
      cout << x << " ";
```

   b)
```
for (int x = 20; x > 5; x -= 3)
      cout << x << " ";
```

   c)
```
for (int x = 20; x < 5; x -= 3)
      cout << x << " ";
```

   d)
```
for (int x = 20; x <= 5; x -= 3)
     cout << x << " ";
```

5. How can we BEST improve the following program?

```
#include <iostream>
#include <cstdlib>
#include <ctime>
using namespace std;

int main()
{
  const int MIN_VALUE = 1;
  const int MAX_VALUE = 6;

  unsigned seed = time(0);
  srand(seed);
```

```
  int die;

  cout << "Rolling the dice...\n";
  die = (rand() % (MAX_VALUE - MIN_VALUE + 1)) + MIN_VALUE;
  cout << die << endl;
  cout << "Rolling the dice...\n";
  die = (rand() % (MAX_VALUE - MIN_VALUE + 1)) + MIN_VALUE;
  cout << die << endl;
  cout << "Rolling the dice...\n";
  die = (rand() % (MAX_VALUE - MIN_VALUE + 1)) + MIN_VALUE;
  cout << die << endl;
  cout << "Rolling the dice...\n";
  die = (rand() % (MAX_VALUE - MIN_VALUE + 1)) + MIN_VALUE;
  cout << die << endl;
  cout << "Rolling the dice...\n";
  die = (rand() % (MAX_VALUE - MIN_VALUE + 1)) + MIN_VALUE;
  cout << die << endl;

  return 0;
}
```

a) create a function
b) change the constants to regular integers
c) use global variables
d) add input checking
e) make die a constant

CS 2163: Computer Science II

Program in two or more computer languages (introduced)

Programmer(s)

Program Assessment
3 = Good 2 = Average 1 = Below Average 0 = Not gradable

Criteria	Comments	Points			
The program fully implements the solution to the given problem		3	2	1	0
The solution produces the correct result		3	2	1	0
The program runs without error, has reasonable error checking and exception handling		3	2	1	0
The user is prompted for input when required		3	2	1	0
The user is presented with appropriate feedback as to the results of an operation		3	2	1	0
All project files and source code required to build and execute are submitted		3	2	1	0
Code is properly indented, and adequately commented		3	2	1	0

Code is well written and free from unnecessary complexity or redundancy		3	2	1	0
Programmer defined symbol names are reasonable (variable names, class names, function names, etc...)		3	2	1	0
The user interface is accurate and is free from misspelled words and bad grammar		3	2	1	0
Total Points out of 30:					
Comments:					

Design computer algorithms to solve problems (reinforced)

1. State output for the following:

```
int fun3(int a[],  int b[], int cnt);
int main(void)   {
   const int TOP = 3;
   int x[TOP] = {5, 3,  2};
   int y[TOP] = {1, 4,  3};
   int val;

   val = fun3(x, y, TOP);
   for(int i=0; i<TOP; i++)
      cout << x[i] << ",  ";
   cout << endl << val;
 }
int fun3(int a[],  int b[], int cnt)
{
   int val=0;
   for(int i=0; i<cnt; i++)
   {
     if(b[i] > a[i])
       a[i] = b[i];
     val += a[i];
   }
   return val;
 }
```

A. 5, 4, 3, 12
B. 5, 4, 3,
   12
C. 5, 4,
   3, 12
D. 5, 3, 2, 1, 4, 3

2. Which of the following code will correctly begin a linked list to hold a double?

A.

```
struct ListNode
  {
```

```
    double value;
    struct ListNode *next;
  };
  ListNode head;
```

B.

```
struct ListNode
  {
    double value;
    struct ListNode *next;
  };
  ListNode *head;
```

C.

```
struct ListNode
{
   double value;
   struct ListNode next;
};
ListNode *head;
```

D. None of these answers are correct

3. Which of the following code will correctly append an element into a linked list?

A.

```
void NumberList::appendNode(double num)
 {
    ListNode newNode;
    ListNode nodePtr;
    newNode = new ListNode;
    newNode->value = num;
    newNode->next = NULL;
    if (!head)
      head = *newNode;
    else    {
      nodePtr = head;
      while (nodePtr->next)
        nodePtr = nodePtr->next;
      nodePtr->next = newNode;
      }
    }
```

B.

```
void NumberList::appendNode(double num)
 {
    ListNode *newNode;
    ListNode *nodePtr;
    newNode = new ListNode;
    newNode->value = num;
    newNode->next = NULL;
    if (!head)
      head = newNode;
    else    {
      nodePtr = head;
      while (**nodePtr)
        nodePtr = *next;
      **next = *newNode;
      }
```

```
  }
```

C.
```
void NumberList::appendNode(double num)
{
  ListNode *newNode;
  ListNode *nodePtr;
  newNode = new ListNode;
  newNode->value = num;
  newNode->next = NULL;
  if (!head)
    head = newNode;
  else    {
    nodePtr = head;
    while (nodePtr->next)
      nodePtr = nodePtr->next;
    nodePtr->next = newNode;
    }
  }
```

D. None of these answers is correct

4. Given the following output:

```
enter an integer  10
that value doubled is 20
```

Which of the following programs below produces the above output?

A.
```
void getNumber(int *input)
  {
        cout << "enter an integer " ;
        cin >> *input;
  }
  void doubleValue(int *val)
  {
        *val *= 2;
  }

  void main()
  {
        int number = 0;
        getNumber(&number);
        doublevalue(&number);
        cout << "that value doubled is " << number << endl;

  }
```

B.
```
void getNumber(int *input)
  {
        cout << "enter an integer " ;
        cin >> input;
  }
  void doubleValue(int *val)
  {
        val *= 2;
  }

  void main()
```

```
{
        int number = 0;
        getNumber(&number);
        doublevalue(&number);
        cout << "that value doubled is " << number << endl;

}
```

C.
```
void getNumber(int input)
{
        cout << "enter an integer " ;
        cin >> input;
}
void doubleValue(int val)
{
        val *= 2;
}

void main()
{
        int number = 0;
        getNumber(number);
        cout << "that value doubled is " << number << endl;

}
```

D. None of these answers are correct.

Apply computer algorithms to solve problems (introduced)

1.Which algorithm is shown below?

```
int func(const int array[], int size, int value)
{
     int index = 0;
     int position = -1;
     bool found = false;

     while (index < size && !found)
     {
        if (array[index]  == value)
        {
           found = true;
           position = index;
        }
        index++;
     }
    return position;
}
```

a) linear search
b) bubble sort
c) binary search
d) selection sort

2. which algorithm is shown below?

```
void func(int array[], int size)
```

```
{
  bool swap;
  int temp;

  do
  {
    swap = false;
    for (int count = 0; count < (size - 1); count++)
    {
      if (array[count] > array[count + 1])
      {
        temp = array[count];
        array[count] = array[count + 1];
        array[count + 1] = temp;
        swap = true;
      }
    }
  } while (swap);
}
```

a) bubble sort
b) binary search
c) selection sort
d) linear search

3. Which algorithm is shown below?

```
void func(int array[], int size)
{
  int startScan, minIndex, minValue;

  for (startScan = 0; startScan < (size - 1); startScan++)
  {
    minIndex = startScan;
    minValue = array[startScan];
    for(int index = startScan + 1; index < size; index++)
    {
      if (array[index] < minValue)
      {
        minValue = array[index];
        minIndex = index;
      }
    }
    array[minIndex] = array[startScan];
    array[startScan] = minValue;
  }
}
```

a) selection sort
b) binary search
c) bubble sort
d) linear search

4. Which algorithm is shown below?

```
int func(const int array[], int size, int value)
{
   int first = 0,
         last = size - 1,
         middle ,
         position = -1;
   bool found = false;
```

```
    while (!found && first <= last)
    {
       middle = (first + last ) / 2;
       if (array[middle]  == value)
       {
          found = true;
          position = middle;
       }
       else if (array[middle] > value)
          last = middle -1;
       else
          first = middle +1;
    }
    return position;
}
```

a) binary search
b) selection sort
c) bubble sort
d) linear search

5. Which algorithm is shown below?

```
int func(int arr[], int l, int r, int x)
{
   if (r >= l) {
      int mid = l + (r - l) / 2;
      if (arr[mid] == x)
         return mid;
      if (arr[mid] > x)
         return func(arr, l, mid - 1, x);
      return func(arr, mid + 1, r, x);
   }
   return -1;
}
```

a) binary search
b) selection sort
c) bubble sort
d) linear search

CS 3033: Object Oriented Programming

Program in two or more computer languages (mastered)

Programmer(s)

Program Assessment
3 = Good 2 = Average 1 = Below Average 0 = Not gradable

Criteria	Comments	Points			
The program fully implements the solution to the given problem		3	2	1	0
The solution produces the correct result		3	2	1	0
The program runs without error, has reasonable error checking and exception handling		3	2	1	0
The user is prompted for input when required		3	2	1	0

The user is presented with appropriate feedback as to the results of an operation		3	2	1	0
All project files and source code required to build and execute are submitted		3	2	1	0
Code is properly indented, and adequately commented		3	2	1	0
Code is well written and free from unnecessary complexity or redundancy		3	2	1	0
Programmer defined symbol names are reasonable (variable names, class names, function names, etc...)		3	2	1	0
The user interface is accurate and is free from misspelled words and bad grammar		3	2	1	0
Total Points out of 30:					
Comments:					

Design computer algorithms to solve problems (mastered)

1. Consider the program below:

```
public class Test
{
  public static void main(String[] args)
  {
    int[] a;
    a = new int[10];

    for (int i = 0; i < a.length; i++)
      a[i] = i + 2;
    int result = 0;
    for (int i = 0; i < a.length; i++)
      result += a[i];
    System.out.printf("Result is: %d%n", result);
  }
}
```

The output of this program will be:

a) Result is: 62.
b) Result is: 64.
c) Result is: 65.
d) Result is: 67.

2. Assume class Book has been declared. Which set of statements creates an array of Books?

a.
```
Book[] books;
books = new Book[numberElements];
```
b.
```
Book[] books];
books = new Book()[numberElements];
```
c.
```
new Book() books[];
books = new Book[numberElements];
```
d. All of the above.

3. Which of the following class members should usually be private?
   a) Methods.
   b) Constructors.
   c) Variables (or fields).
   d) All of the above.

4. Static class variables:
   a) are final.
   b) are public.
   c) are private.
   d) are shared by all objects of a class.

5. Which of the following statement displays Hello World?
   a) System.out.printf("%2s", "Hello " "World");
   b) System.out.printf("%s %s", "Hello", "World");
   c) System.out.printf("%s%s", "Hello, World");
   d) System.out.printf("s% s%", "Hello", "World");

6. Which of the following statements is true?
   a) Local variables are automatically initialized.
   b) Every instance variable has a default initial value—a value provided by Java when you do not specify the instance variable's initial value.
   c) The default value for an instance variable of type String is void.
   d) The argument types in the method call must be identical to the types of the corresponding parameters in the method's declaration.

7. When must a program explicitly use the this reference?
   a) Accessing a private variable.
   b) Accessing a public variable.
   c) Accessing a local variable.
   d) Accessing an instance variable that is shadowed by a local variable.

8. A constructor *cannot*:
   a) be overloaded.
   b) initialize variables to their defaults.
   c) specify return types or return values.
   d) have the same name as the class.

9. Which superclass members are inherited by all subclasses of that superclass?
   a) private instance variables and methods.
   b) protected instance variables and methods.
   c) private constructors.
   d) protected constructors.

10. Overriding a method differs from overloading a method because:
   a) Overloaded methods have the same signature.
   b) Overridden methods have the same signature.
   c) Overridden methods must have different signatures
   d) Neither of the above.

Apply computer algorithms to solve problems (reinforced)

1.Which algorithm is shown below?

```
public static int func(int arr[], int x)
```

```
{
   int n = arr.length;
   for(int i = 0; i < n; i++)
   {
      if(arr[i] == x)
         return i;
   }
   return -1;
 }
```

a) linear search
b) binary search
c) interpolation search
d) selection sort
e) bubble sort
f) merge sort
g) quick sort

2.Which algorithm is shown below?

```
public static  int func(int arr[], int l, int r, int x)
   {
      if (r >= l) {
         int mid = l + (r - l) / 2;
         if (arr[mid] == x)
            return mid;
         if (arr[mid] > x)
            return func(arr, l, mid - 1, x);
         return func(arr, mid + 1, r, x);
      }
      return -1;
   }
```

a) linear search
b) binary search
c) interpolation search
d) selection sort
e) bubble sort
f) merge sort
g) quick sort

3.Which algorithm is shown below?

```
static int func(int x)
   {
      int lo = 0, hi = (arr.length - 1);
      while (lo <= hi && x >= arr[lo] && x <= arr[hi])
      {
         if (lo == hi)
         {
            if (arr[lo] == x) return lo;
            return -1;
         }
         int pos = lo + (((hi-lo) /
               (arr[hi]-arr[lo]))*(x - arr[lo]));
         if (arr[pos] == x)
            return pos;
         if (arr[pos] < x)
            lo = pos + 1;
         else
            hi = pos - 1;
```

```
            }
            return -1;
        }
```

a) linear search
b) binary search
c) interpolation search
d) selection sort
e) bubble sort
f) merge sort
g) quick sort

4.Which algorithm is shown below?

```
void func(int arr[])
{
   int n = arr.length;

   for (int i = 0; i < n-1; i++)
   {
      int min_idx = i;
      for (int j = i+1; j < n; j++)
         if (arr[j] < arr[min_idx])
            min_idx = j;
      int temp = arr[min_idx];
      arr[min_idx] = arr[i];
      arr[i] = temp;
   }
}
```

a) linear search
b) binary search
c) interpolation search
d) selection sort
e) bubble sort
f) merge sort
g) quick sort

5.Which algorithm is shown below?

```
void func(int arr[])
{
   int n = arr.length;
   for (int i = 0; i < n-1; i++)
      for (int j = 0; j < n-i-1; j++)
         if (arr[j] > arr[j+1])
         {
            int temp = arr[j];
            arr[j] = arr[j+1];
            arr[j+1] = temp;
         }
}
```

a) linear search
b) binary search
c) interpolation search
d) selection sort
e) bubble sort
f) merge sort
g) quick sort

6.Which algorithm is shown below?

```
void func1(int arr[], int l, int m, int r)
{
   int n1 = m - l + 1;
   int n2 = r - m;
   int L[] = new int[n1];
   int R[] = new int[n2];
   for (int i = 0; i < n1; ++i)
      L[i] = arr[l + i];
   for (int j = 0; j < n2; ++j)
      R[j] = arr[m + 1 + j];
   int i = 0, j = 0;
   int k = l;
   while (i < n1 && j < n2) {
      if (L[i] <= R[j]) {
         arr[k] = L[i];
         i++;
      }
      else {
         arr[k] = R[j];
         j++;
      }
      k++;
   }

   while (i < n1) {
      arr[k] = L[i];
      i++;
      k++;
   }
   while (j < n2) {
      arr[k] = R[j];
      j++;
      k++;
   }
}

void func2(int arr[], int l, int r)
{
   if (l < r) {
      int m = (l + r) / 2;
      func2(arr, l, m);
      func2(arr, m + 1, r);
      func1(arr, l, m, r);
   }
}
```

a) linear search
b) binary search
c) interpolation search
d) selection sort
e) bubble sort
f) merge sort
g) quick sort

7.Which algorithm is shown below?

```
int func1(int arr[], int low, int high)
{
   int pivot = arr[high];
```

```
        int i = (low-1);
        for (int j=low; j<high; j++)
        {
            if (arr[j] < pivot)
            {
                i++;
                int temp = arr[i];
                arr[i] = arr[j];
                arr[j] = temp;
            }
        }

        int temp = arr[i+1];
        arr[i+1] = arr[high];
        arr[high] = temp;
        return i+1;
    }
    void func2(int arr[], int low, int high)
    {
        if (low < high)
        {
            int pi = func1(arr, low, high);
            func2(arr, low, pi-1);
            func2(arr, pi+1, high);
        }
    }
```

a) linear search
b) binary search
c) interpolation search
d) selection sort
e) bubble sort
f) merge sort
g) quick sort

CS 3173: Basic Computer Architecture

Collaborate as a member of a team (introduced)

Do some class activities as groups. For instance, breakout rooms in Zoom or discussing a list of potential test questions in a group. At the end of the semester, include the following questions as part of final exam. Students will score themselves (self-evaluation). Answers will not be scored for correctness, but they will need to answer all questions to get the points (10% of final exam score).

Work-Related Interactions with Others (self-evaluation)
Score yourself on your participation in the breakout rooms/ group discussions during the course.
There are no right or wrong answers, but you do have to answer all questions.

a. Collaboration: How well did you cooperatively work with others? (1 poor – 2 fair – 3 average – 4 good – 5 excellent)
b. Participation: How well did you contribute your "fair share" to the tasks? (1 poor – 2 fair – 3 average – 4 good – 5 excellent)
c. Attitude1: How positive was your approach doing the tasks? (1 poor – 2 fair – 3 average – 4 good – 5 excellent)
d. Attitude2: How well did you make constructive comments? (1 poor – 2 fair – 3 average – 4 good – 5 excellent)
e. Communication: How clearly did you express your thoughts? (1 poor – 2 fair – 3 average – 4 good – 5 excellent)

f. Responsiveness: How sensitively did you respond to verbal and nonverbal cues of other team members? (1 poor – 2 fair – 3 average – 4 good – 5 excellent)

- Future (essay): What single thing will you try to do better in the future when working in groups?

Explain core elements of computer science theory (reinforced)

1. If a digital thermometer measures between - 100 and +300 degrees, and the digital readout is 0110 1100 (8 bit), what is the analog value being measured?
   a) 127.11
   b) 84.71
   c) 69.41
   d) 172.95

2. What is the hexadecimal value for a binary unsigned integer of 0011 1001?
   a) 57
   b) 39
   c) 1100 0111
   d) does not exist

3. What is the binary two's complement for 1011 1011?
   a) 0100 0101
   b) 0100 0100
   c) BB
   d) does not exist

4 The binary two's complement 1011 0111 has a decimal value of:
   a) 183
   b) 73
   c) -37
   d) -73

5. An 8-bit value has a minimum of -1,023 and a maximum of + 1,023. The number is a(n)
   a) binary coded decimal
   b) unsigned integer
   c) two's complement
   d) signed integer

6. The 32-bit binary float of 1100 1000 0101 1111 0101 1100 1001 0000 has a decimal value of:
   a) -1.8*2^-17
   b) +1.8*2^-17
   c) -1.8*2^17
   d) +1.8*2^17

7. The subnet for the IP address 108.235.251.35/22 ranges from:
   a) 108.235.251.1 - 108.235.251.254
   b) 108.235.248.1 - 108.235.251.254
   c) 108.235.251.1 - 108.235.252.254
   d) 108.235.251.1 - 108.235.254.254

8. The datastream 10110101000 is converted to a Hamming code and arrives as 001101000101000. Which bit of the original data stream was corrupted?
   a) the sixth bit
   b) the fifth bit
   c) the fourth bit
   d) there was no corruption

9. What is the chip select for a memory device with a low hexadecimal address of D8000 and a high hexadecimal address of DBFFF?
   a) 11011011

b) 1101
c) 1101101
d) 110110

52. What best describes the difference between Dynamic RAM (DRAM) and Static RAM (SRAM)?
a) SRAM is faster and cheaper
b) SRAM is faster and DRAM is cheaper
c) DRAM is faster and SRAM is cheaper
d) DRAM is faster and cheaper

CS 3343 Computer Operating Systems

Explain core elements of computer science theory (mastered)

1. _____ involves moving all or part of a process from main memory to disk.
a) Swapping
b) Relocating
c) Suspending
d) Blocking

2. The portion of the operating system that selects the next process to run is called the _____
a) program status word
b) trace
c) process control block
d) dispatcher

3. A situation in which a runnable process is overlooked indefinitely by the scheduler, although it can proceed, is _____
a) mutual exclusion
b) starvation
c) deadlock
d) livelock

4. The requirement that when one process is in a critical section that accesses shared resources, no other process may be in a critical section that accesses any of those shared resources is _____ .
a) atomic operation
b) critical section
c) livelock
d) mutual exclusion

5. A _____ is an integer value used for signaling among processes.
a) message
b) mutex
c) semaphore
d) atomic operation

6. The _____ condition can be prevented by requiring that a process request all its required resources at one time and blocking the process until all requests can be granted simultaneously.
a) no preemption
b) mutual exclusion
c) circular wait
d) hold and wait

7. A _____ is issued if a desired page is not in main memory.
a) page fault
b) paging error

c) page re-placement policy
d) page placement policy

8. Which of the following scheduling policies allow the O/S to interrupt the currently running process and move it to the Ready state?
a) Non-preemptive
b) Preemptive
c) First In First Out
d) First Come First Serve

9. _____ is a technique that smoothes out peaks in I/O demand.
a) Buffering
b) Blocking
c) Smoothing
d) Tracking

10. The most common form of file structure is _____
a) the pile
b) the indexed file
c) the sequential file
d) the indexed sequential file

CS 3403: Data Structures

Apply computer algorithms to solve problems (mastered)

1. Which of the below given sorting techniques has the highest best-case runtime complexity?
a) Quick Sort
b) Selection Sort
c) Insertion Sort
d) Bubble Sort

2 The time required to merge two sorted lists of size m and n, is
a) O(m + n)
b) O(m | n)
c) O(m log n)
d) O(n log m)

3. What is the worst case time complexity of the linear search algorithm?
a) O(n^2)
b) O(1)
c) O(log n)
d) O(n)

4. Which of the following searching techniques do not require the data to be in sorted form?
a) Interpolation Search
b) Binary Search
c) Linear Search
d) All the other answers

5. On which principle does the stack work?
a) Shortest Process Next
b) First Come First Serve
c) First In First Out
d) First In Last Out

6. In AVL trees, the fastest operation is _____.

a) insertion
b) deletion
c) retrieval
d) updating

7. Which of the following linked lists below have the last node of the list pointing to the first node?
a) circular linked list
b) circular doubly linked list
c) circular singly linked list
d) doubly linked list

8. In a heap, the element with the greatest key is always in the _____
a) first node of right sub tree
b) leaf node
c) first node of left sub tree
d) root node

9. Which of the following ways is a pre-order traversal?
a) Root->left sub tree-> right sub tree
b) Root->right sub tree-> left sub tree
c) right sub tree-> left sub tree->Root
d) left sub tree-> right sub tree->Root

10. A full binary tree with n leaves contains _____ nodes.
a) N
b) n^ 2
c) Log n^2
d) 2n –1

11. What is the average time complexity of the heap sort?
a) O(n^2)
b) O(n log n)
c) O(log n)
d) O(n)

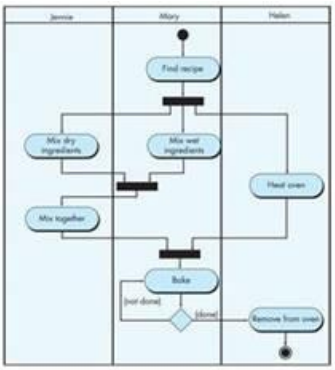
CS 4203: Software Engineering

Create software using software engineering principles (mastered)

1. Does software suffer from "wearing out"? Give the best answer
a) yes, instructions deteriorate over time
b) yes, due to changing requirements
c) yes, but you can just reinstall it
d) no, but we do have spikes in software failures when new features are introduced

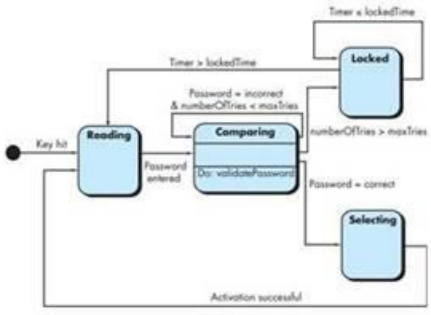
2. How is most software built?
a) custom-built
b) constructed from components
c) by combining different software packages
d) from bits and bytes

3. This picture shows the _____ (model)
   a) waterfall model
   b) incremental model
   c) spiral model
   d) concurrent model



4. This picture shows a(n) _____ (diagram)
   a) activity diagram
   b) activity diagram with swim lanes
   c) state diagram
   d) use-case diagram

5. What is the most important rule for use case diagrams?
   a) for the use case circle, use a verb and noun(s)
   b) only people are external to the system
   c) each use case is connected to at least two external entities
   d) each external entity is connected to at least two use cases



6. Which type of diagram is shown in this picture?
   a) state diagram
   b) activity diagram
   c) class diagram
   d) sequence diagram

7. What are the two major ways to PERMANENTLY store data?
    a) flat files and databases
    b) variables and objects
    c) arrays and lists
    d) CDROM and DVD

8. If you refactor code, and you try to reduce redundancy, how can you do that?
    a) creating functions that can be called repeatedly from multiple places
    b) use global variables
    c) use constants
    d) include more libraries

9. If you refactor code, and you want to remove unused design elements, how can you typically spot them?
    a) they are underlined with a green squiggly line
    b) they are underlined with a red squiggly line
    c) debugging stops and the line with the unused element is highlighted red
    d) use Debug / Refactoring / Redundancy

10. In cohesion and coupling, we want .....
    a) tight cohesion and loose coupling
    b) tight cohesion and tight coupling
    c) loose cohesion and loose coupling
    d) loose cohesion and tight coupling

Collaborate as a member of a team (mastered)

Do some class activities as groups. For instance, breakout rooms in Zoom or discussing a list of potential questions on the test. At the end of the semester,l include the following questions as part of final exam. Students will score themselves (self-evaluation). Answers will not be scored for correctness, but they will need to answer all questions to get the points (10% of final exam score).

Work-Related Interactions with Others (self-evaluation)
Score yourself on your participation in the breakout rooms/ group discussions during the course. There are no right or wrong answers, but you do have to answer all questions.

a. Collaboration: How well did you cooperatively work with others? (1 poor – 2 fair – 3 average – 4 good – 5 excellent)
b. Participation: How well did you contribute your "fair share" to the tasks? (1 poor – 2 fair – 3 average – 4 good – 5 excellent)
c. Attitude1: How positive was your approach doing the tasks? (1 poor – 2 fair – 3 average – 4 good – 5 excellent)
d. Attitude2: How well did you make constructive comments? (1 poor – 2 fair – 3 average – 4 good – 5 excellent)
e. Communication: How clearly did you express your thoughts? (1 poor – 2 fair – 3 average – 4 good – 5 excellent)
f. Responsiveness: How sensitively did you respond to verbal and nonverbal cues of other team members? (1 poor – 2 fair – 3 average – 4 good – 5 excellent)

- Future (essay): What single thing will you try to do better in the future when working in groups?

Solve problems by applying the software development process (reinforced)

1. The best strategy to follow when developing software that consists of many modules is to?
    a) Test each code module separately then link all modules together and test the software as a whole
    b) Combine unit and integration testing
    c) Test the software product only once, at the end of the software development
    d) All of the above

2. A high level of cohesion is an indication that a module?
   a) Can be written more compactly
   b) Focuses on just one function
   c) Interacts with many modules
   d) Can perform variety of functions

3. Which of the following are your best choice for developing Test cases derived from "Equivalence classes and boundary value analysis." for the following scenario: The daily discount must be in the range between and including $0.00 and $100.00.
   a) Testing the values: $0.00, $11.01, $59.31, $82.02, $100.00
   b) Testing the values: -$1.00, $0.00, $1.00, $50.00, $99.00, $100.00, $101.00
   c) Testing the values: $0.00, $11.21, $21.11, 28.82, $82.28, $100.00, $000.01
   d) Random Testing of at least 10% of the values in the range between and including $0.00 and $100.00

4. All else being equal, which is more desirable?
   a) Higher cohesion and higher coupling
   b) Higher cohesion and lower coupling
   c) Lower cohesion and lower coupling
   d) Lower cohesion and higher coupling

5. More time and money are spent in the _____ phase than any other phase.
   a) Requirements
   b) Analysis
   c) Maintenance
   d) Implementation

CS 4233: Professional Development in CS

Communicate technical ideas both orally and in writing (mastered)

	1(Unacceptable)	**2 (Marginal)**	**3 (Good)**	**4 (Excellent)**	**Score**
Body Language	No movement or descriptive gestures.	Very little movement or descriptive gestures.	Movements or gestures enhance articulation.	Movements seemed fluid and helped the audience visualize.	
Eye Contact	No eye contact with audience.	Minimal eye contact with audience.	Consistent use of direct eye contact with some audience.	Holds attention of entire audience with the use of direct eye contact.	
Speaking Skills	▪ inaudible or too loud ▪ rate too slow/fast ▪ speaker seemed uninterested and used monotone voice	▪ some mumbling ▪ uneven rate ▪ little or no expression	Clear articulation but not as polished	▪ Poised, clear articulation ▪ proper volume ▪ steady rate ▪ good posture ▪ enthusiasm ▪ confidence	
Organization	▪ displays neither clear	▪ displays some level of	▪ displays introductory or closing	▪ delivers clear opening and closing	

	introductory nor closing remarks ▪ does not present the segments of the body of the presentation in a coherent manner ▪ irrelevant statements are made ▪ leaves the audience wondering where the presentation is headed.	organization with discernible theme, but the presentation is not organized clearly or in a coherent manner. ▪ introductory and closing remarks are missing.	remarks, but segments of the body of the presentation are not presented in a coherent manner. ▪ presents the segments of the body of the presentation in a coherent manner, but introductory or closing remarks are missing.	remarks that capture the attention of the audience and set the mood ▪ provides a "road map" for the audience ▪ each segment relates to the others according to a carefully planned framework	
Voice	Consistently uses a monotone voice	Displays some level of inflection throughout delivery.	Satisfactory use of inflection, but does not consistently use fluid speech.	Use of fluid speech and inflection maintains the interest of the audience.	
Visuals	Used no visuals.	▪ ran too quickly through visuals and spoke more to the screen than to the audience ▪ visuals did not detract from the presentation.	▪ gave audience almost enough time to absorb material, but occasionally read the slide ▪ visuals added to the presentation.	▪ gave audience ample time to absorb information on visual ▪ spoke to the audience, not the screen ▪ visuals greatly enhanced presentation.	
Language	Multiple grammar errors and use of inappropriate vocabulary.	▪ one or two minor grammar errors. ▪ vocabulary use is too elementary or not effective	▪ correct grammar ▪ vocabulary mostly appropriate for the purpose and the audience	▪ correct use of grammar ▪ use of some advanced language ▪ effective use of appropriate vocabulary for the purpose and for the audience	
Totals					

Solve problems by applying the software development process (mastered)

Student(s):

_____

Capstone Assessment
3 = Good 2 = Average 1 = Below Average 0 = Not gradable

Criteria	Comments	Points			
The project includes a clearly defined problem or need		3	2	1	0
The student provided sufficiently frequent status updates		3	2	1	0
The project includes modeling diagrams that accurately describe the context		3	2	1	0
All project files and source code required to build and execute are submitted		3	2	1	0
The software runs without error, and has reasonable error checking and exception handling		3	2	1	0
The interfaces have a professional appearance		3	2	1	0
The project includes a comprehensive test plan		3	2	1	0
The project includes documentation for installation and operation		3	2	1	0
Total Points out of 24:					
Comments:					

CS 4343: Database Management Systems

Create software using software engineering principles (reinforced)

1. One disadvantage of spreadsheet in comparison to database management systems is:
   a) reduced data duplication.
   b) program-data independence.
   c) limited data sharing.
   d) enforcement of integrity constraints.

2. A join that keeps unmatched rows from one or both tables is called a(n):
   a) equi-join.
   b) outer join.
   c) multivariate join.
   d) inner join.

3. List the types of database applications. (Please feel free to choose more than one correct answer.)
   a) PC databases
   b) workgroup databases

c) dept databases
d) enterprise databases
e) all of above

4. In term of relational database, Conceptual schema includes:
a) Conceptual design
b) Description of data requirements
c) Includes detailed descriptions of the entity types, relationships, and constraints
d) Transformed from high-level data model into implementation data model
e) All of above

5. Which of the following is not a component of ER models?
a) Entities
b) Relationships
c) Attributes
d) Store procedures

6. The CARTESIAN PRODUCT (Cross product) operation: Choose all that are true.
a) useful when followed by a selection that matches values of attributes
b) denoted by x
c) unary set operation
d) relations do not have to be union compatible
e) A, B, D

7. Match the following with each statement
Data definition language (DDL)
Storage Definition language (SDL)
View Definition Language (VDL)
Data Manipulation Language (DML)

a) allows retrieval, insertion, deletion, modification
b) specifies the internal schemas
c) defines both schemas
d) Specifies user views/mappings to conceptual schema

MATH 3023: Discrete Mathematics

1. Add the following binary numbers
10101111
+11000011

a) 101110010
b) 01110010
c) 111110010
d) None of these answers is correct

2. Add the following hexadecimal numbers
ABC
+ DEF

a) 18AB
b) 24AB
c) 181B
d) 182B

3. Add the following hexadecimal numbers
BAA
+ AAA

a) 1FFD
b) 16FE
c) 1120
d) 1654

4. Given that *p* and *q* are false and *r* is true, evaluate the proposition
$(q \rightarrow r) \rightarrow (\neg r \rightarrow p)$
a) FALSE
b) TRUE
c) UNKNOWN
d) None of these answers is correct

5. Given that *p* and *q* are false and *r* is true, evaluate the proposition
$(r \leftrightarrow p) \vee (p \rightarrow q)$
a) FALSE
b) TRUE
c) UNKNOWN
d) None of these answers is correct

6. In how many ways can we select a committee of 3 women and 3 men from a group of 5 distinct women and 7 distinct men?
a) 45
b) 350
c) 140
d) None of these answers is correct

7. How many strings can be formed using the letters *M I S S I S S I P P I* ?
a) 34650
b) 11!
c) 11
d) None of these answers is correct

8. Given the sets $A = \{1, 3, 5, 9\}$, $B=\{1, 2, 3, 5, 8\}$, $C = \{2, 4, 6, 10\}$, $U = \{1,2,3,\ldots,10\}$, list the elements that result from the following set operation:
$(B - C) - A$
a) {9}
b) {1,3,5,8}
c) The empty set
d) None of these answers is correct

Appendix E – Programming Assignments

Programmer(s)

_____

Program Assessment
3 = Good 2 = Average 1 = Below Average 0 = Not gradable

Criteria	Comments	Points			
1. The program fully implements the solution to the given problem		3	2	1	0
2. The solution produces the correct result		3	2	1	0
3. The program runs without error, has reasonable error checking and exception handling		3	2	1	0
4. The user is prompted for input when required		3	2	1	0
5. The user is presented with appropriate feedback as to the results of an operation		3	2	1	0
6. All project files and source code required to build and execute are submitted		3	2	1	0
7. Code is properly indented, and adequately commented		3	2	1	0
8. Code is well written and free from unnecessary complexity or redundancy		3	2	1	0
9. Programmer defined symbol names are reasonable (variable names, class names, function names, etc...)		3	2	1	0
10. The user interface is accurate and is free from misspelled words and bad grammar		3	2	1	0
		Total Points out of 30:			
Comments:					

Against "Flexibility": Tightening the Cage of Academic Rigor with Instructors' Responsibility and Rationality

Yaojie Li
li_yaojie@columbusstate.edu
Columbus State University

Jason Xiong
xiongjj@appstate.edu
Appalachian State University

Jennifer Pitts
pitts_jennifer@columbusstate.edu
Columbus State University

Scott Hunsinger
hunsingerds@appstate.edu
Appalachian State University

Abstract

The end of the COVID-19 pandemic is in sight, but it brought tremendous challenges and opportunities to academia. This paper analyzes the equilibrium between teaching laxity and strictness and corresponding outcomes through the lens of the approach-avoidance framework. On one side, instructors are likely to allow flexibility in course policies and procedures, given the learners' difficulties amid the pandemic. On the other hand, however, this intention could lead to a lack of control in class and eventually jeopardize academic integrity and rigor. Therefore, we explicate the motivation mechanism and the adverse effects of teaching laxity. Furthermore, a portfolio of teaching methods and mapping based on psychological distance and behavioral control theories is presented. This research contributes to a further understanding of pedagogical innovations in the Information Systems (IS) domain in the unprecedented crisis.

Keywords: COVID-19, IS Education, Laxity and Strictness, Academic Integrity and Rigor, Approach-Avoidance

1. INTRODUCTION

The end of the COVID-19 pandemic is near, but it has profoundly changed academia. Colleges and their stakeholders learned and grew from the challenges and difficulties caused by the global pandemic while gradually adopting the "new normal"(Barnes, 2020; Carroll & Conboy, 2020; Nah & Siau, 2020). Many universities have followed states on lifting mask mandates, while some others will remain in place as a pandemic precaution (Moody, 2022). Meanwhile, the school administration offers instructors more autonomy and flexibility than before in adjusting instructional activities and learning assessments. For example, many school administrators recommended choosing asynchronous lectures and system-embedded proctoring plugins rather

than synchronous lectures and commercial proctoring services.

This type of flexibility is in recognition that students may be ill, providing care for family members who are ill, or they may not have exclusive access to a computer and Internet service at the scheduled time. Besides, many students lost their part-time work on campus and in restaurants and struggled with personal financial difficulties (Brewster, 2020). Due to the infection dynamics and public policy uncertainties, school policies and procedures adapting to the epidemic are possibly inconsistent and delayed, leading to instructors' fears, confusion, and frustrations amid teaching from home. The instructor laxity could arise as instructors are disoriented or cannot expeditiously adjust themselves to the new teaching situation. In this sense, instructors may loosen the assessment rules while providing convenience for both sides. While we acknowledge the importance of flexibility (c.f. Americans with Disabilities Act or ADA in higher education) for accommodating students with disabilities, we argue that this *born-in-the-pandemic flexibility* may implicitly "support and satisfy" students who attempt to avoid heavy lectures and workload. A critical problem suffices – academic integrity and rigor can be impaired while the school's academic reputation can be challenged if the laxity in teaching and learning spreads and the pandemic.

In the following section, our discussion will revolve around a possible explanation of instructor laxity in teaching from home, drawing on the theoretical framework of approach-avoidance dynamics. More importantly, a portfolio of methods based on psychological distance and behavioral control theories will be introduced to mitigate the adverse effects of instructor and student laxity. Relevant teaching experience and examples will be illustrated, as well.

2. APPROACH-AVOIDANCE DYNAMICS IN TEACHING FROM HOME

The approach-avoidance dynamics have been recognized as a fundamental framework in motivation and decision theories (Carver & White, 1994; Elliot, 2006; Liang & Xue, 2009; Tversky & Kahneman, 1992). The term "approach" refers to moving near towards something, whereas "avoid" means keeping away from something. As shown in Figure 1, this research adopts the approach-avoidance dynamics in the context of teaching from home.

	Instructor Laxity	**Instructor Strictness**
Student Strictness	Arbitrary Laxity	Complete Rigor
Student Laxity	Collusive Laxity	Discontinued Rigor

Table 1: The Matrix of Instructor-Student Rigor and Laxity

There are two competing forces of positive and negative valence that act on an instructor simultaneously. On one side, instructors have to accomplish additional work, as most face-to-face courses have been interrupted and changed into an online model. For example, instructors have to reorganize learning resources, record videos, deliver virtual lectures, and set up virtual office hours. On the other side, instructors need to help address students' personal issues (e.g., illness, technical issues, or other pandemic-relevant instances), possible infection issues for themselves, and domestic distractions and interruptions (Myers et al. 2020). As such, some instructors may tend to choose a temporary, easy teaching model, such as reducing student workloads and offering grade leniency. Saliently, instructor-student interactions have been reduced because of reduced mandatory tasks (e.g., face-to-face office hours and class activities) and increased leeway for virtual activities (e.g., virtual office hours and class activities). From this, instructor laxity can undermine teaching quality while hindering students' motivation in pursuing academic rigor with enthusiasm amid the pandemic.

As illustrated in Table 1, we propose a classification of laxity and rigor based on the interactions between instructors and students. *Arbitrary laxity* is formulated as the instructor would pursue ease and convenience in teaching from home, despite that some students would like to keep strict class requirements and assessments. In contrast, instructor laxity can often lead to *collusive laxity* - when laxity exists in both teaching and learning. Students can do less or avoid work with minor or minimal punishment (e.g., lower course grades and course failures) because of loosened course policies implemented by instructors. Indeed, *complete rigor* cannot be achieved without either instructor consent and student support – if instructors adhere to strict and rigorous learning requirements and assessments, students may avoid them by using the pandemic as an excuse, then ensuring the academic rigor is likely to discontinue (*discontinued rigor*).

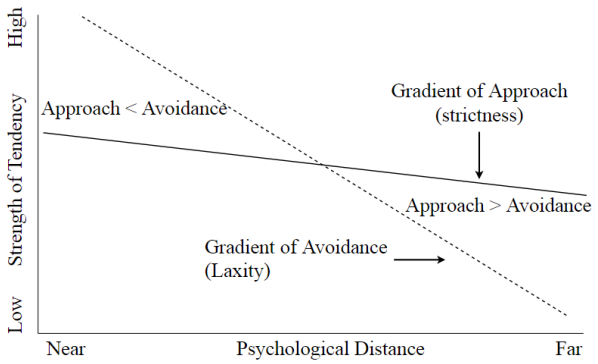


Figure 1: The Approach-Avoidance Dynamics in Teaching from Home (Adapted from Miller, 1944)

Whereas instructors may give academic laxity and excessive "flexibility," most instructors, or instructors most of the time, we believe, are dedicated to their career since teaching is not only a knowledge-intensive, sophisticated job but a prestigious profession that aims at instructing, inspiring, and illuminating students, even in difficult times like the present. While it is not mandatory, many instructors assume more roles and shoulder more responsibilities in supporting students to face challenges while maintaining the rigor of the learning community. Instructors have a solid intention to approach challenging but meaningful work while being distracted by "flexible" teaching at the cost of academic rigor. Indeed, instructors are experiencing a conflict between approaching academic rigor (i.e., avoiding laxity) and avoiding arduous work (i.e., approaching laxity).

In this approach-avoidance conflict, one needs to recognize that as the psychological distance extends (i.e., the degree to which people feel removed from a phenomenon or an object, see Figure 1), the slope of avoidance becomes steep, indicating the tendency to be lax is diminished. In contrast, the slope of avoidance will move slowly, as the tendency to be strict with academic requirements is strengthened or strictness becomes easier to achieve. Further, in this case, we would see when approach (strictness and rigor) is more robust than avoidance, eventually.

To that end, two approaches can be used to promote the advancement of rigor rather than laxity. The first method is to devalue the valence of laxity. For instructors, it can be intervened by a series of self-consciousness and self-suggestion when instructors are conscious of their environment (e.g., how the pandemic influences the learning context and learners) and of themselves (e.g., how should we adapt to this situation). For example, talking to yourself – "Cyberloafing to me is relatively meaningless as a responsible professor who loves his or her career and students," "I'm an educator who adheres to his or her teaching philosophy and principles."

Another method – extending the psychological distance, includes more practical strategies and tactics. For instance, Rebecca Hamilton (2015) illustrated three aspects that people can adjust the distance in the workplace. Likewise, in work/teaching from home, we can apply these good practices. First, instructors can manage the hypothetical distance, imagining that an event is likely or unlikely. For instance, if one wanted to become a responsible and respected professor among students – high teaching evaluations, good word-of-mouth, and self-value actualization, then he or she would avoid laxity while moving toward strictness with extra efforts. The unlikely circumstances, such as failure to gain tenure or cutoff, will drive an instructor to pursue a high quality of instruction.

Also, we can "manually" gear the temporal and spatial distances toward strictness rather than laxity. For example, using self-imposed deadlines and schedules, i.e., in the weekly announcement, informing students that you will post a video about the special topic about forecast methods and COVID-19 infection cases, or there is a virtual Q&A session you will be there, at least. From this, instructors can visualize future events and detailed procedures to accomplish the goal. As for spatial distance, an excellent way to manipulate it is to choose a suitable place to focus on the teaching – e.g., your study room in the early morning or evening, when and where you can avoid domestic influence and interruptions. Social distance (not to be confused with public health social distance for reducing virus spread) can be utilized to increase virtual communications and interactions with your students, colleagues, and other stakeholders in the learning community. Sharing is a good thing; information, knowledge, feelings, ideas, thoughts, and, more importantly, support are all encouraged. The potential benefits of social distance are experience and skills to promote teaching from home and psychological well-being.

3. THREE METHODS TO MITIGATE THE LAXITY IN TEACHING

This distance manipulation can be replaced with each other or applied in a portfolio. Hence, we synopsize and suggest three practical methods (3S methods) to mitigate laxity while improving

academic strictness and rigor – *the self method, the solid method, and the systematic method* (Table 2). One stream of literature is the behavioral control theory extensively applied in Information Systems studies (e.g., Jaworski, 1988; Kirsch, 1997; Choudhury & Sabherwal, 2003; Soh et al., 2011). Briefly stated, the self method is based on an instructor's responsibility and self-control. Besides estimating the likelihood of an event occurring (e.g., furloughs, layoff, tenure, promotion), one's rumination and reflection on their career pursuit largely determine their approach-avoidance intention. For that, a self method can be a good starting point for instructors to change the prior equilibrium of acceptance and avoidance.

Instead, the solid method relies on fixed and established rules and procedures in the teaching at home. It can be directed by the school administrator or self-imposed. It is fixed schedules, timelines, and "places" to meet, rather than a flexible manner that an instructor can arbitrarily modify or cancel. Hence, managing temporal and special distances is essential in this method.

Lastly, the systematic method is a social solution to engage with significant stakeholders in the learning system. With that, each member can be mutually supported and monitored. Also, this method embodies the spirit of learner-based instruction, as shown in many pedagogy studies (e.g., Barr & Tagg, 1995; Landry et al., 2008; Saulnier et al., 2008; Weimer, 2002). In this method, instructors need to leverage social distance management.

4. CASE ANALYSIS AND COURSE MAPPING

This section provides a course mapping for the Information Systems discipline to discuss the challenges, teaching practices, and methods applied.

Principles of IS/IT Management
Course Description
This course mainly provides a solid foundation and overview of information systems in business and emphasizes how competitive strategies for companies are formulated and implemented using a combination of information technologies.

Challenge
As an entry-level IS course, one of the biggest challenges is the large number of topics and concepts related to information systems, IT strategies, and emerging technologies to be taught.

The Self Method (Responsibility-based)	
Actors	Instructors – the re-designers of course policies and procedures
Activities	Instructors perform their roles and responsibilities in adapting to teaching from home through practicing self-suggestion and self-control. The goal of the activities here is to address the instructor's laxity. "Self-Control"
Psy. Distance Adjust.	Hypothetical distances, e.g., "I will make a good teacher."
The Solid Method (Rule-based)	
Actors	Instructors and students – the strict implementers of the updated course policies and procedures
Activities	Instructors self-impose fixed schedules and deadlines to promote the accomplishment of teaching and learning tasks "Structural and Process Control"
Psy. Distance Adjust.	Temporal distances and spatial distances, e.g., "I have virtual office hours from 2 PM to 4 PM to meet with my students."
The Systematic Method (Relationship-based)	
Actors	Instructors and students (i.e., co-learners) – the re-developer of course policies and procedures
Activities	Instructors work with students, as well as other stakeholders in the learning community to share knowledge, feelings, thoughts while mitigating the laxity and inefficient learning "Social Control"
Psy. Distance Adjust.	Social distances, e.g., "I learned a lot about online teaching from my colleagues from virtual seminars, workshops, meetings, and interactions with my students in discussion forums."

Table 2: Descriptions for Three Teaching Methods from Home

In addition, there are some unique challenges in teaching entry levels courses. For example, a considerable number of business students whose major is not CIS/MIS/IS/IT will also need to take this course. Hence, teaching the course entirely online might not be the best option during the pandemic.

Teaching Practice
It would be helpful to include emerging technologies, especially technologies playing an essential role during the pandemic, such as

machine learning for detection and diagnosis, big data analytics for tracking, supporting infrastructure, and blockchain technologies in business (He et al., 2021). Students will then have a better appreciation of how information systems can support and sharpen the business process. Additionally, faculty can take this opportunity to record a short video for each of the upper IS courses they are teaching.

Method Applied

The Solid Method. As a foundational course, it is appropriate to adopt the self method with fixed schedules and deadlines to promote instructors' teaching and learning tasks.

Business Data Networks & Security

Course Description

This course introduces networks and data communications, including the design, administration, and theory of local and wide area network systems. Students will learn to plan and design computer networks based on their understanding and lab practices using various network software. Topics include data communication technologies, network architectures, internetworking, protocols for data link, network, transport, and application layers, effective network design, planning, implementation, wireless technologies, network management, and security.

Challenge

It is challenging to show network models entirely online. Many of the activities designed for a lab environment will have to be converted online. This challenge dramatically increases the workload of faculty to convert lab-related materials online.

Teaching Practice

It will be helpful to include some hands-on activities online. Faculty may also incorporate small group projects into online activities.

Method Applied

The Self Method. In this course, instructors should perform their roles and responsibilities in adapting to teaching from home through practicing self-suggestion and self-control – that means, they need to do some extra work to set up the "virtual lab" for students, including seeking appropriate virtual software for students and troubleshoot their problems, often.

Database Management

Course Description

This course teaches students how to use data to stay competitive in a changing business environment. Topics include relational database methodology, modeling, design, database administration, structured query language (SQL), data preparation for analysis, and current innovations and trends in the corporate environment (e.g., NoSQL, distributed data storage, blockchain).

Challenge

It is challenging to show students how to run SQL code without a lab setting.

Teaching Practice

Similar to the network course, it will be helpful to add virtual machines to students so that an online lab can be created. It will be beneficial to students to learn database management in a standard environment.

Method Applied

The Self Method. Again, in this course, instructors should perform their roles and responsibilities in adapting to teaching from home through practicing self-suggestion and self-control while going above and beyond their roles to support students in difficult times.

Systems Analysis & Design

Course Description

This course provides a comprehensive introduction to the strategies and technologies for building information systems in organizations. The course covers a general process for information system development such as analysis, design, development, implementation, and maintenance. Systems analysis and design tools will be used to understand information system issues and design the information systems that address the issues.

Challenge

It is challenging to explain different diagrams online. Also, it may be challenging to explain information systems-related issues online.

Teaching Practice

It will be helpful to pre-record some of the cases and then ask students to work as smaller teams to understand further the crucial steps and roles in the system analysis and design.

Method Applied

The Self Method. Like foundational and concepts-based courses, it will be important for faculty to be strict in implementing the updated course policies and procedures.

Cybersecurity Management
Course Description
This course focuses on the broad areas of descriptive, predictive, and prescriptive analyses to gain insight into an organization's functioning, make predictions, and prescribe courses of action. In addition, students learn to utilize contemporary analytics software and collect data from various sources.

Challenge
Like the network and database course, many activities designed for a lab environment will have to be converted online. This situation dramatically increases the workload on faculty to develop a virtual lab while moving learning resources and materials online.

Teaching Practice
It will be helpful to include some of the most recent cases related to cybersecurity so that students can develop a better understanding of the critical role of cybersecurity in the modern business world. Some special topics related to cybersecurity and privacy during the pandemic would be relevant and engaging for students who need to understand real-world examples.

Method Applied
The Self, Solid, Systematic Method. The reason is the hybrid nature of security courses in the IS domain – behavioral, managerial, and technical aspects. On the one hand, instructors need to reorganize and revamp the learning materials for behavioral and managerial security knowledge. On the other hand, instructors need to set up a "virtual lab" to help students achieve these technical learning outcomes. The systematic method is also appropriate for cybersecurity courses. Through this method, the instructors will need to work with students and other stakeholders to redesign the course. It is important as knowledge and trends in cybersecurity are constantly changing. Notably, COVID-19 brings additional challenges to cybersecurity. Thus, it will be helpful to be more flexible in this course.

Business Analytics
Course Description
This course focuses on the broad areas of descriptive, predictive, and prescriptive analyses to gain insight into an organization's functioning, make predictions, and prescribe courses of action. Students learn to utilize contemporary analytics software and collect data from a variety of sources.

Challenge
Students may need additional support in understanding the technical aspect of business analytics. Without face-to-face hands-on experiences, it is hard to learn the technical components online.

Teaching Practice
Instructors should convert some of the materials online and update some case analyses related to COVID, e.g., forecast analyses using confirmed cases data and vacation data from these public health agencies.

Method Applied
The Self Method. As the content is relatively stable during the pandemic. It would be reasonable to let instructors impose fixed schedules and deadlines to promote teaching and learning tasks.

IS/IT Capstone
Course Description
Using a team concept, students will analyze, design, create and implement a working information system for a public or private organization. Emphasis will be placed on IT project management, rapid application development, quality assurance, and implementation of the system.

Challenge
It is hard for students to accomplish projects in a virtual team.

Teaching Practice
Faculty could help students find potential clients/sponsors off-campus. Also, faculty can consider working with other departments on campus and find internal projects that can fulfill the requirements of the project.

Method Applied
The Systematic Method. It is clear that flexibility should be provided in the course activities. Also, instructors should ardently work with students and external clients to identify a project that can serve the purpose of the capstone.

5. DISCUSSIONS

Like many other studies, this study has several limitations that can be used for future research. First, while this study includes both instructors' and students' laxity and strictness in its framework, it can shed light on more details about the interaction between instructors and students while coping with the laxity caused by the pandemic and other contingent factors. In

addition to the three control methods, we proposed for the instructors, they should also be applicable to students. Nevertheless, the effects of the proposed methods on students may be different from that of instructors, particularly the self method and the solid method. Therefore, we would argue the critical role of instructors in promoting control for students, for example, enforcing the specific rules and learning goals for students at early stages while encouraging students to redesign the methods on their own. Taken together, the last method – the systematic method, should be the most efficient way to achieve complete rigor and mitigate collusive laxity since instructors and students can monitor, support, and encourage each other in maintaining the rigor of teaching and learning, toward a collaborative learning community. Second, this study relies heavily on a conceptual framework based on the approach-avoidance model and the teaching experience of several IS faculty contributed to this article. Hence, it is worthwhile to further theorize and examine the dynamics and driving factors of the classified rigor and laxity with empirical evidence. Lastly, while our study is situated in the context of the pandemic, future studies can look into instructors' and students' laxity and rigor during the normal time.

6. CONCLUSIONS

In this paper, we acknowledge that instructor laxity, in the guise of flexibility, has been a popular excuse for avoiding adequate course preparation and design (possibly including extra and redundant workloads) in adapting to the interruptions due to the COVID-19 epidemic from our teaching experience. Such laxity should be mitigated and reduced in that this could hurt our professional standards and performance, school reputation and enrollments, and, more importantly, our students' motivation to learn at this challenging time. Also, flexibility should be accommodated and given to our colleagues and students with disabilities and real needs. Inspired by the approach-avoidance conflict and behavioral control theories, we propose three methods for bridging the psychological distance in teaching from home and hope these teaching tips will be helpful in our community.

7. REFERENCES

Barnes, S. J. (2020). Information management research and practice in the post-COVID-19 world. International Journal of Information Management, 55, 102175.

Barr, R. B., & Tagg, J. (1995). From teaching to learning—A new paradigm for undergraduate education. Change: The Magazine of Higher Learning, 27(6), 12-26.

Brewster, J. (2020). Poll: 38% Of College Students Lost Work Because Of Coronavirus. Forbes. Retrieved from https://www.forbes.com/sites/jackbrewster/2020/04/15/poll-38-of-college-students-lost-work-because-of-coronavirus/?sh=587a24d75ec9

Carroll, N., & Conboy, K. (2020). Normalising the "new normal": Changing tech-driven work practices under pandemic time pressure. International Journal of Information Management, 55, 102186.

Carver, C. S., & White, T. L. (1994). Behavioral inhibition, behavioral activation, and affective responses to impending reward and punishment: the BIS/BAS scales. Journal of Personality and Social Psychology, 67(2), 319-333.

Choudhury, V., & Sabherwal, R. (2003). Portfolios of control in outsourced software development projects. Information Systems Research, 14(3), 291-314.

Elliot, A. J. (2006). The hierarchical model of approach-avoidance motivation. Motivation and Emotion, 30(2), 111-116.

Hamilton, R. (2015). Bridging psychological distance. Harvard Business Review, 93(3), 116-119.

He, W., Zhang, Z. J., & Li, W. (2021). Information technology solutions, challenges, and suggestions for tackling the COVID-19 pandemic. International journal of information management, 57, 102287.

Jaworski, B. J. (1988). Toward a theory of marketing control: environmental context, control types, and consequences. Journal of Marketing, 52(3), 23-39.

Kirsch, L. S. (1997). Portfolios of control modes and IS project management. Information Systems Research, 8(3), 215-239.

Landry, J. P., Saulnier, B. M., Wagner, T. A., & Longenecker, H. E. (2019). Why is the learner-centered paradigm so profoundly important for information systems education? Journal of Information Systems Education, 19(2), 6.

Liang, H., & Xue, Y. (2009). Avoidance of information technology threats: A theoretical perspective. MIS Quarterly, 71-90.

ILO (2020). ILO Monitor 2nd edition: COVID-19 and the world of work. Geneva: International Labor Office.

Miller, N. E. (1944). Experimental studies of conflict. In J. McV. Hunt (Ed.), personality, and the behavior disorders (Vol. 1, pp. 431-465). New York: Ronald Press.

Moody, J. (2022). Colleges follow states on lifting mask mandates. Retrieved from https://www.insidehighered.com/news/2022/02/15/colleges-follow-states-lifting-mask-mandates

Myers, K., Tham, W. Y., Yin, Y., Cohodes, N., Thursby, J. G., Thursby, M., ... & Wang, D. (2020). Quantifying the Immediate Effects of the COVID-19 Pandemic on Scientists. Available at SSRN 3608302.

Nah, F., & Siau, K. (2020). Covid-19 pandemic–role of technology in transforming business to the new normal. In the International Conference on Human-Computer Interaction (pp. 585-600). Springer, Cham.

Saulnier, B. M., Landry, J. P., Longenecker Jr, H. E., & Wagner, T. A. (2008). From teaching to learning: Learner-centered teaching and assessment in information systems education. Journal of Information Systems Education, 19(2), 169-174.

Soh, C., Chua, C. E. H., & Singh, H. (2011). Managing diverse stakeholders in enterprise systems projects: a control portfolio approach. Journal of Information Technology, 26(1), 16-31.

Tversky, A., & Kahneman, D. (1992). Advances in prospect theory: Cumulative representation of uncertainty. Journal of Risk and Uncertainty, 5(4), 297-323.

Weimer, M. (2002). Learner-centered teaching: Five key changes to practice. John Wiley & Sons.

WHO (2020). Coronavirus disease 2019 (COVID-19). Situation Report -177. Retrieved from https://www.who.int/docs/default-source/coronaviruse/situation-reports/20200715-covid-19-sitrep-177.pdf?sfvrsn=b1a193f3_2

Upskilling and Reskilling for the Future of Work: A Typology of Digital Skills Initiatives

Guido Lang
guido.lang@quinnipiac.edu

Tamilla Triantoro
tamilla.triantoro@quinnipiac.edu

Quinnipiac University
Hamden, CT 06518

Abstract

Governments, businesses, and educational institutions need to collaborate and make significant investments in order to address the growing digital skills crisis. In Europe, hundreds of digital skills initiatives have been launched with different forms of government and private industry support in the last five years alone. Consequently, digital skills initiatives have come to encompass a wide array of interventions. In this context, this paper proposes a typology of digital skills initiatives that was developed based on the analysis of over 300 initiatives listed in the European Commission's repository of best digital skills initiatives. The proposed typology consists of four categories: target group, digital skills, learning format, and sponsoring organization. In terms of target group, digital skills initiatives tend to target one or more of five distinct groups: the general public, educators, adults, seniors and youth. In terms of digital skills, digital skills initiatives tend to focus on general digital skills or specialized digital skills. In terms of learning format, digital skills initiatives tend to offer training and/or a learning resource. In terms of sponsoring organization, the initiatives tend to be sponsored by organizations that are either affiliated or unaffiliated with a technology vendor. The typology is followed by a presentation of mini cases, which highlight different archetypes of the proposed typology. The paper closes with a discussion of practical implications for policy makers, administrators, and scholars interested in digital skills initiatives and the future of work.

Keywords: future of work, digital skills, digital skills gap, digital skills initiatives, typology

1. INTRODUCTION

Technological advancements and digital transformation require the society to adapt digital experiences and acquire digital skills. Digital transformation powered by digital and AI technologies and accelerated by the COVID-19 pandemic, influences individual, organizational, and societal levels. Thus, the development of digital skills has become a focal point on the agenda of policymakers, practitioners, and scholars to empower citizens to fully participate in the increasingly digitized world (Iordache et al., 2017).

Digital skills are dynamic and evolving skills necessary to use software or operate a digital device (van Laar et al., 2020). The future of work depends on digital skills. Governments, businesses, and educational institutions need to collaborate and make significant investments to address the digital skills crisis, which is a gap between necessary digital skills and available digital skills. Approximately 90% of jobs in developed economies require some level of digital skills, while one third of the labor force has a limited ability to use digital skills productively (Broadband Commission for Sustainable Development, 2017). This digital skills gap negatively affects all industries, workers, and

societies and is projected to grow in the foreseeable future. In Europe, hundreds of digital skills initiatives have been launched with different forms of government, academic and organizational support in the last five years alone. Consequently, digital skills initiatives have come to encompass a wide array of actions and offerings.

The goal of this paper is to advance the development of digital skills initiatives by creating a typology based on the analysis of existing digital skills initiatives. A typology is a classification of practice that is used to extract complex data via unearthing key categories (Nind & Lewthwaite, 2020). As such, the proposed typology is useful both as a descriptive tool and as a thinking tool. The complexity of digital transformation brings challenges, which in turn make a typology a valuable tool for policymakers, practitioners, and scholars. Specifically, this paper addresses the following research question: *What categories can be used to classify digital skills initiatives?* The proposed typology, which consists of four categories, has been developed by analyzing the European Commission's Digital Skills & Jobs Coalition Initiatives Repository (European Commission, n.d.). The European Commission, in collaboration with European Union (EU) member states, identified over 300 best digital skills initiatives in Europe, and created a public repository with detailed information about each initiative (European Commission, n.d.), which served as the data source for the present study.

The paper is structured as follows: The next section presents an overview of related literature. Section three describes the methodology used to develop the typology. Section four presents the proposed typology, including its four categories. Section five summarizes mini cases that represent archetypes, or typical examples of digital skills initiatives. Finally, the last section discusses practical implications for policy makers, practitioners, and scholars.

2. RELATED LITERATURE

Digital skills are a key to a successful information society. A recent report by the European Commission (2021) suggests that digital skills are required in all types of jobs, including those that are not directly associated with digitization including farming, construction, and vocational skills. Digital skills are necessary to use software and enable people to use digital services, engage in online activities, consume information, and communicate online. According to the latest Digital Economy and Society Index report (DESI, 2020), although there is an improvement year over year in terms of basic internet user skills and advanced information and communication technology (ICT) skills, there remains a shortage of ICT specialists. In fact, 64% of large enterprises and 56% of small and medium enterprises reported difficulties in filling vacancies for ICT specialists in 2018.

Prior research suggests that the divide between digital skills adoption depends on demographic and socio-economic factors such as age, gender, education, income, and employment status (DiMaggio et al., 2004; Iordache et al., 2017). Digital skills are a prerequisite for the general public to be able to access government related information, participate in online banking, obtain telehealth treatments, consume information, and communicate online. Unfortunately, only 58% of citizens in the EU had basic digital skills in 2019 (DESI, 2020). Despite the attention given to bridging the digital divide, there is a growing concern that the digital divide may have been deepening because of unequal distribution of digital skills and access to digital media (van Dijk & van Deursen, 2014). The unequal access can be generation based. For example, senior citizens are on the lower spectrum of internet adoption, and significant efforts are required to provide basic digital skills to this demographic group (Blažič & Blažič, 2020). On the other side of the spectrum are young adopters of technology. Technology skills among youth are important for the future of work as the new economy requires reskilling with a focus on digital solutions, with an expected doubling of digital labor demand by 2030 (Bughin et al., 2017). Moreover, digital technologies increasingly provide access to communication, entertainment, creativity, citizenship, and civic participation (Donoso et al., 2020). Educators play a significant role in bringing technology skills to the lives of younger individuals via the use of e-learning tools and ICT (Leahy & Wilson, 2014).

Digital skills can be characterized as general and specialized. General skills are non-specialized, generic skills (Lintzeris & Karalis 2020). In the context of digital skills, they include basic functional digital skills necessary for social inclusion and participation in modern day-to-day life, such as access to the Internet, understanding social and digital media, and accessing services online (Broadband Commission for Sustainable Development, 2017). Specialized digital skills are a prerequisite for digital transformation and are necessary to create competitive business models and demonstrate profitability and sustainability. Specialized skills are particularly in high demand

among organizations and include mobile technologies, data analytics, cloud computing, and the Internet of things (Krcmar et al., 2017). Post-pandemic trends accelerate the need for reskilling and upskilling for adults. Some of the trends include the ability of employees to fully operate in a remote world while being productive within the company's ecosystem that includes clients, partners, suppliers, and public authorities. This includes not only moving to a more technology and data-enabled model, but also a basic understanding of critical technology and data concepts including coding, robotics, 3D printing, cybersecurity, data visualization, applied machine learning, and advanced analytics, among others (Agrawal et al., 2020).

To bridge the gap and transform the digital profiles of their countries, many governments spearheaded policy interventions and digital skills initiatives. For example, the Institute of Coding is a £40m initiative of the UK government to transform the digital skills of the country (Davenport et al., 2020). The government of France designated one billion euros for digital education development (Broadband Commission for Sustainable Development, 2017). At the same time, there is interest in increasing the level of digital skills from the private industry. Microsoft launched a global skills initiative aimed at bringing digital skills to 25 million people globally by combining resources from LinkedIn, GitHub, and Microsoft (Smith, 2020). Amazon Web Services (AWS) stated that by 2025, they will provide free cloud computing skills training to 29 million people worldwide (Carlson, 2020).

Prior research focused on the origins and concepts of digital literacy (Bawden, 2008), covered models of constructing a modern and comprehensive skills classification that includes digital skills along with information, communication, collaboration, critical thinking, creativity, and problem-solving skills (Lintzeris & Karalis, 2020; van Laar et al., 2020). Instruments have been developed to measure digital skills related to the internet and web (Hargittai & Hsieh, 2012; Van Deursen et al., 2016). The assessment of digital competencies was conducted in schools (Calvani, 2008) and workplaces (Vieru et al., 2015).

While previous studies provided a thorough overview of digital skills and some of the applications, there has been no systematic effort to classify digital skills initiatives. In addition, there has been a lack of studies with a cross-case approach to existing digital skills initiatives. The cross-case approach to creating a typology based on methods that have been used in practice is considered important research (Nind & Lewthwaite, 2020). In this respect, our study fills an important research gap by offering a typology of existing digital skills initiatives. In addition, prior research considered the demand side, mainly focusing on necessary digital skills for a consumer (e.g., Lintzeris & Karalis, 2020; van Laar et al., 2020). The focus of this research is on the supply side, with a focus on the evaluation of existing resources while providing a framework to assess and develop digital skills initiatives. Based on our knowledge, this is the first paper to analyze different initiatives related to the development of digital skills.

3. METHODOLOGY

The typology was developed using an inductive strategy (Bailey, 1994) by analyzing data from the Digital Skills & Jobs Coalition Initiatives Repository of the European Commission (n.d.). The repository contains 312 digital skills initiatives which the EU considers to be the best digital skills initiatives in Europe. The data included detailed descriptions for each digital skills initiative. The descriptions were iteratively analyzed: Starting without any preconceived categories, the authors first independently identified recurring categories in about 50 initiatives, before collaboratively agreeing on a set of categories. The categories were chosen to form groups of digital skills initiatives that are internally homogenous but distinct from each other (Bailey, 1994). Subsequently, the authors independently re-analyzed all previously analyzed initiatives along with the next set of 50 initiatives. Afterwards the authors collaboratively revised the categories, as needed, to derive a set of categories that maximize both homogeneity within groups and heterogeneity between groups (Bailey, 1994). This process was repeated until all digital skills initiatives were analyzed. For example, after reading the descriptions of the first 50 digital skills initiatives, one of the authors noticed that digital skills initiatives are sponsored by an organization which is either affiliated with a technology vendor or not affiliated with a technology vendor. After discussing the initial finding with the other author, both authors then independently analyzed the next 50 digital skills initiatives with regards to the vendor affiliation of the sponsor. Afterwards both authors agreed that vendor affiliation of the sponsor can be used to separate digital skills initiatives into two roughly equal-sized groups. A shared Google Doc was used to record observations, markup the descriptions, and assign categories. Finally, the authors jointly selected archetypes, or typical

examples, for the proposed typology as described in section 5.

4. PROPOSED TYPOLOGY

The analysis revealed a large variety of digital skills initiatives with different and evolving concepts. To bring order into this complexity, we introduce a typology based on four overarching categories. Each category addresses a fundamental question behind digital skills initiatives. Figure 1 depicts the proposed typology, showing the four categories in onion-like layers with the target group at the center, surrounded by the digital skills, learning format, and ultimately the sponsor.

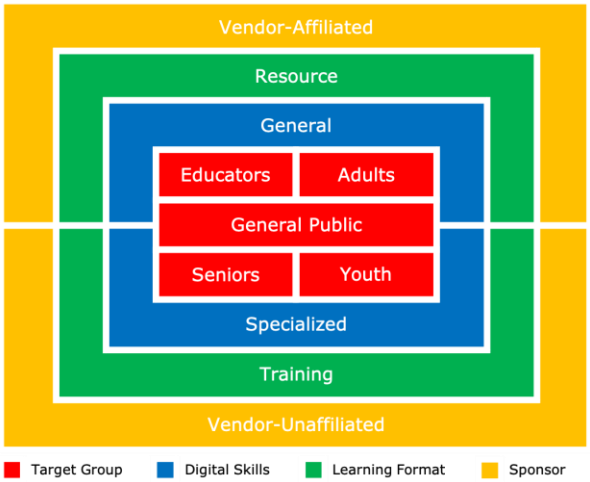


Figure 1: Proposed typology of digital skills initiatives

The following sections describe each of the four categories of the proposed typology in greater detail.

Target Group
This category addresses the question "whose digital skills are improved?". Digital skills initiatives tend to target one or more of five distinct groups: the general public, educators (such as K-12 teachers, vocational school instructors, university professors, etc.), adults (such as professionals wishing to reskill or upskill, veterans, the unemployed or underemployed, women, disadvantaged groups, etc.), seniors, and youth (such as children, students, young adults, young refugees, etc.).

Digital Skills
This category addresses the question "what digital skills are improved?". Digital skills initiatives tend to focus on general digital skills such as Internet usage, word processing, etc., and/or specialized digital skills such as coding, robotics, 3D printing, etc.

Learning Format
This category addresses the question "how are digital skills improved?". Digital skills initiatives tend to offer training such as classes, workshops, summer camps, etc., and/or learning resources such as e-books, websites, videos, games, etc.

Sponsor
This category addresses the question "how is the digital skills initiative funded?". Digital skills initiatives require scaling up that necessitate an investment, particularly from governments and the industry (Broadband Commission for Sustainable Development, 2017) and may include monetary investments, donations or loans of hardware and software, and volunteer work. As a result, digital skills initiatives tend to be sponsored by organizations that are vendor-affiliated or vendor-unaffiliated. Here, vendor affiliation refers to the relationship between the sponsoring organization and a technology vendor (i.e. an organization that sells products or services related to digital technology). Technology vendor affiliation is relevant to the sponsoring of digital skills initiatives since they have the potential to increase the demand for certain digital products or services in the future.

The proposed typology consisting of four categories is as simple as possible and as complex as necessary to classify the variety of existing digital skills initiatives. The following section elucidates the usefulness of the proposed typology by presenting archetypes, or typical examples.

5. ARCHETYPES

Theoretically, there are 40 different combinations of target group, digital skills, learning format, and sponsor possible. We have selected one archetype, or typical example (Bailey, 1994), from the Digital Skills & Jobs Coalition Initiatives Repository of the European Commission (n.d.) for each target group. We chose the target group as the defining category for the archetypes, as the needs and characteristics of the target group should be at the center of any digital skills initiative. Table 1 summarizes the resulting five archetypes.

Arche-type	Target Group	Digital Skills	Learning Format	Sponsor
1	General public	General	Resource	Vendor-Unaffiliated
2	Educators	Specialized	Training	Vendor-Affiliated
3	Adults	Specialized	Training	Vendor-Affiliated
4	Seniors	General	Training	Vendor-Unaffiliated
5	Youth	Specialized	Training	Vendor-Unaffiliated

Table 1: Overview of Archetypes

The aim of the following sections is to highlight the similarities and differences across a wide range of digital skills initiatives, thereby showcasing the appropriateness of the proposed typology.

Archetype 1: Digital Duel (Belgium)
The first archetype represents digital skills initiatives that provide the general public with general digital skills by offering a resource while being sponsored by a vendor-unaffiliated organization. Figure 2 depicts archetype 1.

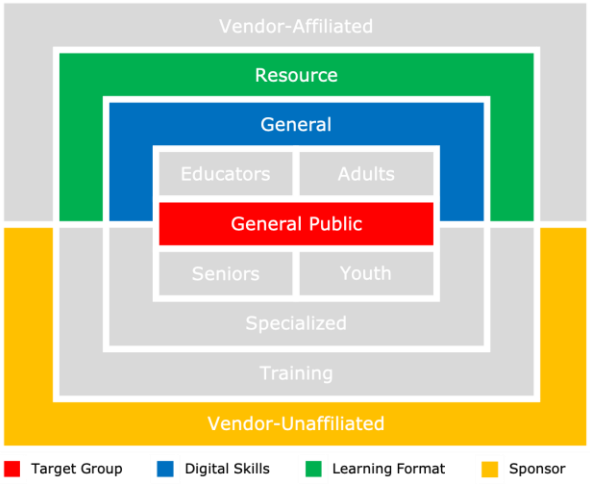


Figure 2: Depiction of Archetype 1

An illustrative example of a digital skills initiative in archetype 1 is *Digital Duel. Digital Duel* is a digital skills initiative that targets citizens in Belgium. It has been designed to enable all Belgian citizens to become active members of the digital society. Thus, the target group of *Digital Duel* is the general public.

Digital Duel aims to help citizens assess and improve their digital skills in five key areas: information processing, communication, content creation, safety, and problem solving. Thus, the digital skills provided by *Digital Duel* are general in nature.

Digital Duel is an online game. It consists of ten interactive challenges which require participants to apply their digital skills. For example, instead of asking participants to rate their ability to conduct online searches, participants have to actually conduct an online search in order to complete a challenge in the game. After completing the ten challenges, participants can see their level of digital skills in each of the five key areas. Finally, participants are given access to free online courses to improve their digital skills in the identified areas. Thus, the learning format offered by *Digital Duel* is that of a resource.

Digital Duel is sponsored by Federal Public Service Economy, a federal government agency tasked with ensuring Belgium's competitiveness in the international economy. Thus, the sponsor of *Digital Duel* is vendor-unaffiliated.

Archetype 2: Informatica365 (Romania)
The second archetype represents digital skills initiatives that provide educators with specialized digital skills by offering training while being sponsored by a vendor-affiliated organization. Figure 3 depicts archetype 2.

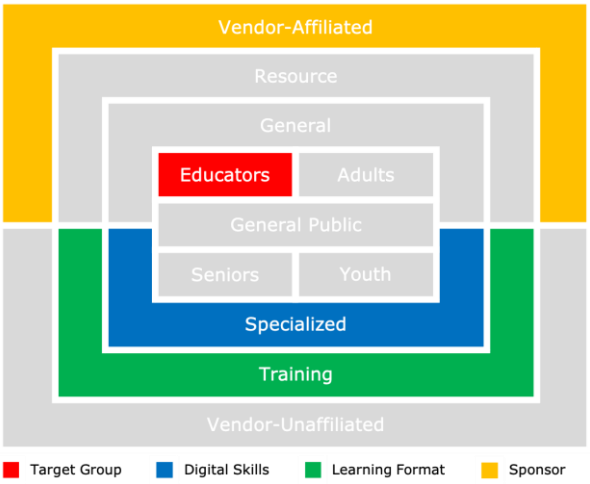


Figure 3: Depiction of Archetype 2

An illustrative example of a digital skills initiative in archetype 2 is *Informatica365. Informatica365* is a digital skills initiative that targets middle school teachers in Romania. The initiative aims to train teachers so that they can teach computer science (CS) to middle school students. Thus, the target group of *Informatica365* is educators.

Informatica365 aims to help teachers acquire the knowledge necessary to teach CS at grades 5-8 (ages 11-15). Topics covered include algorithms, sequencing, loops, conditionals, operators,

events, variables, and functions. The topics are taught in a middle school appropriate context using Microsoft Minecraft Education Edition. Microsoft Minecraft Education Edition is a version of the popular Minecraft game with additional education-specific features. The curriculum is aligned with the CS curriculum standards by the International Society for Technology in Education and the Computer Science Teachers Association. Thus, the digital skills provided by *Informatica365* are specialized in nature.

Informatica365 consists of 42 hours of face-to-face teacher training, which has been accredited by Romania's National Center for Teacher Training. Selected teachers received additional training to become master trainers, who can then train other teachers. In addition, lesson plans and other teacher training materials have been developed or translated to Romanian. Thus, the learning format offered by *Informatica365* is that of training.

Informatica365 is sponsored by Microsoft Philanthropies. Microsoft Philanthropies is part of Microsoft's corporate social responsibility initiative and aims to increase access to digital skills and CS education for youth around the globe. Thus, the sponsor of *Informatica365* is vendor-affiliated.

Archetype 3: AWS re:Start (UK)
The third archetype represents digital skills initiatives that provide adults with specialized digital skills by offering training while being sponsored by a vendor-affiliated organization. Figure 4 depicts archetype 3.

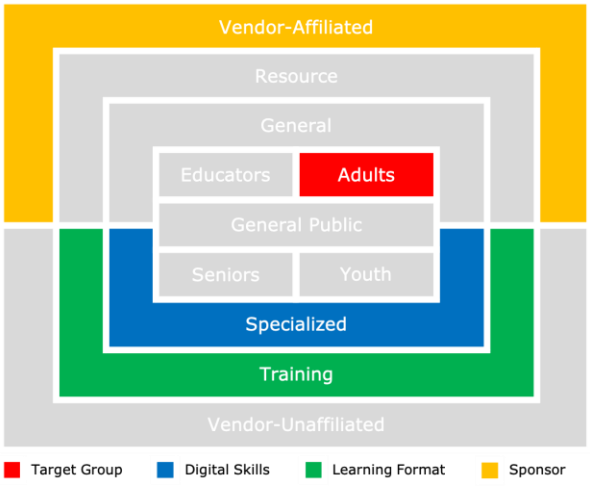


Figure 4: Depiction of Archetype 3

An illustrative example of a digital skills initiative in archetype 3 is *Amazon Web Services (AWS) re:Start*. *AWS re:Start* targets adults in the UK who are under the age of 30 and not in employment, education, or training (so called NEETs), as well as military veterans, members of the military reserve, those leaving the Armed Forces, and service spouses in the UK. Thus, the target group of *AWS re:Start* is adults.

AWS re:Start teaches participants how to architect, design, and develop cloud-based applications using AWS. Specifically, participants learn about agile and software modelling techniques, multi-tier architectures, application programming interfaces, and micro services. The curriculum has been designed to accommodate participants without any prior technical knowledge. After completing *AWS re:Start*, participants are eligible for technical positions such as help desk support, IT support analyst, software developer, IT support technician, network engineer, IT recruitment consultant, and IT sales roles. Thus, the digital skills provided by *AWS re:Start* are specialized in nature.

AWS re:Start consists of a four-week technical training provided in-person. Before beginning the training, potential participants attend a taster day at Amazon offices, where they learn about the program and can assess their desire to work in the industry. After completing the four-week technical training, participants gain practical experience during three months of on-the-job training with one of over 100 AWS customers and partner organizations. Finally, participants are given employability workshops to help them secure full-time positions – potentially with the organization that provided them the on-the-job training. Thus, the learning format offered by *AWS re:Start* is training.

AWS re:Start is sponsored by Amazon as part of its effort to bridge the digital skills gap in Europe. Thus, the sponsor of *AWS re:Start* is vendor-affiliated.

Archetype 4: Connect Seniors to the Digital World (Germany, Lithuania, Romania, Spain)
The fourth archetype represents digital skills initiatives that provide seniors with general digital skills by offering training while being sponsored by a vendor-unaffiliated organization. Figure 5 depicts archetype 4.

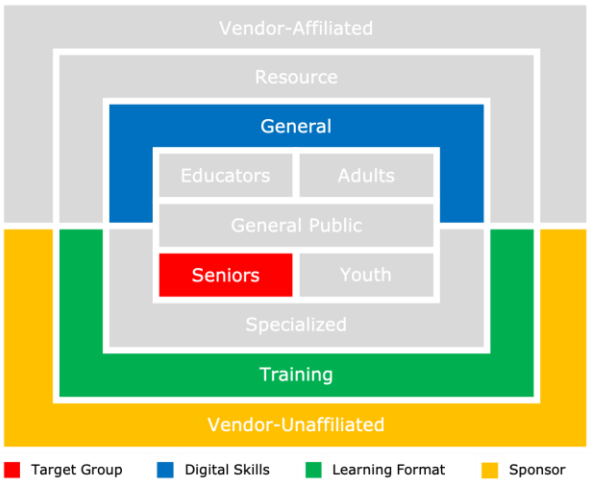


Figure 5: Depiction of Archetype 4

An illustrative example of a digital skills initiative in archetype 4 is *Connect Seniors to the Digital World* (CSDW). *CSDW* is a digital skills initiative that targets senior citizens living in retirement homes or frequenting libraries in parts of Germany, Lithuania, Romania, and Spain. Thus, the target group of *CSDW* is seniors.

CSDW teaches seniors to use tablet computers to become digitally active citizens. This includes such tasks as using video conferencing software to speak with family (for example Skype), scanning QR codes in museums, and borrowing ebooks from the library. Thus, the digital skills provided by *CSDW* are general in nature.

CSDW consists of a two-step process. In a first step, *CSDW* provides an online course to so-called multipliers, who are individuals wanting to become trainers, such as social workers or staff members in libraries or retirement homes. The online course is self-paced and consists of technical instructions on how to use a tablet computer (including handouts to provide to seniors) as well as pedagogical methods for training seniors. It is assumed that potential multipliers are proficient in using personal computers. To support adoption and localization in various countries, the online course has been developed in English. The multipliers then provide in-person training to seniors in libraries and retirement homes. Thus, the learning format provided by *CSDW* is training.

CSDW is sponsored by the Digital Opportunities Foundation, a German nonprofit organization that researches the social consequences of digitization, advocates for equal access to the Internet for all people, and promotes digital skills. The Digital Opportunities Foundation is under the patronage of the German Federal Ministry of Economics and Energy and the German Federal Ministry for Family Affairs, Senior Citizens, Women and Youth. Thus, *CSDW* is sponsored by a vendor-unaffiliated organization.

Archetype 5: Code+Create (Greece)

The fifth archetype represents digital skills initiatives that provide youth with specialized digital skills by offering training while being sponsored by a vendor-unaffiliated organization. Figure 6 depicts archetype 5.

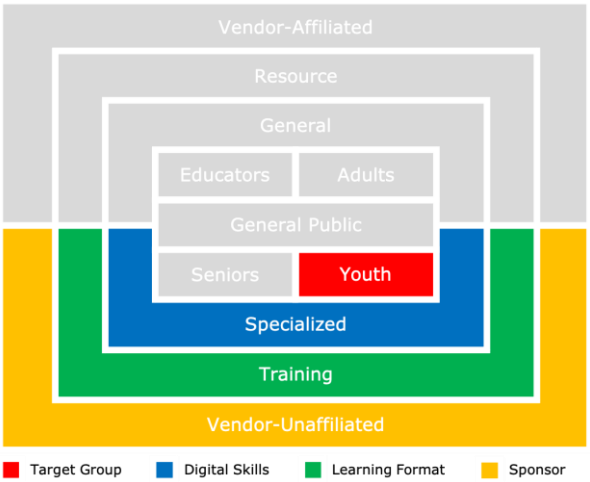


Figure 6: Depiction of Archetype 5.

An illustrative example of a digital skills initiative in archetype 5 is *Code+Create*. *Code+Create* is a digital skills initiative that targets Greek adolescents and adolescent refugees. Thus, *Code+Create* targets youth.

Code+Create teaches web development (using HTML, CSS, JS, Git, Node, Angular, Bootstrap), 3D printing, robotics, Internet of Things (IoT) applications, Python programming, and office automation using LibreOffice. Thus, *Code+Create* teaches digital skills that are specialized in nature.

Code+Create operates two learning spaces in Athens which are fully equipped with tech tools such as Raspberry Pis, NodeMCU IoT kits, Edison robots, Ultimaker 3D printers, projectors, and Linux laptops for participants. Classes last three hours and take place in-person two to three times a week. The program's duration is eight weeks. Classes are held weekdays and weekends. Every class is run by two instructors with the aid of a teaching assistant. Instruction and all instructional materials are in English. Moreover, all instructional materials are openly licensed and

made available online for free. Thus, the learning format offered by *Code+Create* is training.

Code+Create is sponsored by The Organization of Open Technologies in Greece, which aims to promote the development of open standards, free software, open content, open data, and open architecture in the areas of education, the public sector, business, and the economy in Greece. Thus, *Code+Create* is sponsored by a vendor-unaffiliated organization.

6. DISCUSSION

This study aims to answer the question "what categories can be used to classify digital skills initiatives?". To this end, a typology consisting of four categories (i.e. target group, digital skills, learning format, and sponsor) has been proposed and five archetypes, or typical examples, of digital skills initiatives were presented.

As the five archetypes have shown, there exists a great variety of different digital skills initiatives across a range of countries. As mentioned previously, a total of 40 different combinations of target group, digital skills, learning format, and sponsor are theoretically possible. However, a closer analysis of the archetype for each target group reveals that certain combinations are more likely to occur in practice.

For example, as shown in archetype 1, the general public can be relatively easily introduced to general digital skills using a resource – an online game, as in the example of *Digital Duel*. While it is certainly possible to provide the general public with specialized digital skills, it seems likely that there is a lesser need for specialized digital skills in the general public than in other target groups, such as adults or youth. The same applies to archetype 4, which targets seniors. General digital skills could be more needed than specialized digital skills among seniors. However, in contrast to seniors, who are probably more prone to respond to training than a resource, the general public can most likely be approached with a resource and training alike.

Similarly, archetypes 2, 3, and 5, which target educators, adults, and youth, respectively, provide specialized digital skills using training. There should be additional digital skills initiatives available for each of the three target groups to provide them with general digital skills, as needed. This could probably be accomplished equally well through training or a resource. Given the potential for demand generation in the three target groups, it seems more likely to find vendor-affiliated organizations willing to sponsor digital skills initiatives for these target groups than for the general public or seniors.

In summary, the general public and seniors appear to lend themselves to be taught general digital skills using training and/or learning resources from a vendor-unaffiliated sponsoring organization. In contrast, educators, adults, and youth are more prone to be taught specialized digital skills using training and/or learning resources from a vendor-affiliated sponsoring organization.

Contributions

The contribution of this research to practice and theory is three-fold. First, the proposed typology provides a common reference for researchers, practitioners, and policymakers when talking about digital skills initiatives. Up until this point, there has been no systematic effort to classify digital skills initiatives. The present work provides a valuable tool that can be easily used to classify digital skills initiatives and thereby make it clearer what one means when they talk about digital skills initiatives.

Second, the typology can be leveraged to encourage the development of new digital skills initiatives in so-called white spots. In this context, white spots are combinations of the categories in the typology that are currently missing from the landscape of digital skills initiatives in a given country or region. Policymakers could use the typology to identify such white spots and encourage the development of new digital skills initiatives by providing funding or other interventions.

Third, the typology lends itself to be used for the systematic evaluation of existing digital skills initiatives. One of the key prerequisites to a proper evaluation study is the definition of the scope of interventions to evaluate. The typology can be easily used to define what types of digital skills initiatives to include in a particular evaluation study. This, in turn, will enable researchers and policymakers to conduct evaluation studies and ultimately increase the understanding of 'what works when' in relation to digital skills initiatives.

Limitations

Despite its significant contributions, the present study is not without limitations. Specifically, the typology was developed based on an iterative content analysis conducted by the two authors. It is possible that another typology with different categories could be developed using a different

methodology. Likewise, the typology and its archetypes were developed based on an analysis of the over 300 initiatives listed in the repository of the European Commission (n.d.). While this should provide a good basis for insights into digital skills initiatives in Europe, it is also a shortcoming as there might be other digital skills initiatives in Europe or in other regions with different political and socio-economic environments that would have led to the development of different categories and possibly a different typology.

Future Research

Future research may wish to repeat the present study using the same data but a different methodology in order to establish whether or how a different typology might emerge. In addition, future research could identify other repositories of digital skills initiatives, preferably in countries that are different to the EU, and use these digital skills initiatives to refine or extend the typology and its archetypes. Lastly, the proposed typology should be used to guide the development of new digital skills initiatives or evaluation studies and thereby assess the usefulness of the proposed typology in the field.

7. CONCLUSION

The growing digital skills gap negatively affects all industries, workers, and societies. In response, hundreds of different digital skills initiatives have been launched in Europe in the past five years alone. The present study is the first to propose a typology of digital skills initiatives, which has been developed based on an analysis of over 300 digital skills initiatives listed in a comprehensive repository of the best digital skills initiatives as maintained by the European Commission. The proposed typology consists of four categories: target group (general public, educators, adults, seniors, youth), digital skills (general vs. specialized), learning format (training vs. resource), and sponsor (vendor-affiliated vs. vendor-unaffiliated). A subsequent analysis of five archetypes of the typology revealed that certain combinations of the four categories might be more likely to occur in practice. The proposed typology can be used by researchers and policymakers as a common reference when evaluating existing or developing new digital skills initiatives.

8. REFERENCES

Agrawal, S., De Smet, A., Lacroix, S., & Reich, A. (2020). To emerge stronger from the COVID-19 crisis, companies should start reskilling their workforces now. *McKinsey Insights (Issue May)*.

Bailey, K. D. (1994). *Typologies and Taxonomies: An Introduction to Classification Techniques.* Thousand Oaks, CA: Sage.

Bawden, D. (2008). Origins and Concepts of Digital Literacy. In C. Lankshear (Ed.), Digital literacies: concepts, policies and practices (17-32). New York, NY

Blažic, B. J., & Blažic, A. J. (2020). Overcoming the Digital Divide with a Modern Approach to Learning Digital Skills for the Elderly Adults. *Education and Information Technologies*, *25*(1), 259-279.

Broadband Commission for Sustainable Development. (2017). Working Group on Education: Digital skills for life and work.

Bughin, J., Staun, J., Andersen, J., Schultze-Nielsen, M., Aagaard, P., & Enggaard, T. (2017). Digitally-enabled automation and artificial intelligence: Shaping the future of work in Europe's digital front-runners.

Calvani, A., Cartelli, A., Fini, A., & Ranieri, M. (2008). Models and Instruments for Assessing Digital Competence at School. *Journal of E-Learning and Knowledge Society*, *4*(3), 183–193

Carlson, T. (2020). Amazon to help 29 million people around the world grow their tech skills with free cloud computing skills training by 2025. Last accessed 8/01/2021 from https://www.aboutamazon.com/news/workplace/amazon-to-help-29-million-people-around-the-world-grow-their-tech-skills-with-free-cloud-computing-skills-training-by-2025

Davenport, J. H., Crick, T., & Hourizi, R. (2020, April). The Institute of Coding: A University-Industry Collaboration to Address the UK's Digital Skills Crisis. In *2020 IEEE Global Engineering Education Conference (EDUCON)*, 1400-1408.

European Commission (2020). Digital Economy and Society Index (DESI) 2020: Full Economic Analysis.

DiMaggio, P., Hargittai, E., Celeste, C., & Shafer, S. (2004). From unequal access to differentiated use: A literature review and agenda for research on digital inequality. In

K. Neckerman (Ed.), Social inequality (pp. 355–400). New York, NY: Russell Sage Foundation

Donoso, V., Pyżalski, J., Walter, N., Retzmann, N., Iwanicka, A., d'Haenens, L., & Bartkowiak, K. (2020). Report on Interviews with Experts on Digital Skills in Schools and on the Labour Market.

European Commission (n.d.). Digital Skills & Jobs Coalition Initiatives Repository. Last accessed 5/01/2021 from https://ec.europa.eu/digital-single-market/en/digital-skills-jobs-coalition-initiatives

European Commission (2021). ICT for work: Digital skills in the workplace.

Hargittai, E., Hsieh, Y. P. (2012). Succinct survey measures of web-use skills. *Social Science Computer Review, 30*(1), 95–107.

Iordache, C., Mariën, I., & Baelden, D. (2017). Developing Digital Skills and Competences: A Quick Scan Analysis of 13 Digital Literacy Models. *Italian Journal of Sociology of Education, 9*(1), 6-30.

Nind, M., & Lewthwaite, S. (2020). A Conceptual-Empirical Typology of Social Science Research Methods Pedagogy, *Research Papers in Education, 35*(4), 467-487.

Hoberg, P., Krcmar, H., & Welz, B. (2017). Skills for digital transformation. *IDT survey.*

Leahy, D., & Wilson, D. (2014, July). Digital skills for employment. In *IFIP Conference on Information Technology in Educational Management* (pp. 178-189). Springer, Berlin, Heidelberg.

Lintzeris, P., & Karalis, T. (2020). Towards an inclusive skills typology. *European Journal of Social Sciences Studies,* 5(1).

Smith, B. (2020). Microsoft launches initiative to help 25 million people worldwide acquire the digital skills needed in a COVID-19 economy. Last accessed 8/01/2021 from https://blogs.microsoft.com/blog/2020/06/30/microsoft-launches-initiative-to-help-25-million-people-worldwide-acquire-the-digital-skills-needed-in-a-covid-19-economy/

van Deursen, A.J.A.M., Helsper, E.J., Eynon, R. (2016). Development and validation of the Internet Skills Scale (ISS). *Information, Communication & Society, 19*(6), 804–823.

van Dijk J.A.G.M., van Deursen A.J.A.M. (2014) Introduction. In: Digital Skills. Palgrave Macmillan's Digital Education and Learning. Palgrave Macmillan, New York.

van Laar, E., van Deursen, A. J., van Dijk, J. A., & de Haan, J. (2020). Determinants of 21st-century skills and 21st-century digital skills for workers: A systematic literature review. *Sage Open, 10*(1).

Vieru, D., Bourdeau, S., Bernier, A., & Yapo, S. (2015, January). Digital competence: A multi-dimensional conceptualization and a typology in an SME context. In *2015 48th Hawaii International Conference on System Sciences* (pp. 4681-4690). IEEE.

Editor's Note:

This paper was selected for inclusion in the journal as an EDSIGCON 2021 Meritorious Paper. The acceptance rate is typically 15% for this category of paper based on blind reviews from six or more peers including three or more former best papers authors who did not submit a paper in 2021.