ASEE 2022 ANNUAL CONFERENCE
Excellence Through Diversity
MINNEAPOLIS, MINNESOTA, JUNE 26TH-29TH, 2022 ASEE

Paper ID #36485

Analysis of Effect of Answering Reflection Prompts in a Computer Organization Class

Cheryl Lynn Resch (Lecturer)

Cheryl Resch is an Instructional Assistant Professor in the Engineering Education Department at the University of Florida. She teaches core Computer Science courses and Cybersecurity courses in the Computer and Information Science and Engineering Department. Ms. Resch is also a PhD student in Human Centered Computing. Ms. Resch joined University of Florida in 2017. Prior to that she spent 29 years as an engineer at the Johns Hopkins University Applied Physics Laboratory. The last 15 years of her time at APL she worked on a wide variety of cybersecurity projects. Ms. Resch has a BS and MS in Mechanical Engineering from University of Maryland and an MS in Computer Science from Johns Hopkins University.

Patriel Stapleton (Researcher)

I'm of Afro-Caribbean descent and spent most of my life raised on the beautiful islands of St. Kitts and Nevis. I am a double Gator at the University of Florida with a B.S. in Computer Science and am currently pursuing a Ph.D. in Human-Center Computing. I aim to conduct research that examines and produces technologies that support learning. This is incredibly important as the pace of technologies often advances without concern for the experiences of people of color.

Benjamin Martin Rheault (Researcher)

Amy Wu (Undergraduate Student)

Christina Gardner-McCune (Researcher)

Analysis of Effect of Answering Reflection Prompts in a Computer Organization Class

Abstract
This research paper explores whether requiring students to answer reflection prompts immediately before an exam affects their exam performance. Reflective practice encourages critical thinking and answering reflection prompts can aid students in integrating ideas and developing a holistic view of the material they are studying. In our study, 365 students in a Computer Organization course answered reflection prompts about a course module of their choice (e.g. ARM programming or pipelining). Half the students answered reflection prompts before the first exam, and the other half answered reflection prompts before the second exam. We found that most students who answered reflection prompts immediately before an exam performed significantly better than those who did not. This is a significant result because while reflective practice in Computer Science classes is discussed in the literature, quantitative analysis of performance improvement has been missing until now. A qualitative analysis of responses showed that many students reported that reflection helped them gain a holistic view or conceptual understanding, which is a characteristic of deep learning. This is a significant result given that we did not advise students on why we were asking them to reflect, or what we hoped they would gain from the activity. Our study provides quantitative evidence that answering reflection prompts correlates with improved performance on exams and qualitative evidence that students believe that reflection can help with conceptual understanding and deep learning. Reflection should be considered as a strategy for helping students think critically about course material.
Key Words: Reflective practice, Quantitative Study, Computer Organization

1 Introduction

It is important for college students to understand the "big picture" of their undergraduate curriculum and how it prepares them for their careers [1]. In Computer Science (CS), students often don't see how their "systems" classes, like Computer Organization, fit in with their programming classes. This results in decreased engagement with course content, which undermines student learning. Reflective writing is a pedagogical strategy introduced by Dewey [2] and extended by Schon [3] that has been found to increase critical thinking in students by allowing them to analyze their experiences for better understanding. Moon's [4] theory of learning and reflection in undergraduate education explores reflection as a way to guide a student from surface learning (memorizing facts) to deep learning (integrated ideas and a holistic view)[4]. Reflection that promotes deep learning can help CS students realize the value of understanding the systems they are using to write programs, retain the information they are learning, and take that information to future classes and see how it fits in the curriculum as a whole. This paper explores if and how pre-assessment reflection prompts affect the way students retain information learned in a Computer Organization class and aims to answer the following research questions:

- RQ1 - How does answering reflection prompts on course content correlate with student performance on exams?
- RQ2 - What are student perceptions of the usefulness of reflection for exam preparation?

2 Previous Work

In a review of the CS education literature, we found that reflection has been suggested as a way of enhancing problem students' solving skills and self-awareness of their software development process. Several papers pointed out that students find reflection difficult and suggest ways to scaffold the process. Prior et. al [5], Coffey and Owsnicky [6][7]and Nylen and Issomottonen [8] reported on adding reflective writing to software engineering courses and discussed how to encourage students to thoughtfully participate in the activity, such as including length requirements. Caruso et. al [9] reported on having first year CS students reflect on how they could improve their software development process. They found that the plans students created were often vague and that they need scaffolding for developing reflection skills. Kakavouli and Metaxas [10] experimented with reflection questionnaires in a CS2 class and suggested that instructors provide guided reflection prompts, such as "What did you learn?". In this way, students can take charge of their own learning and the instructor can scaffold the process with guided prompts.

Several CS education papers explored the use of guided reflection prompts to promote deeper conceptual learning and retention of course content in introductory programming courses [11], Computer Architecture, and Distributed Computing [12]. Pears and Larzon [12] describe a qualitative analysis of reflections collected from 42 students in a Computer Architecture course and 22 students in a Distributed Computing course. Students wrote a reflection about course modules using prompts such as the following: "Was there something that was especially interesting? If so: what and why?". When students were asked how they benefited from the reflection assignments, the dominant answer was that "they perceive that they have learned more by having to think about what they heard during the lecture and what they really did not understand that well" [12]. The authors concluded that the reflection prompts "encourage behaviours that have been linked to deep learning" [12].

Alzaid and Hsiao [11] describe two studies of reflection in an introductory Java course and also stress the importance of guided prompts. Students reflected on answers to optional daily quizzes to self-assess and monitor their learning progress. In their first study, students were asked to reflect on answers to quiz questions, but without any prompts to guide them. In the second study, students were given the prompt: "Can you tell us why you chose this answer?" Alzaid and Hsiao found that giving students a specific prompt resulted in more constructive reflections and improved performance on quizzes [11].

Use of reflection activities in CS education have been found to help support students with problem solving skills and retention of course material. These studies have been qualitative with a focus on students' perceived benefits of reflection, instructor noted challenges in student reflection and need for support, and guided reflection prompt examples. A gap in current CS education literature is the quantitative evaluation of reflection on students' performance.

Given the lack of empirical studies on the impact of reflection on exam performance, we reviewed quantitative studies in the STEM literature to find evidence that links answering reflection prompts to enhanced student performance on exams. Reinholz [13] compared test scores for experimental and control groups of students in different sections of a calculus course to determine if students who participated in peer assisted reflection had higher scores than the students who did not. Reflection prompts included: “Explain why, not just what”. In two experiments, they found that students who participated in peer assisted reflection scored significantly higher on exams during the semester than students who did not. This study suggests that having students reflect with prompts that ask “why” and “how” helped them do better on exams in the course.

Meneske et al.[14] compared the effect of different sets of reflection prompts on student performance on exams. An introductory engineering course with 208 students assigned one section (N=112) to respond to “generic” prompts and the other section (N=96) to respond to “specific” prompts after lectures. Generic prompts included “Describe what was confusing or needed more detail.” Specific prompts included “Considering the evidence of proficiency for the learning objectives, what is particularly difficult for you? Be specific.” They found that the section that completed the specific prompts performed significantly better on exams during the semester. The results suggest that specific prompts are more effective in enhancing student performance than generic prompts. Just asking students to “be specific” caused them to better retain course material.

Han, et al. [15] evaluated the efficacy of using guided written reflections after weekly experiments in a chemistry lab course. The study split the students into a control group (N=191) who were not required to complete a guided reflection, and an experimental group (N=191) who were required to respond to six guided reflection prompts after a lab. They found that the control and experimental groups performed similarly on the pre-tests before labs. However, the experimental group performed statistically significantly better than the control group on post-tests taken after the labs [15]. The authors conclude that the reflection activity was effective in helping students learn important aspects of experimental work and better prepare them to answer questions about their chemistry knowledge.

Overall, the quantitative analyses in mathematics [13], engineering [14] and chemistry [15] confirm that guided reflection prompts that ask students “why” and “how” can enhance performance on exams. This finding provides further support for the qualitative studies in CS education that report that students better engage with a reflection activity if they are given guided prompts [10][12] that ask students to explain why they chose a particular answer [11]. Combined, these results strengthen the evidence for the effectiveness of reflection prompts to enhance students’ content retention and exam performance. However, there is still a need to replicate the empirical study of exam performance results in CS courses.

3 Study Description

This paper aims to measure the impact of reflection on students’ exam performance in a Computer Organization course. We adopted a similar approach to that used by Reinhold [13], Han et al.[14], and Meneske et al.[15] that separated students by course section. We use the prompts from Pear and Larzon’s study[12] and do a quantitative analysis of the difference in performance on exams between students who wrote reflections immediately before exams and students who did not.

Our study was conducted at a R1 university in the southeastern United States during the Fall 2020 semester. This is a required core course for students majoring in CS and Computer

Engineering, and for students minoring in CS. This large enrollment course (n=374) consisted of 3 weekly lectures taught by the first author and one weekly discussion. The study was approved by the Institutional Review Board at our university and the instructor worked with teaching assistants to collect the student data during the term to eliminate conditions of conflict of interest.

3.1 Participants
Participants were recruited from six discussion sections enrolled in the course. Participants earned 1 percentage point of extra credit on their final grade by agreeing to participate in the study and taking a pretest. Participation was voluntary, and students could opt not to participate in the study but still receive extra credit through other opportunities. Students from three of the six discussion sections were placed into Group 1 (N=175) and students in the other three sections were placed into Group 2 (N=192). The majority of students in both groups agreed to participate in the study: 175 out of 178 students in Group 1 and 192 out of 196 students in Group 2. Table 1 summarizes the demographics of the participants.

Gender	283 male, 84 female
Class	9 soph., 108 junior, 202 senior, 47 5th year senior, 1 post bacc
Major	202 Computer Science, 85 Computer Engineering, 80 CS minor

Table 1 – Demographics of Participants

3.2 Course Context
The Computer Organization course consisted of 8 modules: 1) Introduction and Computer Performance, 2) Digital Logic, 3) ARM Programming, 4) Binary Numbers and Arithmetic, 5) Single Cycle Datapath, 6) Pipelined Datapath, 7) Memory, and 8) Parallel Processing. Two exams were given in the class: one exam on modules 1-4 and one exam on modules 5-8. Table 2 shows the number of lectures devoted to each module, the point values of exam questions for each module and the number of students who chose to reflect on each module. The exam points per module are roughly proportional to the number of lectures per module. The exam questions were mostly applied, with students solving close-ended problems. On the first exam, there was one open ended question on the use of instruction set architectures, and one open ended question describing what an ARM procedure is doing. On the second exam, there was one open ended conceptual question on why branch calculation is done early in the pipeline, one open ended conceptual question on cache design, and one open ended conceptual question Flynn's taxonomy of parallel processing schemes.

Module	Lectures	Exam Points	N
Intro. / Computer Perf.	5	20 (10 points conceptual)	11 of 175
Digital Logic	3	5	11 of 175
ARM Programming	8	50 (10 points conceptual)	83 of 175
Numbers and Arithmetic	5	25	35 of 175
Single Cycle Datapath	3	10	11 of 190
Pipelined Datapath	8	40 (10 points conceptual)	80 of 190
Memory	8	40 (10 points conceptual)	43 of 190
Parallel Processing	3	10 (10 points conceptual)	6 of 190

Table 2 – Point Value of Exam Questions

Throughout the semester, quizzes were given to assess student understanding of the modules. Before Exam 1, there were three quizzes: one each on Digital Logic, ARM Programming, and Numbers and Arithmetic. No quiz was given on Introduction and Computer Performance because that week students had a participation activity where they identified a platform and its instruction set architecture. Before Exam 2, there were three quizzes: one each on Single Cycle Datapath, Pipelined Datapath, and Memory. No quiz was given on Parallel Processing because that topic was covered in the last week of the semester and there was no time for a quiz.

3.3 Intervention

During the first week of class, participants took a pretest with one basic question from each of the 8 modules in the class. Group 1 (N=175) was assigned to write a reflection on one of the first four modules (see Table 2) that was due the day before Exam 1. Group 2 (N=192) was assigned to write a reflection on one of the last four modules (see Table 2) due the day before Exam 2. Table 3 shows the switching-replications experimental design where O is an evaluation and X is an intervention [16]. With this design, we can analyze whether answering reflection prompts immediately before an exam correlates with better performance on the exam.

	Pretest	Reflection	Exam 1	Reflection	Exam 2
Group 1	X	O	X		X
Group 2	X		X	O	X

Table 3 – Experimental Design

For the reflection assignment, students were required to choose a course module and answer the questions given in Table 4 for that module. Prompts 1-5 are from the work by Pears and Larzon [12]. Prompt 6 was added by our research team to gain insight into students' attitudes toward the assignment.

1. Was there something that was especially interesting? If so: what and why? If not, why not?
2. Was there something that was confusing or unclear? If so: what? If not, what was the least clearly explained in what was covered?
3. What did you do to help your understanding of the confusing or unclear material?
4. Was there something that was totally irrelevant or felt meaningless? If so: what and why? If not, what was the least relevant in what was covered
5. What was the most important thing you learned and why?
6. How is preparing this assignment different from what you normally do to study for a test?

Table 4 – Reflection Prompts

We did not advise students on why they were asked to answer reflection prompts or what we hoped they would gain from the assignment.

3.4 Quantitative Analysis

We computed the mean and standard deviation of the Pretest, Exam 1 and Exam 2 scores for Group 1 and Group 2. Measures of skew and kurtosis indicated that these scores were normally distributed. Group 1 and Group 2 were formed from pre-existing sections, so are not truly independent. A scatterplot of the residuals for the Pretest, Exam 1 and Exam 2 scores shows no patterns, so they can be modeled as independent [17]. Because the appropriate assumptions are met, the difference in the means of the Pretest scores between Group 1 and Group 2 was measured with independent-samples t-test [17]. The differences in mean Exam 1 and Exam 2 scores between Group 1 and Group 2 were measured with an analysis of covariance (ANCOVA) with Pretest score as a covariate. Linearity of covariate (Pretest) versus dependent variable (Exam1 or Exam 2) were verified using scatterplots.

We then split Groups 1 and 2 into subgroups based on which module participants chose to reflect on and computed the mean and standard deviation of Exam 1 and Exam 2 scores for the subgroups. For each subgroup, we calculated the difference between that subgroup's exam scores and those for students not in that subgroup using an ANCOVA with the Pretest score as a covariate. Next, we calculated each student's score per module, e.g., how many points a student scored on exam questions related to module 1, and so on. We then calculated the average score per module for each subgroup. Skew and kurtosis values indicate a normal distribution of scores for the following modules: 1 (Introduction), 3 (ARM Programming), 4 (Numbers and Arithmetic), 6 (Pipelining) and 7 (Memory). Scatterplots of residuals for the scores for these modules indicate that independence can be assumed. Therefore, for these modules we calculated the difference between module score for the subgroup of students who reflected on that module and the module scores for the students not in that subgroup with an ANCOVA with the quiz score on that module as a covariate. The quiz score indicates the level of students' knowledge of a module before an exam and before being asked to reflect. Including the quiz score as covariate will help us determine whether differences in exam score are predicted by prior knowledge or having responded to reflection prompts. Scatterplots of the exam scores versus quiz scores for each module indicate that the relationship is linear.

3.5 Qualitative Analysis
The first five authors performed an inductive analysis of 365 student responses to Prompt 6 in Table 4. Each of the five authors read through a non-overlapping one fifth of the responses to discover inductive categories. The five authors then discussed the categories they identified and agreed that the responses generally fell into "positive", "negative" and "neutral" categories. They further developed subcategories based on common themes found within each category. A codebook with descriptions and examples was then created for raters to refer to when coding.

To establish inter-rater reliability, 25 reflection responses were then randomly chosen and the first five authors each independently coded them. The entire response was used to determine the category and subcategory. The coder first chose whether the response was "positive", "negative" or "neutral". They then chose the subcategory for the response. If multiple sentiments were expressed, the coder chose the first clear fit they found in the response. Pairwise Cohen's kappa [18] was calculated and the values ranged from fair to moderate agreement. The five authors then discussed the reflection responses for which there was disagreement, discussed the subcategory descriptions and coding process, and agreed on subcategories for those reflection responses for which there was disagreement. The five authors then independently coded another 25 randomly chosen reflection responses. Pairwise Cohen's kappa was calculated among the five raters and the average value used as the interrater reliability [19][20]. This value is 0.628 which is considered to be substantial agreement [18]. We considered this agreement to be adequate to allow the five authors to individually code the remaining reflection responses. The five authors then evenly split coding the remaining 315 reflection responses.

4 Results
4.1 Quantitative Results
Table 2 indicates how many students chose to reflect on each module along with the number of exam points for each module. Modules 2, 5, and 6 had the largest number of exam points and also the largest number of students choosing to reflect on them. In each group there were students who did not follow instructions and did not choose a particular module to reflect on. Instead, they answered the prompts on the material as a whole rather than a particular module. These students' data were included in the analysis as members of their respective groups (e.g., Group 1 or Group 2). Two students in Group 2 dropped the class before Exam 2 and did not complete a reflection. Their scores are included in the Exam 1 data. Descriptive statistics for the Pretest, Exam 1 and Exam 2 scores are shown in Table 5. There was no significant difference in the Pretest score between Group 1 and Group 2.

4.1.1 Exam 1 Results
An ANCOVA of the Exam 1 score with the Pretest score as the covariate indicated that Pretest is not a significant predictor of Exam 1 score ($F(1, 364)=0.900$, $p=0.344$). In addition, there is no significant difference in the Exam 1 scores between those who reflected before Exam 1 (Group 1) and those who did not reflect before Exam 1 (Group 2) ($F(1, 364)=0.352$, $p=0.553$).

		N	Low	High	Mean	Std Dev
Pretest	Group 1	175	0	8	3.33	1.924
	Group 2	192	0	8	3.14	2.035
Exam 1	Group 1	175	17	100	76.93	16.06
	Group 2	192	31	100	77.99	15.67
Exam 2	Group 1	174	36	100	85.00	13.08
	Group 2	190	47	100	88.15	9.743

Table 5 – Exam Scores

Table 6 shows the average score on Exam 1 (mean = 77.48) as a function of which module students chose to reflect on. Of the 175 students who reflected before Exam 1, 83 students (47.4%) reflected on module 2, ARM Programming. We compared the scores for students who reflected on module 2 with the scores of the rest of the students (N=284) with an ANCOVA with the Pretest score as covariate and found that the students who reflected on module 2 had a mean Exam 1 score that is significantly higher than that of the rest of the students ($F(1, 364)=5.758$, $p=0.017$). The ARM programming section represented 50% of the total Exam 1 score and the most lectures per module (N=8). For module 4, Numbers and Arithmetic, an ANCOVA with the Pretest score as covariate indicated that the 35 students who reflected on this module had significantly lower scores on Exam 1 than the other students ($F(1, 364)=5.927$, $p=0.015$). We looked at the distribution of scores for students who reflected on this module and did not find evidence of very low scores skewing the mean. The research team could find no factor that would correlate with the low performance by the students who reflected on Module 4.

Module	N	Mean (GM=77.48)	Std Dev
1	11	76.09	11.29
2	11	80.21	12.45
3	83	80.98	16.49
4	35	71.52	16.17
Did not choose a module	35	73.40	14.23
Did not reflect before Exam 1	192	77.99	15.67

Table 6 – Exam 1 Scores

4.1.2 Exam 2 Results

An ANCOVA of the Exam 2 score with the Pretest score as the covariate indicated that Pretest is not a significant predictor of Exam 2 score ($F(1, 364)=0.364$, $p=0.547$). The difference in Exam 2 score between those who reflected before Exam 2 (Group 2, mean=88.15) and those who did not reflect before Exam 2 (Group 1, mean=85.00) is significant ($F(1, 361)=6.721$, $p=0.01$). Those who reflected before Exam 2 scored significantly higher on the exam. For the analysis of Exam 2 scores, although Levene's test indicates that the equality of homogeneity of variance is not met [21], the number in each group is close enough that the effect of this violation is small [21][17], so the use of ANCOVA is appropriate [17].

Table 7 shows the average score on Exam 2 (mean=86.97) as a function of which module students chose to reflect on. Reflection on any of the four modules (5-8) on Exam 2 correlated with a higher average score. The students who reflected on module 8, Parallel Processing had the highest mean score on Exam 2 (92.61), but the difference between their mean score and the mean score of the rest of the students is not significant ($F(1, 361)=1.588$, $p=0.208$). The students who performed best on Exam 1, those who reflected on module 2, ARM Programming, did not perform better than other students on Exam 2. This suggests that reflecting before Exam 1 may have helped their performance on Exam 1, since their superior performance was not repeated when they did not reflect before Exam 2. The students who reflected on module 4, who performed poorly on Exam 1, actually performed better on Exam 2 than those who reflected on module 2. The research team could find no factor to help explain this result.

Module	N	Mean (GM=86.97)	Std Dev
1	11	85.65	13.151
2	10	83.55	8.172
3	83	85.46	13.868
4	35	84.36	12.713
5	11	88.54	4.677
6	80	88.74	10.196
7	43	87.32	10.499
8	6	92.61	4.182
Did not choose a module	85	86.23	11.258

Table 7 – Exam 2 Scores

When analyzing scores for exam questions on modules 1, 3, 4, 6 and 7, the data passed the assumptions required for ANCOVA (independence, normality, homogeneity of variance, linearity of covariate and dependent variable) [17]; the results of the ANCOVA tests are shown in Table 8. The difference in module score for students who reflected on module 3, ARM Programming, and module 6, Pipelined Datapath, is significantly higher for students who reflected on those modules.

Module	Points	Mean – students who reflected on module	Mean – students who did not reflect on module	F
1	20	16.09 (N=11)	15.36 (N=356)	$F(1,364)=0.035$, $p=0.852$
2	5	4.50 (N=11)	4.49 (N=356)	N/A
3	50	40.47 (N=83)	37.92 (N=284)	$F(1,364)=3.714$, $p=0.050$
4	20	17.20 (N=35)	19.05 (N=332)	$F(1,364)=3.335$, $p=0.069$
5	10	7.93 (N=11)	7.36 (N=353)	N/A
6	40	34.97 (N=80)	33.10 (N=284)	$F(1,161)=7.643$, $p=0.006$
7	40	37.60 (N=43)	35.29 (N=321)	$F(1,161)=2.590$, $p=0.108$
8	10	10 (N=6)	9.65 (N=358)	N/A

Table 8 - Scores on Portion of Exam Students Reflected On

Summarizing key findings:

- The mean Exam 1 score for the students who reflected before the exam is not significantly different from the mean score for those who did not reflect before Exam 1
- The mean Exam 1 score for 83 students in Group 1 who reflected on ARM Programming is significantly higher than the mean score for all other students. The same group of students did not have a higher mean score than the rest of the students on Exam 2.
- The mean Exam 2 score for the students who reflected immediately before Exam 2 is significantly higher than that of students who did not reflect immediately before Exam 2.
- The mean score on exam questions for two modules that were heavily weighted in lectures and exam questions, ARM Programming and Pipelined Datapath, is significantly higher for students who reflected on those modules compared to students who did not reflect on those modules.

In answer to RQ1 - "How does answering reflection prompts on course content correlate with student performance on exams?", the results for Exam 1 indicate that the correlation of reflection and exam performance was mixed. Students that chose the most heavily weighted module on the exam performed significantly better on the exam. However, students who reflected on a less heavily weighted module before Exam 1 performed significantly worse on the exam. The results for Exam 2 more clearly show students who reflected immediately the exam performed better. This result provides evidence that having students respond to guided reflection prompts immediately before an exam correlates with improved performance on the exam. The effect is more apparent when students reflect on material that is more heavily represented in the number of lectures and number of points on exams devoted to it. Overall, this study provides empirical evidence linking reflection to improved exam performance with caveats in the context of a Computer Organization course.

4.2 Qualitative Results

To answer RQ2 - "What are student perceptions of the usefulness of reflection for exam preparation?", we inductively coded the responses to the reflection prompt: "How is preparing this reflection different from what you normally do to prepare for an exam?" Even though the question did not ask for positive or negative impressions, we found that the responses contained positive, negative, or neutral sentiments. We categorized 46% of the responses as positive, 42% as neutral, and 12% as negative. We then subcategorized the responses according to themes that best characterized students' answers to the reflection prompt (e.g., positive - reflection helps with forming a holistic view). Table 9 shows the subcategories inductively identified from students' responses. For each subcategory, the table provides an example and the total number of responses.

While the students were not advised that the reflection was intended to help their exam performance, almost half the students (46%) recognized that the activity was helpful for studying (i.e., responses coded as positive). These response subcategories were 2nd and 3rd most common response subcategories: Positive - reflection helps with forming a holistic view (27%) and Positive - reflection helps me figure out what to study (19%). However, the most common response subcategory (28%) was a neutral response describing the difference between the reflection activity and how the student typically studied for exams, characterized by practicing problem-solving, which was also echoed in the negative responses.

The students who positively viewed reflection featured two characteristics of deep learning associated with Moon [4]: “making meaning” and “transformational learning”. A holistic view or conceptual understanding are characteristics of transformative learning [4], indicating that many students found value in using reflection to improve their understanding and value of the content for their future careers beyond an exam. Students' use of reflection to determine what to study is suggestive of Moon’s “making meaning”. For these students, reflection promoted self-awareness and metacognitive evaluation of their understanding of concepts and helped the student tailor their studying to specific concepts where they lacked confidence. These are significant results given that we did not instruct students on why we were asking them to reflect, or what we hoped they would gain from the activity.

Our experience has been that many students do not see how the topics in the Computer Organization class fit into their CS studies as a whole and struggle with some of the content. These results suggest that reflection assignments may be a pedagogical tool that can help students understand how the topics connect to the overall curriculum. Asking students to reflect on course material with specific reflection prompts that ask students to explain why they chose an answer can result in better retention of course material and help them to gain a holistic understanding of the course material. These results also provide additional evidence to support the claims made by Pears and Larzon that reflection promotes deep learning [12].

However, it is important to note that not all students valued reflection or saw it as a means for improving their exam preparation. Namely, some students saw reflection as only improving conceptual understanding and not contributing to their typical approach to studying, which focused on practicing their problem-solving skills. Future studies should explore the use of reflection prompts specifically focused on helping students enhance their problem-solving skills similar to those used by Meneske et al. [14], Reinholz [13] and Han [15].

4.3 Threats to Validity

There are several internal validity threats of the quantitative results due to students’ selection of modules to study. Students chose which module to reflect on and may have chosen their “favorite” module. Thus, they could have done well on that portion of the test because they enjoyed the material in the module, in addition to having reflected on it. This threat was mitigated by using the quiz scores as a covariate which showed that students performed better on the exam even after correcting for the effect of the quiz. Next, students could have been pragmatic in their approach to reflection and simply recognized which material is most emphasized by the number of lectures and exam study guides. Thus, the reflection activity possibly enhanced students' natural ability to make wise choices which could mean that they are the type of students who would score better on the exam. Future work will explore assigning students a module to reflect on, and evenly distributing the assignment of students to modules to reflect on.

As stated earlier, the assignment to groups was not random, students were assigned to groups according to their section to aid in grading. Because assignment was not random, no causal inferences can be made.

Category Description	Example	N (out of 365)
Positive - reflection helps with forming a holistic view	I usually would just review my notes and material, but this helped me place more applicable value on certain concepts and look at the material from a view beyond just this exam. I feel that I will remember certain things more now that I've elaborated on their importance.	98
Positive – reflection helps me figure out what to study	I think this reflection allows us to re-read the content in the section we chose to reflect on and further prepare ourselves for the exam. It's also different because it lets the student think about what topics they are not too confident about and work on them.	69
Neutral - gave a description of how they study	For this reflection I have mostly been using what I remember from watching the videos and doing the exercises. When studying for an exam I actually go to those resources.	104
Neutral - not like doing practice problems	I also usually do a lot of practice problems when preparing for an exam, but this assignment was more writing about what I had learned than actually practicing	31
Neutral - not as in depth as studying	When I planned my answers for this reflection, I looked over my notes from the semester as I was reading the questions. However, I wasn't diving deep into the material as I would be if I was studying for the exam	19
Negative - reflection is an opinion	This is far less informative regarding understanding the material itself. It is a review of my opinion of the materials' worth rather than a review of the material.	19
Negative - felt negative toward the assignment	This reflection is very different from normal studying because I think more about my feelings and personal experiences with the module. Personally, this does not help with my studying for an exam because it feels more like doing work rather than improving my understanding of the material	25

Table 9 - Qualitative Category Descriptions and Examples

5. Conclusion

This paper replicates and extends prior qualitative research in CS education and quantitative research in STEM that suggests that students benefit from reflection activities. Our analysis provides statistically significant evidence that students who answer guided reflection prompts score better on exams, especially if they reflected on material that represented a large portion of the exam. Our qualitative analysis of student responses is encouraging and indicates that guided reflection prompts are a pedagogical tool that can be used to help students retain course material, and gain a holistic understanding of the course material in the context of a Computer Organization course.

References

[1] ACM. [n.d.]. *Curriculum Guidelines for Undergraduate Degree Programs in Computer Science 2013*. Retrieved April 3, 2021 from https://www.acm.org/binaries/content/assets/education/cs2013_web_final.pdf

[2] John Dewey, *How we think: A restatement of the relation of reflective thinking to the educative process.* DC Health, 1933.

[3] Donald A. Schon, *The reflective practitioner: How professionals think in action.* Basic books, 1984

[4] Jenny Moon, "PDP working paper 4: Reflection in higher education learning*,"* Higher Education Academy, 2004.

[5] Julia Prior, Samuel Ferguson, and John Leaney, "Reflection is Hard: Teaching and Learning Reflective Practice in a Software Studio," in *Proceedings of the Australasian Computer Science Week Multiconference (Canberra, Australia) (ACSW '16,.* Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/2843043.2843346, 2016.

[6] John W. Coffey, "A Study of the Use of a Reflective Activity to Improve Students' Software Design Capabilities," in *Proceedings of the 2017 ACM SIGCSE Technical Symposium on Computer Science Education* (Seattle, Washington, USA) (SIGCSE '17). Association for Computing Machinery, New York, NY, USA. https://doi.org/10.1145/3017680.3017770, 2017. pp. 129-134.

[7] John W. Coffey and Bernd Owsnicki, "Introducing a Reflective Activity into the Design Process in an Advanced Computer Programming Course*," Journal of Computing Sciences in Colleges*, vol. 31 no. 5, pp. 29–37, 2016.

[8] Aletta Nylén and Ville Isomöttönen. "Exploring the Critical Incident Technique to Encourage Reflection during Project-Based Learning" in *Koli Calling '17*, Association for Computing Machinery, New York, NY, USA, 88–97. https://doi.org/10.1145/3141880.3141899, 2017.

[9] Tammy VanDeGrift, Tamara Caruso, Natalie Hill, and Beth Simon. "Experience Report: Getting Novice Programmers to THINK about Improving Their Software Development Process," In *Proceedings of the 42nd ACM Technical Symposium on Computer Science Education* (Dallas, TX, USA) (SIGCSE'11). Association for Computing Machinery, New York, NY, USA https://doi.org/10.1145/1953163.1953307, 2011, pp. , 493–498.

[10] Stella Kakavouli and Panagiotis Metaxas. “Also "Your" Job to Learn! Helping Students to Reflect on Their Learning Progress,” *Journal of Computing Sciences in Coll*eges vol 27, no 6, 113–120, June 2012

[11] M. Alzaid and I. H. Hsiao, “Effectiveness of Reflection on Programming Problem Solving Self-Assessments,” In *2018 IEEE Frontiers in Education Conference (FIE)*. IEEE, 1–5, 2018.

[12] Arnold Pears and Lars-Åke Larzon, “Student Perceptions of Reflections as an Aid to Learning,” In *Proceedings of the 6th Baltic Sea Conference on Computing Education Research: Koli Calling 2006* (Uppsala, Sweden) (Baltic Sea ’06). Association for Computing Machinery, New York, NY, USA. https: //doi.org/10.1145/1315803.1315812, 2006, 38–45.

[13] D. L. Reinholz, “Developing mathematical practices through reflection cycles,” *Mathematics Education Research Journal* vol 28, no 3 2016, 441–451.

[14] M Meneske, S. Anwar, and Z.G. Akdemir, “How Do Different Reflection Prompts Affect Engineering Students’ Academic Performance and Engagement?” in *The Journal of Experimental Education* vol 1 no 19, 2020.

[15] Ng Sook Han, Ho Ket Li, Lee Choy Sin, and Keng Pei Sin, “The evaluation of students’ written reflection on the learning of general chemistry lab experiment,” in *MOJES: Malaysian Online Journal of Educational Sciences* vol 2 no 4, 2014, pp 45–52.

[16] R. B. Kline, *Becoming a behavioral science researcher: A guide to producing research that matters.* Guilford Press, 2008

[17] Debbi L. Hahs-Vaughn and Richard G. Lomax, *An introduction to statistical concepts.* Routledge, 2020.

[18] Jacob Cohen, *Statistical power analysis for the behavioral sciences*. Academic press, 2013

[19] Conger, A. J., “Integration and generalization of kappas for multiple raters,” in *Psychological Bulletin*, vol 88 no 2, 1980, p 322.

[20] Berry, K. J., & Mielke Jr, P. W., “A generalization of Cohen's kappa agreement measure to interval measurement and multiple raters,” in *Educational and Psychological Measurement*, vol 48 no 4, 1988, 921-933.

[21] Howard Levene, *Contributions to Probability and Statistics: Essays in Honor of Harold Hotelling.* Stanford University Press, 1960, pp. 278–292.