SHOWING PROGRAMMERS THAT THEY ALREADY KNOW HOW TO WRITE WELL

A Thesis

presented to

the Faculty of California Polytechnic State University,

San Luis Obispo

In Partial Fulfillment

of the Requirements for the Degree

Master of Science in Computer Science

by

Alanna Buss

March 2018
COMMITTEE MEMBERSHIP

TITLE: 
Showing Programmers that They Already Know How to Write Well

AUTHOR: 
Alanna Buss

DATE SUBMITTED: 
March 2018

COMMITTEE CHAIR: 
Clark Turner, Ph.D.
Professor of Computer Science

COMMITTEE MEMBER: 
Fooad Khosmood, Ph.D.
Associate Professor of Computer Science

COMMITTEE MEMBER: 
Phillip Nico, Ph.D.
Professor of Computer Science
ABSTRACT

Showing Programmers that They Already Know How to Write Well

Alanna Buss

Being able to communicate effectively through writing is an important skill for computer scientists and software engineers. In addition to being able to write code, programmers need to write clearly and concisely for everything from comments in their code to documentation. Despite writing being essential to their future careers, many programming students from Cal Poly find writing to be unimportant and many of them believe that they are not capable of writing good papers. How can Cal Poly help these students?

This paper describes a pedagogy to help students improve their writing skills. It is based heavily on the belief that programmers already possess many of the skills necessary to be good writers. The pedagogy hopes to show students how to utilize the skills from writing code in their paper writing.

This paper answers the question “Can transferring programming knowledge to writing help improve student confidence or writing ability?”. Results from the study show that the pedagogy can be associated with an improvement in students’ confidence in their ability to write logical, well-organized papers. It also show that the pedagogy can be associated with students being able to split their paragraphs into single topics more effectively.
ACKNOWLEDGMENTS

Thanks to:

- Leanne Fiorentino for always being there for me and all of the computer science department. I genuinely could not have done it without you.

- Michael Williams for always putting up with me and my antics.

- My sister Jessica for saving the day with her stats knowledge and help.

- Foaad Khosmood for literally saving the day with his help and support at the end of this process.

- Jason Peters and Chelsea Milbourne for helping me get a grasp on how transfer works.

- Andrew Guenther and all other contributors, for uploading this template [3]
# TABLE OF CONTENTS

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>LIST OF TABLES</td>
<td>ix</td>
</tr>
<tr>
<td>LIST OF FIGURES</td>
<td>x</td>
</tr>
<tr>
<td>CHAPTER</td>
<td></td>
</tr>
<tr>
<td>1 INTRODUCTION</td>
<td>1</td>
</tr>
<tr>
<td>1.1 The Problem</td>
<td>1</td>
</tr>
<tr>
<td>1.2 The Solution</td>
<td>4</td>
</tr>
<tr>
<td>1.3 Thesis Outline</td>
<td>5</td>
</tr>
<tr>
<td>2 BACKGROUND</td>
<td>7</td>
</tr>
<tr>
<td>2.1 CSC 300 Professional Responsibilities</td>
<td>7</td>
</tr>
<tr>
<td>2.2 Knowledge Transfer</td>
<td>8</td>
</tr>
<tr>
<td>2.2.1 Where and How is Transfer Studied</td>
<td>9</td>
</tr>
<tr>
<td>2.2.2 Types of Transfer</td>
<td>9</td>
</tr>
<tr>
<td>2.2.2.1 Near Transfer</td>
<td>10</td>
</tr>
<tr>
<td>2.2.2.2 Far Transfer</td>
<td>10</td>
</tr>
<tr>
<td>2.2.3 How Transfer Relates</td>
<td>10</td>
</tr>
<tr>
<td>3 RELATED WORK</td>
<td>12</td>
</tr>
<tr>
<td>3.1 How Writing is Being Taught to Programmers</td>
<td>12</td>
</tr>
<tr>
<td>3.1.1 Single Class Focused on Writing</td>
<td>12</td>
</tr>
<tr>
<td>3.1.1.1 U.C. Irvine</td>
<td>12</td>
</tr>
<tr>
<td>3.1.1.2 University of Texas at Austin</td>
<td>13</td>
</tr>
<tr>
<td>3.1.2 Multiple Technical Courses with Writing Assignments</td>
<td>13</td>
</tr>
<tr>
<td>3.1.2.1 University of Delaware</td>
<td>13</td>
</tr>
<tr>
<td>3.1.2.2 Carnegie Mellon University Qatar Campus</td>
<td>14</td>
</tr>
<tr>
<td>3.1.3 Strategies</td>
<td>14</td>
</tr>
<tr>
<td>3.1.3.1 Muhlenberg College</td>
<td>14</td>
</tr>
<tr>
<td>3.1.3.2 University of the Balearic Islands</td>
<td>15</td>
</tr>
<tr>
<td>3.1.4 Comparison to this Approach</td>
<td>16</td>
</tr>
<tr>
<td>3.2 Knowledge Transfer’s Applications</td>
<td>16</td>
</tr>
</tbody>
</table>
# LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Comparison of existing approaches and the pedagogy</td>
<td>17</td>
</tr>
<tr>
<td>4.1</td>
<td>Table of when lessons were taught and the concepts transferred in those lessons</td>
<td>31</td>
</tr>
<tr>
<td>6.1</td>
<td>Section one’s paragraph error rates for the half-length draft and final paper. n = 17</td>
<td>45</td>
</tr>
<tr>
<td>6.2</td>
<td>Section two’s paragraph error rates for the half-length draft and final paper. n = 21</td>
<td>46</td>
</tr>
<tr>
<td>6.3</td>
<td>Summary of the two section’s average error rate for both half-length draft and the final paper. n = 38</td>
<td>47</td>
</tr>
<tr>
<td>6.4</td>
<td>Section one lab participation by student. A dot signifies that that student submitted that lab. A line emphasizes that a student has done consecutive labs. n = 23</td>
<td>50</td>
</tr>
<tr>
<td>6.5</td>
<td>Section one lab participation by student. A dot signifies that that student submitted that lab. A line emphasizes that a student has done consecutive labs. n = 27</td>
<td>51</td>
</tr>
<tr>
<td>6.6</td>
<td>Number of total students who did each lab. n = 50</td>
<td>52</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>1.1</td>
<td>How important students think writing is on a scale from 1 - 10. n = 58</td>
<td></td>
</tr>
<tr>
<td>1.2</td>
<td>How good students think they are at writing on a scale from 1 - 10. n = 58</td>
<td></td>
</tr>
<tr>
<td>6.1</td>
<td>Student confidence in writing with the exit survey on the x-axis and the introduction survey on the y-axis. The gray line shows no change in confidence. Every point below the line shows an increase in confidence after the lessons and every point above the line shows a decrease in confidence after the lessons. The opacity indicates multiple points plotted on the same location. n = 35</td>
<td></td>
</tr>
<tr>
<td>6.2</td>
<td>Student belief that writing is important in the field with the exit survey on the x-axis and the introduction survey on the y-axis. The gray line shows no change in importance. Every point below the line shows an increase in importance after the lessons and every point above the line shows a decrease in importance after the lessons. The opacity indicates multiple points plotted on the same location. n = 35</td>
<td></td>
</tr>
<tr>
<td>6.3</td>
<td>Student confidence in their writing ability with the exit survey on the x-axis and the introduction survey on the y-axis. The gray line shows no change in confidence. Every point below the line shows an increase in confidence after the lessons and every point above the line shows a decrease in confidence after the lessons. The opacity indicates multiple points plotted on the same location. n = 35</td>
<td></td>
</tr>
<tr>
<td>6.4</td>
<td>Student confidence in their ability to write logical, well-organized papers with the exit survey on the x-axis and the introduction survey on the y-axis. The gray line shows no change in confidence. Every point below the line shows an increase in confidence after the lessons and every point above the line shows a decrease in confidence after the lessons. The opacity indicates multiple points plotted on the same location. n = 35</td>
<td></td>
</tr>
<tr>
<td>6.5</td>
<td>Change in error rate by class with the final paper’s error rate on the x-axis and the half-length draft’s error rate on the y-axis. The gray line shows no change in error rate. Every point above the line shows a decrease in error rate and every point below the line shows an increase in error rate. The blue points are for section 1 and the orange points are for section 3. n = 38</td>
<td></td>
</tr>
</tbody>
</table>
6.6 Students’ opinions on which lessons were the best, worst, and most helpful. $n = 11$
Chapter 1

INTRODUCTION

1.1 The Problem

Being able to communicate effectively through writing is an important skill for computer scientists and software engineers. In addition to being able to write code, programmers need to write clearly and concisely for everything from comments in their code to documentation. Writing is so important that Dr. Theresa Beaubouef, a computer science professor at Southeastern Louisiana University, goes as far as to say that an individual’s ability to write and communicate effectively will determine that individual’s career advancement opportunities.[9] She also explains how “Job candidates with spelling and grammatical errors on their resumes and cover letters are often perceived as careless at best, undereducated at worst.”[9]

Writing is also crucial for universities to obtain ABET accreditation. ABET is an organization that accredits university programs such as computer science and software engineering. According to their website, “ABET accreditation provides assurance that a college or university program meets the quality standards of the profession for which that program prepares graduates.”[4] Criteria on their general student outcomes states students must have “an ability to communicate effectively.”[4] This means that the ABET association, along with the 3,852 programs that have their accreditation, believe communication skills, such as writing, to be an important skill for students to master before joining the workforce.[4]

After joining the workforce, some programmers find themselves surprised at how much writing is required of them. An anecdote from Robert Dugan, a computer science professor at Stonehill College, explains how during an internship at IBM he
was surprised to learn about the importance of writing within the field. In this anecdote, he explains how during the internship, he spent twice as much time writing a manual for a piece of code he had written, than writing the code itself. After being hired by IBM, Dugan found that the amount of writing that was previously required of him wasn’t simply busy work relegated to an intern, it was a core part of his career.

Despite writing being so critical to the field, many computer science and software engineering students find communication skills, like writing, unimportant. In two classes, during the Spring 2017 quarter, a survey shows that at California Polytechnic State University, San Luis Obispo (Cal Poly), around a third of programming students don’t see writing as an important skill for their future. Figure 1.1 shows the result of asking these students, “On a scale from 1 - 10 where 1 is unimportant and 10 is absolutely crucial, how important do you feel writing is to success in your field?”.

![How Important Students Think Writing Is](image)

**Figure 1.1: How important students think writing is on a scale from 1 - 10. n = 58**

Even if students believe that writing is important, many of them lack confidence in their writing ability. Confidence is a good indicator of academic ability, so low
confidence in their skills is also concerning. Figure 1.2 shows the results of asking the same students, “How confident are you in your overall writing ability?” From this graph, it is evident that while most students are fairly confident in their writing ability, there are many students that are not.

![How Good Students Think They are at Writing](image)

**Figure 1.2: How good students think they are at writing on a scale from 1 - 10. n = 58**

From Figures 1.1 and 1.2, it is important to note that there is a good portion of students who do think writing is important and feel that they are good at writing. While this is good, it also shows that there is a substantial amount of students who do not. This study will focus on trying to help the students who responded with low importance or confidence, rather than reinforcing the students who responded with higher importance or confidence.

Programming students not being able to write well is a problem that must be solved by programming teachers. While English teachers can provide a base for how to write, they cannot show students why it matters for the programming field. Having teachers of technical courses emphasize the importance of writing is the only way students will take writing seriously. Therefore, if computer science and software
engineering departments continue to defer the responsibility of teaching programming students how to write, no progress will be made. While most software engineering problems are based on algorithms or code, writing is a problem that is fundamental yet neglected in the field.

Currently at Cal Poly, programming students only need to take first-year composition (FYC) and a technical writing course for their undergraduate degree requirements.[2, 1] Other than these two classes, students do not get formal teaching about writing unless they opt to take writing electives. Anecdotally, students have said that without more instruction, they believe that their writing skills will not improve, no matter how much they write. So how can Cal Poly’s computer science department help these students improve their on writing abilities?

1.2 The Solution

This work proposes a new pedagogical approach to teach writing to computer science and software engineering students. Throughout the rest of this paper writing papers and other documents will be referred to as writing and writing code will be referred to as programming or coding. The study around the pedagogy will look at student performance and confidence levels about writing. Both students’ writing ability and the way they feel about their writing are important to their writing process. This is supported by a study published in 1994, explaining that looking at the confidence levels of individuals can help be a good indicator of academic performance in areas such as writing.[32] Based on this, both students’ performance and their confidence in their abilities can help show the effects of the pedagogy and therefore both will be analyzed at the end of the study.

The pedagogy tries to help facilitate the transfer of programming knowledge to writing, to help improve student confidence about their writing, and actually improve
their writing ability. The concept of transferring knowledge is mainly used in English classes to help teach writing in different situations. The pedagogy uses the concept of transfer to help programming students use their coding knowledge to help them write better. The goal of this study is to answer, “Can transferring programming knowledge help students’ writing ability or confidence in their writing ability?”

This study has found that helping students transfer their programming knowledge to writing is beneficial for most students. Based off of setbacks that will be discussed in the results chapter, the pedagogy might not be the direct cause of the changes but it can be associated with these changes. According to the survey done at the end of the study, 89% of participating students found the activities from the pedagogy helped improve their confidence in their writing abilities. The study found that students’ confidence in their ability to write logical, well-organized papers increased with 99.5% confidence. The research also shows that students’ ability to separate paragraphs to focus on one idea improved with 99.9% confidence. These results will be discussed in more detail in the results chapter of this paper.

1.3 Thesis Outline

This section will give a brief overview of the chapters to come. Chapter 2, background, starts by explaining CSC 300, the class the pedagogy was taught in. It describes the class, the required paper, and why the students have traditionally struggled with the writing of this paper. It will then explain information about the concept of transfer.

Chapter 3, related works, starts by explaining how programmers are currently being taught writing. It will go over three approaches and explain how the pedagogy is different than the other approaches currently being used. It will then go over how the concept of transfer is being used in academia. The related works chapter concludes with how this study fits into both the programming community and the
current studies of transfer.

Chapter 4, experimental design, explains the general concepts of the pedagogy. It will explain ideal results from the study and how the pedagogy was tested for both confidence and writing improvement. It concludes with an explanation of each of the lessons. For each lesson it will explain what is being transferred and what the lab was. It finally gives an explanation of the order the lessons were taught with explanations on why each lesson was taught when it was.

Chapter 5, user study, explains some of the finer details of the user study. These details include when lessons were taught and the format of the class.

Chapter 6, results, will explain our results and their meaning for future studies. It will start by explaining setbacks that were found with the approach taken for teaching this pedagogy. Finally, it explains the results gathered about student confidence and writing ability.

Chapter 7, conclusions and future works, will explain thoughts on how to fix some of the setbacks encountered and conclude about what was found in the results.
Chapter 2
BACKGROUND

2.1 CSC 300 Professional Responsibilities

This study was done in two sections of CSC 300, Professional Responsibilities, at Cal Poly in Spring 2017. This course is required for all computer science and software engineering students as a part of their undergraduate education. The purpose of this class is to get students to think about the ethics and consequences of coding as well as help students’ oral and written communication skills. While this class encourages the improvement of these communication skills it does not teach students how to improve them.

CSC 300 is important to Cal Poly’s computer science and software engineering undergraduate degrees because it is where these degree programs fulfill, among others, ABET outcome (g) for software engineering, and ABET outcome (f) for computer science.[4] These two ABET outcomes specifically have to do with communicating effectively through writing. To help students communicate effectively through writing, the class requires students to write a four thousand word academic paper. This paper should analyze the ethics of a current event in the computer science field.

The paper for CSC 300 requires students to use the IEEE-CS/ACM Software Engineering Ethics and Professional Practices (SE Code of Ethics) to determine whether a software engineer acted ethically in a specific real-world case.[5] The course requires a research style of writing suitable for publication in a computer science or software engineering conference.\(^1\) The paper must be objective, deductive, and unambiguous with clear conclusions supported by logical analysis and evidence. This paper

\(^1\)Over the years several students have submitted their papers with minor changes to conferences and had them accepted.
is required to have the following sections: abstract, facts, focus question, social implications (Why even ask this question?), others arguments (Literature review), and analysis (Rational, logical arguments).

The facts, focus question, social implications and others arguments sections should not be biased and concentrate on the facts. The social implications section should explain the importance of analyzing the real-world case. Students should use this section to link the case to important values, for the rest of society, such as privacy, the economic value of the industry in question, and security.

In analysis section, students write about whether the software engineer acted ethically according to individual code tenets from the SE Code of Ethics. Students are required to support their analysis with facts from primary sources and formal data. The purpose of this section is not to persuade the reader of a conclusion, but to use logic to lead them from premise to conclusion. Appendix A, has the paper specifications that students are given which outline these sections in more detail.

Professor Clark S. Turner, the professor of this class at Cal Poly since 1999, explains that every computer science and software engineering student should have some experience in critical analysis and research writing in their own field. This is also important for their comprehension of other research works. Anecdotally, Professor Turner has observed that students have difficulty writing clearly and unambiguously in their own field.

2.2 Knowledge Transfer

Two of the biggest pioneers in transfer, Perkins and Salomon, explain, to transfer knowledge is to have “something learned in one context [that] has helped in another.” [34] A classic example of transfer is transferring knowledge about how to throw a baseball to learning how to throw a football. Imagine a baby who has never seen a
ball before. Now, a baseball is given to that baby and eventually, the baby learns what the baseball is and how to throw it. If the same baby is later given a football, the baby could use the knowledge of how to throw the baseball to throw the football. This, at its core, is knowledge transfer since the baby is using the knowledge it learned in one context, the baseball, and transferring it to another, the football.

Transfer is related to learning but should not be thought of as just learning an ability or skill.[35] Knowledge transfer describes the phenomena in which a person uses knowledge from one task to learn or better understand another task. In the last example, the baby learned how to throw a baseball and transferred it to better understand how to throw a football. This means for someone to successfully transfer their knowledge, they must gain an understanding of something new, by applying their understanding of something they already know.

2.2.1 Where and How is Transfer Studied

Transfer is mainly studied in the English field and is focused on how to prepare students for writing tasks in their major. Most studies focus on students in their first-year composition (FYC) courses and whether students transfer the knowledge from that class elsewhere.

In 2012 Jessie Moore surveyed how various studies attempted to promote transfer.[31] Methods for these studies included surveys, focus groups, and interviews with people involved in the research. Moore found inconsistent results with how transfer helped people and who benefited from it.[31]

2.2.2 Types of Transfer

There are two main types of transfer: near transfer and far transfer. A basic understanding of both is necessary to understand the approach of the pedagogy described
in this paper.

2.2.2.1 Near Transfer

Near transfer is the more common of the two types of transfer. It uses similar information for similar contexts.[34] The example from section 2.2 about the baseball and the football is an example of near transfer. This type of transfer can happen without recognizing that one is using previous knowledge.

2.2.2.2 Far Transfer

Unlike near transfer, which focuses on the reuse of information, far transfer focuses on adapting information.[34] An example of far transfer might be using gardening skills to help build a house. Far transfer is less common because one must consciously link the two uncommon concepts, unlike near transfer which links two similar concepts and can happen subconsciously.

One technique used to promote far transfer is called bridging.[34] Bridging is the process whereby an individual forms a connection between the different contexts. The hardest part about bridging and far transfer, in general, is that the individual must consciously abstract out the connections and apply them in context.

2.2.3 How Transfer Relates

As explained previously, students from the CSC 300 class at Cal Poly have a difficult time fully grasping how to write technical papers. The pedagogy tries to make progress on this problem by helping facilitate the transfer of students’ programming knowledge to writing. Since programming and research writing do not seem immediately related, the pedagogy will be using far transfer and bridging to try to help students use their programming knowledge in the new context of research writing.
The lessons in the pedagogy will try to help students bridge a concept from programming to a concept from writing. These concepts will be further discussed in the experimental design chapter of this paper.
Chapter 3

RELATED WORK

3.1 How Writing is Being Taught to Programmers

Our survey (depicted in figures 1.1 and 1.2) show that the two general education writing courses students have to take are not sufficient for students to write confidently for their field. This pedagogy is attempting to introduce a new way to teach programming students how to write. To understand what this pedagogy is introducing, it is important to understand how writing is currently being taught to students. There are three main approaches other universities are using to help their programming students with their writing: a single class focused on writing, multiple technical courses with writing aspects, and strategies to improve students’ writing.

3.1.1 Single Class Focused on Writing

The first approach adds a class to the curriculum that focuses on writing instead of technical skills. The rest of this section will look at some examples of universities that implement this approach.

3.1.1.1 U.C. Irvine

In 1998, David Kay wrote about a class that U.C. Irvine created. Kay explains how faculty at Irvine also see problems with their student’s poor writing ability but are hesitant to try and correct their mistakes.[21] To try and correct this, U.C. Irvine has added a class to help students with their writing. The largest assignment in this class is a four thousand word research paper that students must write individually. Kay explains that teachers of this class focus on the concepts that the students write
about rather than grammar or spelling errors.[21] He concludes, that it is important for students to have computer science faculty, show students the importance of writing by teaching classes like this.

3.1.1.2 University of Texas at Austin

Lisa Kaczmarczyk, addressed similar problems of students’ poor writing ability, at the University of Texas at Austin in a 2003 paper.[19] She decided to create a separate class in order to focus on student’s writing abilities. In this class, students learn about both academic and industrial technical writing.[19] In both portions of the class, students are expected to read and write in the specified style of writing. At the end of the study, Kaczmarczyk found that the students in this class reported an increase in confidence in their ability to express their opinions in writing, and an increase in their overall confidence in their writing ability.

3.1.2 Multiple Technical Courses with Writing Assignments

The second approach is to add writing to multiple technical courses. This allows universities to teach the same technical skills they would normally, while improving their student’s writing abilities. The rest of this section will look at some examples of universities that have done this

3.1.2.1 University of Delaware

Lori Pollock wrote about how the University of Delaware incorporated writing into its curriculum in 2002. At her university, students are mostly taught in seminar classes where both students and teachers take turns leading the discussion.[36] Pollock found that most of the students who were not participating in the class discussions were ill-prepared for the class. This led to long, boring class periods with limited
discussion.[36] To combat this problem, the University of Delaware has integrated writing assignments into many of their seminar courses without taking away from their technical aspects. Pollock explains how she believes that by integrating more writing assignments, the students would take the additional opportunities to fix problems with their writing.[36]

3.1.2.2 Carnegie Mellon University Qatar Campus

In 2011, Karatolis, et al., describe a similar approach of adding writing aspects to technical courses that was implemented at the Carnegie Mellon University, Qatar campus. They explain that this approach gives students more technical practice.[20] This practice allows students to simulate what types of writing and communication scenarios they could encounter in the field. The authors observed that students are less reluctant to write after taking the courses.[20]

3.1.3 Strategies

The last approach is to use different strategies to teach writing. Instead of teaching writing for a general audience like most English classes do, these strategies focus on the needs and knowledge of computer science and software engineering students.

3.1.3.1 Muhlenberg College

In 2005, Clifton Kussmaul published a paper describing a strategy being used in Muhlenberg College. He noticed that the processes used by successful writers are very similar to the agile software development method.[23]

Agile development is an iterative process where programmers and clients have frequent meetings about the software and changes that need to be made to it. This is in contrast to waterfall software development, where clients and programmers would
only meet at the beginning of the development process. Agile development allows client’s feedback to change the product while still in development. Between each of the meetings with the clients, the programming team adds to the product by adding new features or by fixing bugs.

Kussmaul uses ideas from agile development, like risk management and peer review, and explains how these ideas could be used for writing.[23] Although it explains possible ways this could be used in a classroom setting, this paper did not explain the results of using this strategy in the classroom.

3.1.3.2 University of the Balearic Islands

Joe Miró Juliá explains different strategy in his paper published in 2011. He believes that the barrier keeping programmers from writing well is the terminology used in writing.[30] Programmers are used to being able to see if their work is correct or not by testing if their programs have all the required functionality. Writing, on the other hand, is much more subjective.[30] This makes it harder for students to tell if they are doing well or not on their own.

Juliá tries to solve this by creating a set of “crafts.”[30] These crafts consist of a principle and a rationale. The principle provides guidance on how to write and the rationale explains why the principle makes writing better.[30] These crafts allow writers to test if they have followed the principle to see if their paper is good in this regard. Juliá also explains that following the crafts alone is not enough to make a good paper; papers also need their writer’s own writing style. Overall, Juliá says that the crafts do help students write better papers.[30]
3.1.4 Comparison to this Approach

The rest of this section goes over what other universities’ approaches are to help improve students’ writing performance. This subsection will explain how the pedagogy compares to the other approaches and what gaps the pedagogy is trying to fill. Table 3.1 gives a visual summary of how all four approaches compare to one another. In the table a check mark means that that approach has that benefit.

The universities who teach a single class centered around writing papers and the universities who include writing papers in multiple technical courses do not focus on teaching students how to write. While writing is required in these approaches, they assume students already possess the abilities necessary to write papers for the programming field effectively. This pedagogy and the strategy approaches described, fix this by trying to improve students’ writing abilities by teaching them new ways to think about their writing.

All three of the described approaches lack empirical data or metrics to gather this data on. None of them look at actual numbers or quantifiable student accounts. One of the primary goals of this study is to be able to analyze if the strategy the pedagogy is using is effective. The pedagogy does this by gathering both subjective and objective data from the students and their work. The metrics that were used to analyze the effectiveness of the pedagogy are discussed in the results chapter. These metrics are based on the lessons that are fully explained within the experimental design chapter.

3.2 Knowledge Transfer’s Applications

The pedagogy uses transfer to help students understand writing. To fully understand how transfer can be used, this section will go over a few cases where transfer has been
Class centered
around writing

Continual
Reinforcement

Teaches
Programmers
How To Write

Empirical Data
on Set Metrics

<table>
<thead>
<tr>
<th></th>
<th>Class centered around writing</th>
<th>Continual Reinforcement</th>
<th>Teaches Programmers How To Write</th>
<th>Empirical Data on Set Metrics</th>
</tr>
</thead>
<tbody>
<tr>
<td>Single Class</td>
<td>✓</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Multiple Classes</td>
<td></td>
<td>✓</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Strategies</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td></td>
</tr>
<tr>
<td>The pedagogy</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
</tbody>
</table>

Table 3.1: Comparison of existing approaches and the pedagogy

3.2.1 Facilitating Transfer for Writing

Transfer is mainly studied in the English field. As such, most research in transfer techniques is based on how to encourage students to transfer skills from their first-year composition (FYC) courses to their future writing. Techniques discovered in these studies serve as a solid foundation on how to teach for transfer.

3.2.1.1 James

In 2010, Mark Andrew James did a case study about students transferring skills from their FYC course to other courses.[18] The study followed eleven students enrolled in an FYC course by interviewing them and observing their writing from other courses they were taking. In the interviews students were asked about whether they used what they learned in their FYC course in their writing for other classes.[18] After analyzing students papers and interviews, they conclude that although transfer happens in all of the cases, it is more frequent in some disciplines and tasks than others.[18]
3.2.1.2 Alexander, et al.

Alexander, et al., explained how multi-literacy centers could help facilitate transfer between different types of media in 2016. A multi-literacy center is a writing center that also helps with other disciplines. They theorize about how tutors in multi-literacy centers need to help students transfer knowledge across different mediums.[7] An example of this, is knowing what skill to use when taking a book and making a movie from it, or vice versa. To look into this idea, they followed a student named Sophie who had to write a paper and make a media presentation about the same topic.[7] The researchers drew conclusions about what made Sophie successful in this task, like sectioning and music choice. They also describe 5 challenges Sophie’s final media presentation included that a multi-literacy center consultant could have helped with. [7]

3.2.1.3 Fallon, et al.

Diane Fallon, et al., did a study in 2009 that looked at transfer between FYC and writing in a psychology course.[15] During the study, they observed how much students had retained from their FYC course while they wrote papers for the psychology course. The study concluded that there was no one course that could assure the students would become effective writers.[15]

Their research did give some very important, unexpected results. From the surveys and papers they received, the researchers noticed that students are generally aware of what their weaknesses are.[15] This means that when a student wrote that they were not good at spotting and correcting run-on sentences, it generally meant that the paper the student wrote also had many run-on sentences.[15]
3.2.2 Transfer Summary

All of the examples given, focus on students transferring writing knowledge to new writing situations. This study will be attempting to help students transfer programming knowledge to writing research papers instead of transferring writing knowledge to another writing context. In addition to trying to improve students’ writing ability, the study will also look at students’ confidence. This is based on both Fallon’s results and a study done in 1994 that found confidence to be a good indicator of academic performance.[32, 15]

3.3 Related Works Summaries and Conclusions

Section 3.1 looked at how other universities are attempting to help students write better. These schools were grouped into three different types of approaches: a single class based around writing papers, multiple technical courses with writing activities, and strategies that try to help students learn to write better. None of these approaches gathered data based on specified metrics and only the strategy approaches tried to teach students how to write. Unlike these approaches, the primary focus of this study is gathering data on the effectiveness of the pedagogy.

Section 3.2 looked at how transfer has been used before. Many studies have found transfer useful in improving students’ writing. Fallon’s results showed that confidence should be taken into account when looking into students’ academic performance as well. Unlike the other studies, this study is transferring programming knowledge to writing. From this, the pedagogy looks at whether transferring programming knowledge can help improve students’ confidence or writing ability.
To help programming students with their writing, this study proposes a new pedagogy. This pedagogy focuses on facilitating the transfer of programming knowledge to writing. The study was done on two sections of CSC 300, in Spring 2017. The demographics of the students who originally signed up for the study can be seen in Appendix E. To facilitate this transfer, in these students, the pedagogy consists of a set of 6 lessons, which are a combination of a lecture and a lab.

The lecture and lab format allows students time in class to practice implementing that week’s lesson. Students have time in class as well as time over the weekend to complete the labs. This allows students to have time to clarify any misunderstandings they had from the lecture.

The lecture for each lesson focus on bridging the programming practice to the writing practice. The lessons also took time to explain where the students could apply these practices, in their paper for the class.

The lab for each lesson focused on having students identify places in their own paper where they could apply that week’s lesson. After applying the lesson, they would turn in excepts from their paper, with an explanation on why they changed that excerpt. This should help show that the students are successfully creating bridges for high-road transfer.

4.1 Goals of the Pedagogy

The pedagogy was intended to improve students in two main ways: their confidence about their ability to write and their actual writing ability, measured against the
4.2 Evaluating the Pedagogy

There are two sets of tests that were used to evaluate the pedagogy’s effectiveness. One set of tests looked at the improvement of student confidence. The other looked at whether student writing improved according to the study’s specifications, after being taught the lessons.

Student perception of each lesson will also be examined in order to potentially cross check for threats to validity. A threat to validity would be found if students found a lesson to be ineffective, when a result is dependent on that lesson. The rest of this section will describe these tests as well as what the ideal results of these tests would look like.

4.2.1 Test for Confidence

4.2.1.1 Evaluating Confidence

Answers from surveys, filled out by the students, will be examined to see if student confidence has improved. Their responses are gathered and checked to see if it has a multivariate normal distribution. If so, a Hotelling’s t-squared (T2) test will be done for the results.[17] Hotelling’s T2 test was chosen over a normal t-test because it helps eliminate type-1 errors. This will be explained more fully in the results chapter.

4.2.1.2 Confidence Data

Students received two surveys that they completed in class. These surveys can be seen in Appendix B. The introduction survey was filled out during the first meeting with the students. The exit survey was filled out after all the lessons were taught but
before students turned in their final papers for grading. The timing of the exit survey was calculated to reduce bias based on final grades.

4.2.1.3 Ideal Results

The pedagogy succeeded if there is a general rise in student confidence compared to the beginning of the study. If the pedagogy changes the way the students view writing, it is positively affecting the way they will continue writing.

4.2.2 Test For Writing Improvement

4.2.2.1 Evaluating Writing Improvement

The number of errors from student papers are gathered to see if their writing has improved. The number of errors in the paper are divided by the number of pages that have text on them to create an error rate. The number of pages that have text on them does not include pages that are only bibliography or the table of contents. However, it does include the title page as it contains the abstract. Tests based off of the error rates were used to measure student writing improvement.

4.2.2.2 Writing Improvement Data

Throughout CSC 300, students are expected to turn in several drafts of their paper. These drafts are their proposal, a revised proposal, a half-length draft, a revised half-length draft, and their final paper. For each student, a sheet was filled in, which can be found in Appendix C, according to the guidelines that can be found in Appendix D.
4.2.2.3 Ideal Results

The pedagogy does not intend to fully eliminate the errors in the student’s writing. Instead, it attempts to create an overall decrease in the rate in which students make these errors. If the pedagogy can decrease this rate it can be concluded that the lesson relating to that skill has helped improve student writing.

4.3 Lessons

The rest of this chapter describes in detail the six lessons made for this pedagogy. Each of these sections will describe the lesson’s concept, why this lesson was included, the concepts being transferred in the lesson, and the lab that students completed for that lesson. It concludes with a timetable of when lessons were taught in comparison to when students’ drafts were due.

4.3.1 Prompt Testing

The purpose of the prompt testing lesson is to have students check how well their papers meet the specifications. It attempts to help students see the importance of checking their papers against the specifications, early and often. Students already know how to check their programs against a specification on a regular basis and understand the consequences of failing to do so. In practice, Microsoft explains that “Testing is a vital part of software development, and it is important to start it as early as possible, and to make testing a part of the process of deciding requirements.”[6]

The prompt testing lesson was chosen because students in previous quarters have failed the paper by not fulfilling all of the specifications. Although, this might be second nature for many students, it is important to make sure that students know the value of following the specifications set for their writing.
4.3.1.1 Concepts Being Transferred

Acceptance tests are tests set out by clients to determine if a program has the required features.[29] “The acceptance tests give the customer confidence that the application has the required features and that they behave correctly.”[29]

In a class, the customer is the professor who wrote the specification for the paper. The paper specifications give the features of the paper that are required in order to pass the class. Prompt testing is the act of checking whether the features outlined in the specification have been fulfilled.

Both prompt testing and acceptance testing require looking at specifications and going back through the product, whether it be the program or the paper, to ensure that the customer will be satisfied that the specifications were met. Students already know how to do this when they are programming. By showing students a parallel in the programming field, the goal is that they will be more diligent about checking if their papers fulfill the specifications.

4.3.1.2 Prompt Testing Lab

The prompt testing lesson was taught right after students turned in their proposals. The lab of this lesson was to identify and correct parts of their proposal that didn’t meet the specifications. Ideally, by giving the students time to correct any of the errors they would get more constructive instructor feedback.

4.3.2 Single Term Per Concept

In the single term per concept lesson, the importance of consistent word choice in technical writing was stressed. It has students transfer knowledge about def-use chains to their word choice in writing.
Historically, students have had a habit of using different words to explain a single concept. While this is acceptable for other styles of writing, it can take away from the explanation in technical writing. Previous students have stated that changing their writing style to keep consistent word choice for topics was one of the hardest parts of the paper. The single term per concept lesson is meant to help students with this hurdle so they could focus on the logic of their paper.

**4.3.2.1 Concepts Being Transferred**

Def-use chains are a programming concept to help programmers keep track of their variables from definition to use without any redefinitions.[25] Redefining variables changes what their underlying data is. This makes the original definition of the variable useless because it can no longer be accessed. This can lead to confusion between the documented use of the variable versus what the variable will be used for after the redefinition. Students are taught not to do this because it will likely cause the program to no longer compile or give incorrect results.

Using a single, consistent term for a concept in technical writing helps keep readers focused on what the writer is trying to explain. Technical writing should focus on the clarity of the word choice rather than the elegance of the words. [39] This clear, concise language helps readers see the reality of the situation rather than obstructing the author’s original meaning. [28] Because of this, the students of CSC 300 are also required to keep consistent terms for their concepts. Using a single term for a concept prevents ambiguity for readers. It makes sure that the reader knows exactly what is being explained each and every time.

Def-use chains force programmers to keep track of where they define and use variables. Readers have a similar problem to the compiler if they do not understand the underlying semantics correctly. By transferring this skill, students can make their
writing less ambiguous to their readers.

4.3.2.2 Single Term Per Concept Lab

In the students’ papers they have to write a subsection for each of their software engineering code of ethics tenets called the Domain Specific Rule (DSR). A DSR is a section where students specialize the broad terms of the code tenet to match the terms and concepts in the domain of the real-world case. The purpose of this is to enable a more direct application of the code tenet to the specific real-world case.

For this lab, students were asked to write at least one of their DSR sections. Anecdotally, the students would make the error of using multiple terms for the same concept the most, in this section.

4.3.3 Peer Review

The peer review lesson tries to help students improve how they do peer reviews. Peer review can be an invaluable tool in a writer’s arsenal. Despite this, many of the students’ peer reviews focused on small errors like spelling and grammar instead of looking at whether the paper conceptually made sense.

The peer review lesson was chosen to encourage students to get help from their peers. The goal was that this would give the students a format for them to review each other’s papers that could be beneficial for both parties.

4.3.3.1 Concepts Being Transferred

Code review is where a team of programmers all meet up to go over an important piece of code. They go over the code, line by line, out loud, so any errors or misconception can be pointed out. This helps the whole group be on the same page when it comes
to the piece of code.[14] With all errors and misconceptions noted for review, the members of the team can work to make sure that everything that uses this code runs correctly. Code review can be done in many different ways. The type of code review referenced here was first set out by M.E. Fagan in 1976.[14] This is a highly structured version of code review compared to some of the newer approaches.[8]

Similar to code review, peer review is where an individual gets someone else to read their paper and leave comments on it. [26] Preferably, the other person is also working on a similar paper or knows the requirements of the work so they can help, if the author deviates from the original purpose. Peer review is important for students because it allows them to get feedback on the clarity of their writing. It can also help them get a better understanding of the specifications set for their writing. This means that both the reviewer and the person getting reviewed can get valuable information from the peer review.

In the lecture, the structure that the code review uses was stressed. It mainly focused on the idea of sitting down and verbally going through the information. Getting students to vocalize the problems they see in their peer’s papers allows the author to see a problem and start a conversation about how to fix it. By opening up this conversation, students, can not only identify problems with each other’s papers but also discuss ways to fix those problems.

4.3.3.2 Peer Review Lab

For the peer review lab, students paired off and did peer reviews together. They had to turn in pictures of the paper that was reviewed as well as a new draft of their paper. This was done so it was clear whether the students actually made changes to their papers based on their review.
4.3.4 Paragraphs

In the paragraph lesson, students were encouraged to write focused paragraphs with only one main concept explained in each. Students can make their writing easier to read by making sure that each of their paragraphs is focused on one concept. This makes their reasoning easier to follow and their papers, in general, get a better grade.

The paragraph lesson was chosen because many students have a problem with packing too much information into a single paragraph. This makes their writing harder to read, and a chore to comprehend. By students separating their paragraphs into smaller, more concise paragraphs, their writing can become easier to read and comprehend.

4.3.4.1 Concepts Being Transferred

The Single Responsibility Principle (SRP), is the idea that every function or method should only be responsible for one bit of functionality.[27] Following this principle makes code easier to read, debug, and reuse. It makes the code into smaller, functional bits, rather than one large piece of code that does all of the functionality.

A paragraph should be a unit of text explaining one concept.[37] In writing the functionality of a unit of text is the meaning it conveys to the reader. This means that paragraphs that only explain one concept would be responsible for one bit of functionality, just like functions or methods. The benefit of separating paragraphs like this is readers can understand each concept being explained without another concept confusing them. It also allows readers to go back and reread smaller segments to understand a concept if they need to.

Since students understand how to break up their code, they can apply the same thinking to their paragraphs. By changing the way they think about their paragraphs,
this transition can be easy and something they can use throughout their writing.

4.3.4.2 Paragraphs Lab

For the paragraph lab, students went through their papers and identified places they should separate a paragraph into one or more additional paragraphs. Students submitted what the paragraph once looked like, their revisions, and why they chose to separate the paragraph in the way that they did.

4.3.5 Sections

The section lesson, aimed to extend student’s understanding of the paragraph lesson. This lesson focuses on helping students make decisions on how to create their sections. By having well-separated sections a writer makes it easier for readers to navigate the paper. This makes it easier for reader to comprehend and makes the points the author is trying to convey easier to understand.

The section lesson was chosen because it is important that students know how to break their papers up into sections. When a paper is well separated into different sections the table of contents becomes a roadmap for readers to understand what they are going to be reading about and where they are going to be reading about that information. By giving readers this roadmap the paper itself become easier to navigate and understand.

4.3.5.1 Concepts Being Transferred

The Single Responsibility Principle (SRP) was previously explained as it relates to functions. The SRP is also used for classes in object-oriented programming. Classes in programming should contain all the information necessary for that class to have its full functionality.[27] It should not contain information or methods for anything
outside of that class’ functionality. If it does, then the program it is a part of will be more confusing to use and maintain.

Similarly, sections in a paper are paragraphs grouped together to fully explain one concept.[33] As explained in the previous lesson, the functionality of a unit of text is the information that the reader is meant to take from that text. So just like classes must contain all of the methods for its functionality, a section should contain all of the paragraphs necessary to explain a concept to the reader. Sections also help readers understand what to expect in the paragraphs to come. This makes sections one of the most effective ways that the students can make their writing more readable.[33]

The section lesson focuses on changing student’s views on their writing. By showing them how similar these two concepts are, students can adapt how they look at sections in their papers. This lesson can show them the importance of creating well thought out sections in their papers instead of sections being something they are required to have.

4.3.5.2 Sections Lab

For the sections lab, students looked at all of their Domain Specific Rules (DSRs) and created section titles for the analysis of the DSR. This is the main bulk of the student’s papers so having feedback on how they separated this section would help with how they separate the rest of their paper.

4.3.6 The Sixth Lesson

The sixth lesson was meant to help students transfer ideas from using external libraries to how they did citations. However, due to time constraints, this lesson was not taught to the students. This left the previous five lessons that were taught to the students.
The citation lesson, in particular, was cut over the other labs because many of the students did not have a problem with how they were citing their sources. Furthermore, while proper use of citations is a vital aspect of any paper, out of all the lessons planned this lesson would do the least to instill confidence in students for their overall writing ability.

**4.3.7 Order the Lessons were Taught**

This section will explain why lessons were taught in relation to when student drafts were due. Table 4.1 shows when all of the lessons were taught and when drafts were due. Remember that the citation lesson was not actually taught. It is included in this chart for referential purposes only.

<table>
<thead>
<tr>
<th>Week</th>
<th>Lesson</th>
<th>Writing Concept</th>
<th>Programming Concept</th>
<th>Paper draft due</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Introduction to Transfer</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>2</td>
<td>Prompt Testing</td>
<td>Prompt Testing</td>
<td>Acceptance Testing</td>
<td>Proposal</td>
</tr>
<tr>
<td>3</td>
<td>Single Term Per Concept</td>
<td>Single Term Per Concept</td>
<td>Def-Use Chains</td>
<td>N/A</td>
</tr>
<tr>
<td>4</td>
<td>Peer Review</td>
<td>Peer Review</td>
<td>Code Review</td>
<td>Half-length draft</td>
</tr>
<tr>
<td>5</td>
<td>Paragraphs</td>
<td>Paragraphs</td>
<td>SRP</td>
<td>N/A</td>
</tr>
<tr>
<td>6</td>
<td>Sections</td>
<td>Sections</td>
<td>SRP</td>
<td>N/A</td>
</tr>
<tr>
<td>7</td>
<td>Citations</td>
<td>Citations</td>
<td>Using External Libraries</td>
<td>N/A</td>
</tr>
<tr>
<td>8</td>
<td>Instructor Feedback</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>9</td>
<td>Transfer Wrap up</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
</tr>
<tr>
<td>10</td>
<td>N/A</td>
<td>N/A</td>
<td>N/A</td>
<td>Final Draft</td>
</tr>
</tbody>
</table>

**Table 4.1: Table of when lessons were taught and the concepts transferred in those lessons**

The introduction to transfer was where the students were introduced to the concepts behind the pedagogy. This was when students could choose to sign up for the
experiment or not. If they did volunteer, they filled out the introduction survey. Similarly, the transfer wrap up was when students filled out the final surveys and chose whether or not to do an exit interview.

The prompt testing lesson was taught as students were meant to turn in their proposals. This lesson was taught on this date so students could amend their proposals to meet the current specifications.

The single term per concept lesson was taught between the student’s proposals and half drafts. The DSR sections were one of the new requirements of the student’s half-length drafts. This section had traditionally confused many students and made it difficult to move forward. This lesson was taught between the two drafts to help prevent this problem.

The peer review lesson was taught on the day the student half-length drafts were due. Having students go through their papers with each other at this point could be advantageous in several ways. The first is that revising their draft at this point could lead to more productive instructor feedback. By introducing this technique at this point, students would be encouraged to do more peer reviews with each other before the final draft.

The paragraph and section lessons were taught between the student’s half-length and final drafts. These two lessons were taught toward the end of the quarter so students had more writing to look through. Having students work on these final touches toward the end of their papers would give them more practice so they would continue using these techniques in the future.
Chapter 5

USER STUDY

This study was done at Cal Poly during the Spring 2017 quarter. Lessons were taught to students during their Friday class for CSC 300. These lessons were taught by the researcher, and the Monday and Wednesday classes were taught by Professor Clark S. Turner. Students were only taught about how to improve their writing during these Friday classes.

Students in the CSC 300 class were asked to volunteer to join the study, during the introduction to transfer lesson on week one. They were informed that participation was optional and would not affect their grade in the class or the information that they would be taught. Students who did sign up for the study were given a four digit code number to anonymize their work and responses. While all students in the class were given the same lessons, only students who signed up for the study had data gathered on their work. Students who signed up for the study then did a survey in class asking them about themselves and their confidence in writing.

Lessons took place on Fridays from 2 PM to 4 PM for section one and 4 PM to 6 PM for section three. Each 2-hour class consisted of a 30 to 60-minute lecture and the rest of the class was used as a lab period. Each lab was due on Monday. So while students could spend the lab time in class and getting assistance from the researcher, they were not required to do so. Lab assignments were checked for completion the Monday they were assigned after the class had began.

On week nine, students who signed up for the study were asked to do an exit survey. This survey was done in class and given back before the students left. They could also volunteer to do an exit interview with the researcher. The interview asked
students which lessons they felt helped them the most and which lessons did not.
6.1 Setbacks

There were several setbacks that had a major effect on the results. This section will go over what these setbacks were and how each of them affected the results of the study. We will go over how to avoid or solve these setbacks in the conclusions and future work chapter.

6.1.1 Lack of Instructor Feedback

Throughout the course, students, unfortunately, did not get a significant amount of instructor feedback. Without instructor feedback, students had a difficult time seeing if the lessons were improving their papers.

This contributed to student apathy towards the transfer lessons. Many of the students mentioned that they felt like more instructor feedback would have helped them understand whether they were properly applying the lessons.

One area the lack of instructor feedback could have affected is the student’s confidence. Student confidence is likely to decrease because at the point of the exit survey they still did not know how they were doing. This combined with the pressures of finishing the paper are likely to have affected how students felt about their skills.

6.1.2 Student Participation

In this study students were not required to come to the transfer lessons. The labs from the lessons were worth five percent of their total grade, but this did not convince
all students to participate.

Many students did not turn in the labs even though they were worth five percent of their grades. Students’ labs were graded pass/fail based on whether the lab was turned in. This was done to avoid biasing student labs and invalidating the results. Because of the pass/fail grading style students could turn in anything for the labs and get full credit regardless of its content.

Another major factor that influenced student participation was when the class was scheduled. Transfer lessons for the classes were held on Fridays starting at 2 PM or 4 PM based on which section students were in. This meant that this class was likely the last class students had before the weekend. This caused many students to be less attentive or skip the class entirely.

Since students did not actually have to complete the labs for credit many of them either did not come to lecture or did not turn in the lab. Unfortunately, it cannot be determined how many of the lessons’ lectures each individual student was present for. For example, a student who did not come to lecture could have turned in the lab or a student who did come to lecture could have forgotten to turn in the lab. As a result, the pedagogy might not be the direct cause of any of the changes found in the study. However, the pedagogy can be associated with any of the changes found in the study.

6.1.3 Lab Timing

The last major setback was determining when to schedule the lessons. There were two possible approaches to scheduling the lessons, focusing on the study or focusing on what would help the students the most.

The reason these are mutually exclusive is based on when students would need the knowledge. For example, the one term per concept lesson could have been taught
before or after students’ half-length drafts were due. If done before the half-length
draft, students would have an easier time writing their draft but less usable data
could be gathered. If done after the half-length draft, there would be a draft of their
papers where the concept is necessary but has not been taught yet. In this case, data
can be gathered from their papers for this lesson but students would have a more
difficult time writing their draft.

This study chose to put the students first. Therefore, if there was a choice to
teach a lesson before it was needed, the lesson was taught earlier for the benefit of the
students. Unfortunately, this means that for some of the lessons there is no accurate
data. This is because there was not a draft of their papers where the concept of the
lesson was necessary before the lesson was taught.

6.2 Results: Confidence

The confidence results data was gathered from the answers students gave from the
introduction and exit surveys. Since the citation lesson was not taught, data gathered
from questions four and five from the exit survey will not be analyzed. Instead,
questions one through three and questions six and seven from the exit survey will be
analyzed.

To analyze the data gathered from the survey, for all questions but seven, a
paired Hotelling’s T2 was used test to check for a statistically significant change in
confidence. This test is useful for multivariate data because it accounts for type 1
error. A type 1 error is when the null hypothesis is incorrectly rejected, also known
as a false positive. This study is prone to this type of error because multiple data
points are being gathered from a single sample.

For the paired Hotelling’s T2 test, a p-value less than 0.05 is necessary for sta-
tistical significance, this indicates at least 95% confidence in the results. The null
The hypothesis used is that all of the questions have a mean difference equal to zero. From the paired Hotelling’s T2 test a p-value of 0.04571 was obtained, therefore at least one of the questions has a significant change. This means individual t-tests need to be done to see which questions have a significant change and what this means to the study.

For all of the questions, the number students gave in the introduction survey were subtracted from the number they gave in the exit survey. This means that if there is a positive mean that the students’ confidence improved where a negative mean shows that students’ confidence worsened. For each of the tests a p-value less than 0.05 is necessary to reject the null hypothesis. The null hypothesis in this case is the mean is equal to zero.

Question one from the exit survey, asked students to rate how good of a writer they believed themselves to be. Figure 6.1 shows how students rated their confidence in their writing at the beginning and end of the study. Given this figure it seems students confidences did not increase by a lot if they did increase. There are some outliers on both sides however, the trend seems to be a slight increase overall. Doing a t-test on this data returned a p-value of 0.05227. The null hypothesis cannot be conclusively rejected because the p-value is over 0.05. However, this p-value is very close to being a statistically significant result so it cannot decisively be concluded that the pedagogy did not improve students’ views. Therefore, this is the area of our study that would benefit from additional research.

Question two from the exit survey, asked students about how important they think writing is to the programming field. Figure 6.2 shows students’ ratings on how important they viewed writing to be in their field at the beginning and end of the study. This figure has points all over the place, both next to the line and farther away from it. This would seem to indicate that there isn’t a connection between the
Figure 6.1: Student confidence in writing with the exit survey on the x-axis and the introduction survey on the y-axis. The gray line shows no change in confidence. Every point below the line shows an increase in confidence after the lessons and every point above the line shows a decrease in confidence after the lessons. The opacity indicates multiple points plotted on the same location. n = 35
pedagogy and how students view the importance of writing. The t-test on this data returned a p-value of 0.6334. This means the null hypothesis cannot be conclusively rejected because the p-value is over 0.05. A p-value this high shows conclusively that the pedagogy has no effect on how important students think writing is. Therefore, future research on this pedagogy should not focus on this aspect of student’s views towards writing.

Figure 6.2: Student belief that writing is important in the field with the exit survey on the x-axis and the introduction survey on the y-axis. The gray line shows no change in importance. Every point below the line shows an increase in importance after the lessons and every point above the line shows a decrease in importance after the lessons. The opacity indicates multiple points plotted on the same location. n = 35

Question three from the exit survey, asked students to rank their confidence in
their writing ability. Graph 6.3 shows students’ confidence in their writing ability before and after the lessons. From this graph it seems most students’ confidence increased after the lessons because there are more opaque points below the line. The t-test on this data returned a p-value of 0.08313 which means null hypothesis cannot be rejected because the p-value is over 0.05. However, this p-value is also reasonably close to being a significant result. This result and how similar this question is to question one warrants future research in the context of this pedagogy.

Figure 6.3: Student confidence in their writing ability with the exit survey on the x-axis and the introduction survey on the y-axis. The gray line shows no change in confidence. Every point below the line shows an increase in confidence after the lessons and every point above the line shows a decrease in confidence after the lessons. The opacity indicates multiple points plotted on the same location. n = 35
Question six from the exit survey, asked students to rate their ability to write logical, well-organized papers. Graph 6.4 shows students’ confidence in their ability to write logical, well-organized papers before and after the lessons. Most of the points are below the lines as indicated by the more opaque points below the lines. There is also one drastic change, point (7,1) showing an increase of 6 points of confidence. This would mean that students confidence generally increased with some student’s confidence increasing a lot by the end of the study. The t-test on this data returned a result of 0.00977. This means the null hypothesis can be conclusively rejected because the p-value is less than 0.05. Then the data was ran with the alternate hypothesis that the mean is greater than zero. This returned a p-value of 0.004889 which means the alternative can be accepted. Since the mean is greater than zero, it can be concluded that the pedagogy is associated with an increase in student’s confidence in their ability to write logical, well-organized papers. This means future work can look at this factor and possibly prove that the pedagogy is the cause of this increase.

Question seven from the exit survey, asked students if the lessons had helped their confidence in any of the previous areas. Out of the thirty-eight exit survey responses, thirty-four of them responded that the lessons had helped. This means that 89.47% of the students felt like the lessons were beneficial to their confidence in some way.

6.3 Results: Writing Ability

To check if student’s writing ability improved, data was gathered from the student’s drafts. A form was filled out for each student that can be found in Appendix C. The form was filled out based off of guidelines that can be found in Appendix D. After some analysis of the raw data for the “Times a defined term was used” and “Times a section had multiple concepts” it was determined that these rows did not contain valuable information. This is likely because of reasons explained in the lab timing.
Figure 6.4: Student confidence in their ability to write logical, well-organized papers with the exit survey on the x-axis and the introduction survey on the y-axis. The gray line shows no change in confidence. Every point below the line shows an increase in confidence after the lessons and every point above the line shows a decrease in confidence after the lessons. The opacity indicates multiple points plotted on the same location. $n = 35$
subsection of the setback section. This leaves the “Times a paragraph explained multiple concepts” row for analysis.

To analyze if students had gotten better at separating their paragraphs, data was gathered from student’s most recent half-length drafts and final papers. These two drafts were chosen because the half-length draft was turned in before the paragraph lesson was taught and the only draft after it is the final paper.

To account for the increase in the length of student papers, the error rate was used rather than just the number of errors. The error rate of a paper is the number of errors divided by the number of pages. These pages did not include the table of contents or the pages for the bibliography. However, it did include students’ cover pages because these contained the abstract. Table 6.1 and Table 6.2 show the results for section one and section three respectively and Table 6.3 shows the averages for both classes. The change in error rate is calculated by subtracting the half-length draft error rate from the final draft error rate. Therefore, a negative mean overall shows students improved while a positive mean overall shows students worsened after the lesson.

Figure 6.5 shows a visualization of the error rate of the final paper on the x axis, and the error rate of the half-length draft on the y axis. The first section of the class is visualized in blue and the third section of the class is visualized in orange. For this figure every point above the line shows a decrease in error rate and every point below the line shows an increase in error rate. From this figure it seems like there is an overall decrease in student error rate, because a majority of the points are above the gray line.

To test whether students’ error rate actually decreased, a two-sided t-test was done. Again, a p-value less than 0.05 is necessary to reject the null hypothesis, and the null hypothesis is that the mean is equal to zero. This resulted in a p-value of
<table>
<thead>
<tr>
<th>ID Number</th>
<th>Half-length Draft Errors</th>
<th>Half-length Draft Length</th>
<th>Half-length Error Rate</th>
<th>Final Paper Errors</th>
<th>Final Paper Length</th>
<th>Final Paper Error Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>2028</td>
<td>16</td>
<td>5</td>
<td>3.2</td>
<td>17</td>
<td>11</td>
<td>1.55</td>
</tr>
<tr>
<td>2570</td>
<td>38</td>
<td>6</td>
<td>6.33</td>
<td>59</td>
<td>12</td>
<td>4.92</td>
</tr>
<tr>
<td>2826</td>
<td>13</td>
<td>5</td>
<td>2.6</td>
<td>36</td>
<td>10</td>
<td>3.6</td>
</tr>
<tr>
<td>3076</td>
<td>23</td>
<td>7</td>
<td>3.29</td>
<td>26</td>
<td>10</td>
<td>2.6</td>
</tr>
<tr>
<td>4349</td>
<td>13</td>
<td>5</td>
<td>2.6</td>
<td>29</td>
<td>12</td>
<td>2.42</td>
</tr>
<tr>
<td>4584</td>
<td>16</td>
<td>5</td>
<td>3.2</td>
<td>13</td>
<td>13</td>
<td>1.0</td>
</tr>
<tr>
<td>5507</td>
<td>11</td>
<td>5</td>
<td>2.2</td>
<td>14</td>
<td>11</td>
<td>1.27</td>
</tr>
<tr>
<td>5616</td>
<td>12</td>
<td>4</td>
<td>3.0</td>
<td>25</td>
<td>11</td>
<td>2.27</td>
</tr>
<tr>
<td>5627</td>
<td>17</td>
<td>5</td>
<td>2.4</td>
<td>17</td>
<td>13</td>
<td>1.31</td>
</tr>
<tr>
<td>5874</td>
<td>5</td>
<td>9</td>
<td>0.56</td>
<td>0</td>
<td>13</td>
<td>0</td>
</tr>
<tr>
<td>5989</td>
<td>23</td>
<td>9</td>
<td>2.56</td>
<td>17</td>
<td>11</td>
<td>1.55</td>
</tr>
<tr>
<td>7116</td>
<td>5</td>
<td>4</td>
<td>1.25</td>
<td>10</td>
<td>14</td>
<td>0.71</td>
</tr>
<tr>
<td>7434</td>
<td>7</td>
<td>6</td>
<td>1.17</td>
<td>9</td>
<td>13</td>
<td>0.69</td>
</tr>
<tr>
<td>7545</td>
<td>18</td>
<td>6</td>
<td>3.0</td>
<td>10</td>
<td>12</td>
<td>0.83</td>
</tr>
<tr>
<td>7822</td>
<td>5</td>
<td>4</td>
<td>1.25</td>
<td>26</td>
<td>13</td>
<td>2.0</td>
</tr>
<tr>
<td>7851</td>
<td>3</td>
<td>7</td>
<td>0.43</td>
<td>2</td>
<td>12</td>
<td>0.17</td>
</tr>
<tr>
<td>9352</td>
<td>3</td>
<td>8</td>
<td>0.38</td>
<td>2</td>
<td>13</td>
<td>0.15</td>
</tr>
<tr>
<td><strong>Average</strong></td>
<td><strong>11.76</strong></td>
<td><strong>5.62</strong></td>
<td><strong>2.13</strong></td>
<td><strong>17.94</strong></td>
<td><strong>12.06</strong></td>
<td><strong>1.55</strong></td>
</tr>
</tbody>
</table>

Table 6.1: Section one’s paragraph error rates for the half-length draft and final paper. \( n = 17 \)
<table>
<thead>
<tr>
<th>ID Number</th>
<th>Half-length Draft Errors</th>
<th>Half-length Draft Length</th>
<th>Half-length Error Rate</th>
<th>Final Paper Errors</th>
<th>Final Paper Length</th>
<th>Final Paper Error Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1027</td>
<td>16</td>
<td>6</td>
<td>2.67</td>
<td>15</td>
<td>13</td>
<td>1.15</td>
</tr>
<tr>
<td>1084</td>
<td>5</td>
<td>6</td>
<td>0.83</td>
<td>9</td>
<td>11</td>
<td>0.82</td>
</tr>
<tr>
<td>1277</td>
<td>6</td>
<td>7</td>
<td>0.86</td>
<td>18</td>
<td>11</td>
<td>1.64</td>
</tr>
<tr>
<td>1532</td>
<td>6</td>
<td>8</td>
<td>0.75</td>
<td>8</td>
<td>12</td>
<td>0.67</td>
</tr>
<tr>
<td>1675</td>
<td>5</td>
<td>6</td>
<td>0.83</td>
<td>5</td>
<td>12</td>
<td>0.42</td>
</tr>
<tr>
<td>1952</td>
<td>3</td>
<td>3</td>
<td>1.0</td>
<td>11</td>
<td>9</td>
<td>1.22</td>
</tr>
<tr>
<td>2470</td>
<td>4</td>
<td>6</td>
<td>0.67</td>
<td>3</td>
<td>12</td>
<td>0.25</td>
</tr>
<tr>
<td>2954</td>
<td>3</td>
<td>8</td>
<td>0.38</td>
<td>6</td>
<td>11</td>
<td>0.55</td>
</tr>
<tr>
<td>3106</td>
<td>3</td>
<td>6</td>
<td>0.5</td>
<td>1</td>
<td>12</td>
<td>0.08</td>
</tr>
<tr>
<td>3217</td>
<td>11</td>
<td>5</td>
<td>2.2</td>
<td>12</td>
<td>11</td>
<td>1.09</td>
</tr>
<tr>
<td>3915</td>
<td>12</td>
<td>5</td>
<td>2.4</td>
<td>20</td>
<td>10</td>
<td>2.0</td>
</tr>
<tr>
<td>5340</td>
<td>19</td>
<td>7</td>
<td>2.71</td>
<td>17</td>
<td>12</td>
<td>1.42</td>
</tr>
<tr>
<td>5842</td>
<td>8</td>
<td>9</td>
<td>0.89</td>
<td>3</td>
<td>17</td>
<td>0.18</td>
</tr>
<tr>
<td>6520</td>
<td>6</td>
<td>5</td>
<td>1.2</td>
<td>4</td>
<td>12</td>
<td>0.33</td>
</tr>
<tr>
<td>7104</td>
<td>8</td>
<td>6</td>
<td>1.33</td>
<td>16</td>
<td>12</td>
<td>1.33</td>
</tr>
<tr>
<td>7478</td>
<td>5</td>
<td>6</td>
<td>0.83</td>
<td>5</td>
<td>7</td>
<td>0.71</td>
</tr>
<tr>
<td>8056</td>
<td>17</td>
<td>8</td>
<td>2.13</td>
<td>8</td>
<td>11</td>
<td>0.73</td>
</tr>
<tr>
<td>8314</td>
<td>8</td>
<td>9</td>
<td>0.89</td>
<td>9</td>
<td>12</td>
<td>0.75</td>
</tr>
<tr>
<td>8409</td>
<td>2</td>
<td>10</td>
<td>0.2</td>
<td>1</td>
<td>8</td>
<td>0.13</td>
</tr>
<tr>
<td>9422</td>
<td>6</td>
<td>5</td>
<td>1.2</td>
<td>4</td>
<td>10</td>
<td>0.4</td>
</tr>
<tr>
<td>9873</td>
<td>1</td>
<td>7</td>
<td>0.14</td>
<td>1</td>
<td>14</td>
<td>0.07</td>
</tr>
<tr>
<td>Average</td>
<td>7.32</td>
<td>6.5</td>
<td>1.18</td>
<td>8.38</td>
<td>11.38</td>
<td>0.76</td>
</tr>
</tbody>
</table>

Table 6.2: Section two’s paragraph error rates for the half-length draft and final paper. n = 21
<table>
<thead>
<tr>
<th>Section</th>
<th>Half-length Error Rate</th>
<th>Final Paper Error Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2.13</td>
<td>1.55</td>
</tr>
<tr>
<td>3</td>
<td>1.18</td>
<td>0.76</td>
</tr>
<tr>
<td>Overall</td>
<td>1.66</td>
<td>1.15</td>
</tr>
</tbody>
</table>

Table 6.3: Summary of the two section’s average error rate for both half-length draft and the final paper. \( n = 38 \)

0.00003985. Since the p-value is less than 0.05 the null hypothesis can conclusively be rejected. Based on this the mean is not 0. The test was run again with the alternate hypothesis that the true mean is less than 0. This produced a p-value of 0.00001993. Since this also is less than 0.05, the null hypothesis is rejected and the alternate hypothesis is accepted. The true mean is less than 0 which means the students’ error rate decreased after the lesson. Based on how low the p-values were from these tests, the pedagogy is definitely associated with the decrease in error rate. This means the paragraph lesson should continue to be researched.

### 6.4 Results: Lessons

Throughout this study five lessons were taught: prompt testing, one term per concept, peer review, paragraphs and sections. These were described in the experimental design chapter. This section will go over which lessons did well, which lessons did not do as well, and reasons the lessons performed as they did. These conclusions are based on the lab responses and student interviews. Results from these interviews can be seen in Figure 6.6. In these interviews, students were encouraged to talk about multiple lessons as either helpful or unhelpful. Since many students took the opportunity to talk about multiple lessons, some of the categories will have a higher total than the number of students interviewed.
Figure 6.5: Change in error rate by class with the final paper’s error rate on the x-axis and the half-length draft’s error rate on the y-axis. The gray line shows no change in error rate. Every point above the line shows a decrease in error rate and every point below the line shows an increase in error rate. The blue points are for section 1 and the orange points are for section 3. $n = 38$
Figure 6.6: Students’ opinions on which lessons were the best, worst, and most helpful. $n = 11$
<table>
<thead>
<tr>
<th>ID number</th>
<th>Prompt Testing</th>
<th>Single Term Per Concept</th>
<th>Peer Review</th>
<th>Paragraphs</th>
<th>Sections</th>
</tr>
</thead>
<tbody>
<tr>
<td>1169</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2028</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2509</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2570</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2826</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>3076</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4349</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>4584</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5394</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5507</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5616</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5627</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5874</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5964</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>5989</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7116</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7434</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7545</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7943</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7822</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>7851</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>8342</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>9352</td>
<td></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total</td>
<td>22</td>
<td>15</td>
<td>17</td>
<td>21</td>
<td>18</td>
</tr>
</tbody>
</table>

Table 6.4: Section one lab participation by student. A dot signifies that that student submitted that lab. A line emphasizes that a student has done consecutive labs. n = 23
<table>
<thead>
<tr>
<th>ID number</th>
<th>Prompt Testing</th>
<th>Single Term Per Concept</th>
<th>Peer Review</th>
<th>Paragraphs</th>
<th>Sections</th>
</tr>
</thead>
<tbody>
<tr>
<td>1027</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>1084</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>1277</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>1532</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>1582</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>1675</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>1952</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>2470</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>2666</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>2954</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>3106</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>3217</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>3429</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>3689</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>3915</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>5340</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>5842</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>6202</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>6520</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>6581</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>7104</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>7478</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>8056</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>8314</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>8409</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>8422</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>9873</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
<td>•</td>
</tr>
<tr>
<td>Total</td>
<td>23</td>
<td>23</td>
<td>23</td>
<td>17</td>
<td>17</td>
</tr>
</tbody>
</table>

Table 6.5: Section one lab participation by student. A dot signifies that that student submitted that lab. A line emphasizes that a student has done consecutive labs. n = 27
The lab for prompt testing had students go back through their proposals and check if they were meeting specifications. If they were not, they were asked to revise their proposal and write what they had changed. Many students fixed their sections’ length or formatting as a result of this lab. Others took the opportunity to make a more complete draft with more information or citations.

Despite the improvements to the student’s proposals, the prompt testing lesson did not do well. In the interviews, many students either did not remember this lesson or mentioned that it was one of the lessons that did not help them. There are many factors that could have contributed to the poor performance of this lesson.

One factor that contributed to the prompt testing lesson’s performance is that many students already make sure to follow writing specifications. This led to many students saying the first lesson was a waste which could have also contributed to student apathy. Another contributing factor is when this lesson was taught. The goal behind teaching prompt testing first was that it would help students remember to check the specifications early and often. Unfortunately, this was not the case. There were still students who still missed key requirements from the prompt in their final paper.

The prompt testing lesson only benefited a small set of students. Students who
already check whether their papers fulfill the specifications did not benefit from this lesson. Students who do not check the specification did not seem to take the lesson to heart and continued to neglect the specification. For these reasons, this lesson should not be taught to the whole class in future studies.

If it were to be taught again it should be taught to the subset of students who do not meet the specifications early on in the class. The person teaching the lesson should work individually with students to show them how to look for parts of their papers that do not fulfill requirements. This individual approach along with a later teaching time could help make this lesson more successful.

6.4.2 One Term Per Concept

The one term per concept lab had students write at least one of their domain specific rule (DSR) sections. Most students avoided using multiple words for a single concepts. Those who did use multiple terms for the same concept were able to quickly fix it after it was pointed out to them. Students also grasped the concept behind the DSR section easily.

Since students understood how to write their DSR sections, the one term per concept lesson did very well. The concept of this lesson, has been historically difficult for students to grasp but after being taught this lesson, this group of students seemed to gain an understanding of how to choose words for research writing. As shown in Figure 6.6, the one term per concept lesson had the highest number for both, most helpful lesson and best lesson overall. Many of the students who responded this way, explained that the specifications did not make the purpose of the DSR sections clear enough and that this lesson was what helped them understand its purpose.

While going through student papers, it was difficult to find any instance of students using multiple terms for a concept or a defined term again. In previous classes,
this was one of the problems that was easiest to find. One possible reason behind why students grasped this concept so well is when the lesson was taught.

This lesson was taught before the concept was necessary for a draft. Students possibly changed their views on word choice at this time. This made it easier to continue using this practice for the rest of their paper.

Another contributing factor is that the one term per concept lesson was focused on a skill that students were sure to have. Even if they had not heard of def-use chains it is a concept they had been implementing. This lab was effective because it focused changing how students viewed a part of their writing so they could apply skills they already use on a regular basis.

Overall, the one term per concept lesson was successful for the students. It was a quick way to get them acclimated to research writing where clarity is the most important aspect for their word choice. This lesson would benefit from additional research to see if it improves student’s writing.

6.4.3 Peer Review

The peer review lab had students partner up and give feedback about each other’s paper. The students that actually sat down and spoke to each other about their problems improved their papers. However, many students did not follow the format that was taught in the lecture. In those cases, many of the papers had little to no changes even though there were obvious issues in their paper.

The peer review lesson had mixed results because not all students followed the format taught to them. Several of the student interviews mentioned how this lesson helped change their view on the usefulness of peer reviews. Other students seem to disregard this lesson entirely and did peer reviews that did not benefit their partners.
One major factor that could have affected this lesson’s performance was when the class was scheduled. As stated before, the class was in the afternoon on Fridays. This heavily affected how students did the lab related to this lesson. Students were given time to review their peer’s papers after the lecture was done. At this point, many of the students left the class and, unfortunately, many of them did not put effort into their peer review.

Looking at the student’s peer reviews it was clear that many of them just looked at PDF copies of the other’s paper and made a few notes. These peer reviews offered very little assistance when compared to the students who worked together in person for this lab.

The peer review lesson could be very useful if a minor tweak was done to it. The students who stayed and participated in the lab portion of the lesson had great conversations. Their half-length drafts changed and become clearer after working with their peer. Therefore, this lesson would benefit from students having to stay and work on it in class. If students are required to work on it in class more students might have put more effort into their peer review.

6.4.4 Paragraph

The paragraph lab had students look through their paper and find times that they had multiple concepts within one paragraph. The students who put effort into this lab had great results, their separation and explanations showed their understanding of how to separate paragraphs. Student’s paragraphs became much easier to read and understand after this lesson.

As explained previously, the paragraph lesson did very well. Student’s ability to logically separate paragraphs got better after the lesson. Since the results of this lesson have already been discussed, the rest of this subsection will explain which
factors were crucial to this lesson’s success. Even though this lesson did well, Figure 6.6 shows this lesson is one many students considered the worst lesson. The reason the students said this, was they believed they already had the skills necessary to separate paragraphs well. This lesson was very divisive because it is second for all three categories. While many students thought they already had the skills necessary to split their paragraphs, many others thought that the lesson taught them valuable skills.

The most crucial factor of the paragraph lesson is the students’ deep understanding of the single responsibility principle. Since students can already go back to their code to split it up for functionality they only needed to change the way they viewed paragraphs. This change in view allowed for easy transfer for the students.

Most students said the paragraph lab helped them. One of the interviewed students said that they knew that they were meant to separate paragraphs but didn’t really understand how to do so until this lesson. Others said it was the lesson that made the idea of transferring their computer science knowledge to writing make sense.

Overall, the paragraph lesson did very well and should be researched further. If something were to be changed about this lesson it would be when in the course it was taught. It might benefit future studies to teach this lesson earlier since it is one of the lessons with the clearest connection. This could help students see the benefits of the transfer lessons and encourage more student participation.

6.4.5 Section

The sections lab had students separate their DSR into section headings. Students who came to the lecture generally had well-separated sections. Each of them focused on one area of the overall question and showed what they were planning to explain in the paragraphs that would follow. Many of the labs from students who did not come
to lecture did not have the DSRs separated into sections. Most of these looked like students turned in their current drafts just to get credit. These drafts did not have the DSR separated which is the first sections of the paper that were not explicitly detailed in the specifications.

The section lesson is the hardest to say whether it did well. The difficulty comes with the fact that it piggy-backed on the successful paragraph lesson. Students overall did not seem to have much of a problem with separating their sections in their papers. While the lab itself did not get proper submissions, students seemed to understand how to separate their papers into sections.

It is unclear why the section lesson had such apathetic results considering the success of the paragraph lesson. If this lesson were to be taught again it be should combined with the paragraph lesson. Combining the two lessons with a heavier focus on the paragraph portion could have positive results.
Chapter 7

CONCLUSIONS AND FUTURE WORK

7.1 Future Works

This section will discuss adjustments that should be made to improve the pedagogy for future analysis. These changes focus on fixing the setbacks and other problems encountered during the study. It will not include the suggested adjustments to the lessons discussed previously in the results chapter. Future studies should also feel free to add or remove lessons as they see fit to best accommodate their individual classes.

7.1.1 Longer Study

A longer study is necessary to fully understand the effect of the pedagogy. A longer study would be able to show if there are consistent improvements for students over several classes which would contribute more data points to the existing analysis. It would also allow researchers to ask students if they continued to use the skills taught in the study at a later date. This would help reveal if students actually retained and continued using the skills they learned, or just used them for the class they were taught in.

7.1.2 Multiple Papers

Having students turn in multiple papers could help gather more accurate data. One factor that could have affected this study’s data is student procrastination. If a student procrastinated on their paper until the final paper then their final paper will definitely be better than their other drafts. Having multiple papers would largely eliminate this factor. In this scenario, data can be gathered just from students fi-
nal drafts. This would give a better, more accurate representation of the students’ abilities at the time the paper is turned in.

Another setback that could be fixed with having students write multiple papers is the lesson timing and data gathering dilemma. If all of the lessons were taught between one final draft and another there would be a complete paper before the lessons were taught and another paper after them. This way students aren’t penalized for not knowing how to do things taught in the lessons in their first paper and can learn how to do them for their second.

The last major setback that multiple papers could help fix is the lack of instructor feedback. Students would get full feedback on their papers before the next paper was due. This way students would be able to see the importance of the lessons as they are taught to them.

7.1.3 Required Participation

The biggest setback in this study was a lack of student participation. If future studies required student attendance and participation it would produce a more confident and provable result.

The main change that needs to happen to make participation required without introducing bias is to not use scores from the lesson’s labs as part of the analysis. Originally, data gathered from these labs were going to be used to further analyze results from students papers. This ended up backfiring because there were several students who turned in labs that did not show any effort. For example, one of the students handed in a PDF whose entire contents was the letter “L”.

The lectures and labs could be made required for the class as long as data isn’t gathered from the labs. This does not mean that the usefulness of the lessons can no longer be analyzed. Instead of analyzing them based off of their labs like this study
attempted to do, future studies could focus on the results in the students’ final papers and students’ opinions on which lessons helped and which did not.

7.2 Conclusions

In this paper, a pedagogy was introduced that uses the concept of transfer to help programming students write research papers. It aimed to improving students’ confidence and writing ability. The pedagogy consisted of six lessons but only five lessons were actually taught to students. These lessons focus on showing students how to transfer programming knowledge to writing.

Each lesson had varying degrees of success. The paragraph lesson and one term per concept lesson did very well while the prompt testing lesson did not do well. Many of these lessons need further research to fully understand their effect.

To test if students confidence increased students filled out an introduction and exit survey. Four of the questions from this survey were analyzed. An increase in students’ confidence in their ability to write logical and well-organized papers has been shown through statistical analysis.

The pedagogy has no impact on the students’ view of the importance of writing. The results are inconclusive on if the pedagogy improves students confidence in their writing ability and how they view themselves as writers warranting additional research.

To test if the students’ writing ability improved data was gathered through students’ paper drafts. The only useful information from these was if students separated their paragraphs correctly. From this, it was found that students got much better at separating out their paragraphs. Unfortunately, since the study did not track which students came to the associated lesson these results are also inconclusive and warrant
additional research.

In conclusion, this study shown that the pedagogy can be associated with an increase in students’ confidence in their ability to write logical, well-organized papers. It has also shown decrease in students’ paragraphs containing more than one topic. Overall this pedagogy has helped the students in these two classes in some areas. However, it requires a longer study to fully understand the benefits that these transfer lessons can have.
BIBLIOGRAPHY


APPENDICES

Appendix A

CSC 300 PAPER SPECIFICATIONS
Term Paper Specification - Basic Checklist. CSC 300, Turner, Fall 2015, Version 2E

Print NAME: _______________________________ CSC 300 section: ______ Major: ______

PAPER TITLE: _______________________________ Expected grade for paper: ______

A. Length and Format:

WORDCOUNT: _______ (Min 4000 words, exclude title, table of contents, bibliography and abstract.)
__Written in Latex (bibliography should be done in BibTex). Two column formatting required.

Organization:
__Have a Table of contents and utilize section headings as prescribed below. Pages must be numbered (except for the title page and table of contents).
** You must attach a copy of this document to the front of your paper. Check off each requirement once you verify that you have satisfied it before you hand it all in.

Style and Grammar:
__No grammar/spelling errors. Sentences simple and clear. Paragraphs have clear transitions.
__Quotations must be short, single spaced, and indented properly.
__You may hire a professional proofreader or editor, but you must identify that person in your credits with contact information.
__Conclusions are at the end of paragraphs/sections, never the beginning.
__The overall style is objective and research-based: it is based on accepted ethical rules applied to substantiated claims and verifiable facts.
__A single term is utilized for each single concept. Never use different terms to point to the same concept.

* Citations: Citations must be adequate and clear to the reader.
__Must use a standard format (Use IEEE or ACM).
__ALL facts/claims must be cited. This is a big deal. Quotations must be cited. Must use primary sources and well regarded secondary sources only. Wikipedia is unacceptable as a source.
__The paper must have a properly formatted bibliography with at least 10 annotated sources.

B. Sections, point value for sections, expectations for content:

[5%] Abstract (100 Words Max)
__One or two sentences of context about your question domain. Provide your focus question. Mention one major argument relevant to each side of the question. Write your conclusion and the major reason for it in plain language.

[10%] Facts (400 Words Max) This is not a general "introduction" - it has a very specific purpose: to bring the reader from some basic information up to your question.
__Give a very general context for the issue in one paragraph. Do not include the definitions of terms, ethical systems or laws in this section.
__Note that this section must present ONLY ENOUGH substantiated facts to raise the question for the reader (it should NOT contain all the facts needed to answer the question, additional facts will be introduced, as required, during analysis).
__The facts section should end up leading the reader to ask the question naturally - there should be a natural transition to the question (a controversy or a case ending poorly is a great way to do this).
__This section must be written in an unbiased manner - it should not indicate anything about your answer or hint at any particular conclusion.
Focus Question (A single simple sentence that asks your question.)

Focus on a particular, specific case: abstract questions are far too difficult to answer in such a short paper. The question must call for a binary answer (yes / no) that you will draw based on your research and reasoning.

Social Implications (300 words max)

The question should be followed (separately) by a few paragraphs of about 300 words maximum - explaining the why this question is worth asking. What does it mean to real people? It must contain some cited evidence of some sort. It should be focused on what this all has to do with our greater society. What effects are at stake for the rest of society here?

Others’ Arguments (300 Words Max)

Probably 3 – 5 basic arguments of about 60 words each. Citations required. You find these, you do not generate them. Summarize these arguments to be brief, simple and clear. One short paragraph each. Overall, cover the strongest arguments on both sides of your issue. Offer them as though you believe them (do not bias the way you report them to the reader). Do not use ethical principles (including the Code) in this section - they are not arguments about your focus question. You’ll use these in analysis.

Your Analysis (About 3000 Words)

Start with a section showing why you utilize the SE Code to answer your focus question. Do not merely conclude, but show the reader that the SE Code applies to you domain. Be specific.

Liberally utilize sub-headings organized by the important sub-topics you’ll analyze. Delineate your lines of reasoning for the reader: the headings/subheadings should read like an outline/overview of your analysis. An entire page of text with no subheadings to guide the reader is a danger sign.

All your analysis must be thoroughly supported by reason and logic. All claims must be supported by reputable primary sources and formal data. There should be a clear cohesiveness to the analysis such that each argument logically flows into the next and the reader can follow your reasoning with no effort. Your writing must explicitly provide substantial arguments through reason, logic and data that would gain the respect of a skeptic. It is not a "persuasive" approach. It proceeds logically from premise to conclusion, not from conclusion to justification. Opinions + Assumptions + Feelings > /dev/null (they’re not wrong, might be valuable, but do not belong in this paper). Note that a conclusion without substantial evidence/analysis is an empty opinion and has no place in this paper.

Have 1-4 distinct sections of the SE Code utilized in your analysis. General ethical systems such as Deontology, Utilitarianism, or Rawls should not used as a major section heading. These ethical systems are only to be utilized through the SE Code sections that implement them. One exception is when you determine that the analysis by appropriate SE Code sections results in a "tie" - some yield "ethical," an equal number yield "unethical." In such a case, you may break the "tie" with an appropriate general ethical system and analysis based upon it. In such a case, utilize only a few sentences to define the system, but apply it carefully to the facts of your case.

If “public good” is used, there must be significant data and analysis to support it. Simply writing “it benefits the general public because it would make many people happy” is completely insufficient. Be cautious not to do “one sided” utilitarian analysis - always consider the major tradeoffs explicitly. Further, note that Code section 1 can be central to one section of analysis only. For example, if you determine section 1.03 applies, then no other section 1 Code tenet may be used later as a section heading without advance permission based on a solid justification.

Class readings should be referenced as appropriate to support your Code analysis.
Appendix B

SURVEYS

Introduction Survey
This is the beginning survey for the experiment involving transferring programming knowledge to writing. Results from this survey will not affect your grade in the class in anyway. Please answer truthfully and to the best of your ability at this time. If you have any questions about what any of the questions are asking please ask Alanna Buss. Thank you again for participating in this study.

**Academic Writing**
First I’d like for you to consider your writing so far in your college career and answer the next 5 questions.

1. Think about your writing experiences outside of your English classes. Which of these options best describes the frequency of writing you have done in these non-English classes?

   - Almost Never
   - Occasionally
   - Often
   - Very Often

For the following questions write a number from 1 - 10 where 1 is not very confident, and 10 is extremely confident.

2. How confident are you in your ability to find computer science related sources?

3. How confident are you in your ability to cite computer science related sources?

4. How confident are you in your ability to write logical, well-organized papers?

5. How confident are you in your overall writing ability?
Importance of Writing in Your Career
Thank you for answering those questions about academic writing. The next 2 questions are about your thoughts on writing in your field.

1. Think about how important writing is for you in your desired field. Which of the following describes the frequency you believe you will have to write in this field? Explain your thoughts.

   Almost Never   Occasionally   Often   Very Often

2. On a scale from 1 - 10 where 1 is unimportant and 10 is absolutely crucial, how important do you feel writing is to your success in your field?

Overall Thoughts on Writing
Thank you for answering those questions about writing in your career. The next 2 questions are about your overall thoughts on writing.

1. On a scale from 1 - 10, where 1 is horrible and 10 is amazing, how good of a writer do you think you are?

2. On a scale from 1 - 10, where 1 is not enjoyable and 10 is extremely enjoyable, how enjoyable do you find writing?

Demographics
Thank you for answering those questions about your overall thoughts on writing. The last 3 questions are about you.

1. What is your major? (please circle)
   CSC   SE   CPE   Other (please specify):
2. What year are you? (please circle)
   Freshmen    Sophomore    Junior    Senior    Grad

3. What is your gender identity? (please circle)
   Male        Female       Other      Prefer not to specify
Exit Survey
Results from this survey will not affect your grade in the class in anyway. Please answer truthfully and to the best of your ability at this time. If you have any questions about what any of the questions are asking please ask Alanna Buss. Thank you again for participating in this study.

For all scales in this survey 1 is considered low or and 10 is considered high.

1. On a scale from 1 - 10, where 1 is horrible and 10 is amazing, how good of a writer do you think you are?

2. On a scale from 1 - 10 where 1 is unimportant and 10 is absolutely crucial, how important do you feel writing is to your success in your field?

3. How confident are you in your overall writing ability?

4. How confident are you in your ability to find computer science related sources?

5. How confident are you in your ability to cite computer science related sources?

6. How confident are you in your ability to write logical, well-organized papers?

7. Did the transfer lessons help your confidence in any of the fields above? Explain why or why not.
<table>
<thead>
<tr>
<th>Final paper</th>
<th>Half draft revised</th>
<th>Half draft</th>
<th>Proposal Revised</th>
<th>Proposal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Times a defined term was used</td>
<td>Times a paragraph explains multiple concepts</td>
<td>Times a section had multiple concepts within it</td>
<td>Length of paper</td>
<td>Number of sections</td>
</tr>
</tbody>
</table>

77
Appendix D

RUBRIC
For the final rubric I wanted to have parts of writing that we focused on during the lessons of the class. Two of the lessons were not included in this rubric: prompt testing and peer review.

Prompt testing was not included because I feel like it would be assessing the student’s writing for the class rather than checking for general improvement. Instead of this I have decided that for the general correctness of the student to compare it against the student’s final grade on the final paper.

Peer review is not something we can really measure within the papers. Since it involves two students working together to make improvements we can’t measure if they worked with a peer or just improved on the paper.

The following paragraphs will explain the parts of the writing that will be analyzed and counted for the rubric itself.

For the single term lesson there is a dimension called ‘Times a defined term was used.’ This dimension might not be important until around the half draft where the students are required to have their DSRs written and some analysis done. A defined term will be considered to be used if after its original definition that the word itself comes up again in the paper. For example if high quality is defined to be without bugs and the student uses high quality again it would be considered a time a defined term was used. At any point the student can use without bugs without it being considered a defined term being used.

For the paragraph lesson there is a dimension called ‘Times a paragraph explains multiple concepts.’ A paragraph will be considered to be explaining multiple concepts if the paragraph goes over multiple ideas in depth. For example if a paragraph explains what a bill is and what it means in this case it would be explaining multiple concepts. This is because what the bill is is not the same concept as what it means for this case in particular. Some small cases like introducing a question will be excluded assuming that the question can be gained from the sentences before it.

Times where a paragraph should have been split more than once, meaning that it contained three or more main topics within it, will be counted as the number of main topics minus one. Meaning a paragraph with four main topics will be counted as three and a paragraph with two main topics will be counted as one. Any paragraphs that only have one main topic will not be counted because they are thought to be correct. This decision was made so a paper with one large paragraph that has five main topics would not seem better in the data than a paper with two smaller paragraphs with two main
topics each. In this case the first paper would be counted as a four and the later would be counted as a two, where if we didn't count in this way the first paper would be counted as a one while the second would be counted as a two.

For the section lesson there is a dimension called ‘Times a section had multiple concepts within it.’ A section will be considered to be explaining multiple concepts if the section explains more than one concept in its entirety. For example if a section explains how a company is bug testing and how they are keeping the price of their product low it would be considered to be explaining two concepts. The example itself would be specifically for lower subsections and each section will be considered based on its level of abstraction. For example a section explaining code tenet 3.08 would be considered to be only explaining one concept as long as it doesn’t try to explain anything other than code tenet 3.08.

The following paragraphs will be explanations of the meta-data gathered from each of the papers. Any ambiguities based off their names will be explained there.

The length of paper will be counted as everything from the student’s cover page, which includes the abstract, to the last of the student’s reference pages. If the students include any figures or anything at the end of the paper in an index it will also be counted. For this metric there will only be whole numbers so even if a page is only half full it will be counted as a full page.

By having the length of the paper we can normalize the above metrics. Missing one citation, or forgetting to separate one paragraph in a 17 page paper is much less concerning than in a 3 page paper. By having this metric we can get an idea of how many errors per page were made.

The number of sections will be counted as the total number of sections the student has included. This includes any subsections of a section and any farther depth of subsections the student might have gone into.

The number of sections made can give us an insight into how the student broke up their paper. A paper with more subsections might be more dense and have more problems with the paragraph or sections metrics above. By having a number of subsections we can look into if the section metric was useful also if there is a correlation between higher end scores and the number of sections that the student used.

The number of SE code tenets used will be counted as the number of tenets that the student had decided to use on their paper. If a student uses a code tenet to prove a
different code tenet, for example code tenet 6.10 says to “Avoid associations with businesses and organizations which are in conflict with this code.” For this code to be fully explained the student would have to explain what code tenet the business or organization is not following. So code tenet 6.10 will be counted as well as the code tenet that the student uses to prove 6.10.

By taking the number of SE code tenets we can look into how much of that student’s paper was separated into. For the CSC 300 paper students must provide at least 3,000 words of analysis. This means that if a student for their final paper uses 3 sections we can assume that they have about 1,000 words of analysis for each. But if a student used 6 sections we can assume that they only used 500 which might correlate to a lower score overall.
The graphs in this appendix depict the demographics of both sections combined. They include every student who originally signed up for the study. The number of students depicted in each is 58.