6-2017

The Introduction of Informal Cooperative Learning into our Programming Laboratories

Guity Ravai
Purdue University

Ludmila Nunes
Purdue University

Ronald Erdei
Purdue University

The Introduction of Informal Cooperative Learning into our Programming Laboratories

Guity Ravai
Ludmila Nunes
Ronald Erdei

PURDUE
POLYTECHNIC
COMPUTER AND
INFORMATION TECHNOLOGY

INTRO

REDESIGNING EDUCATION

IMPACT

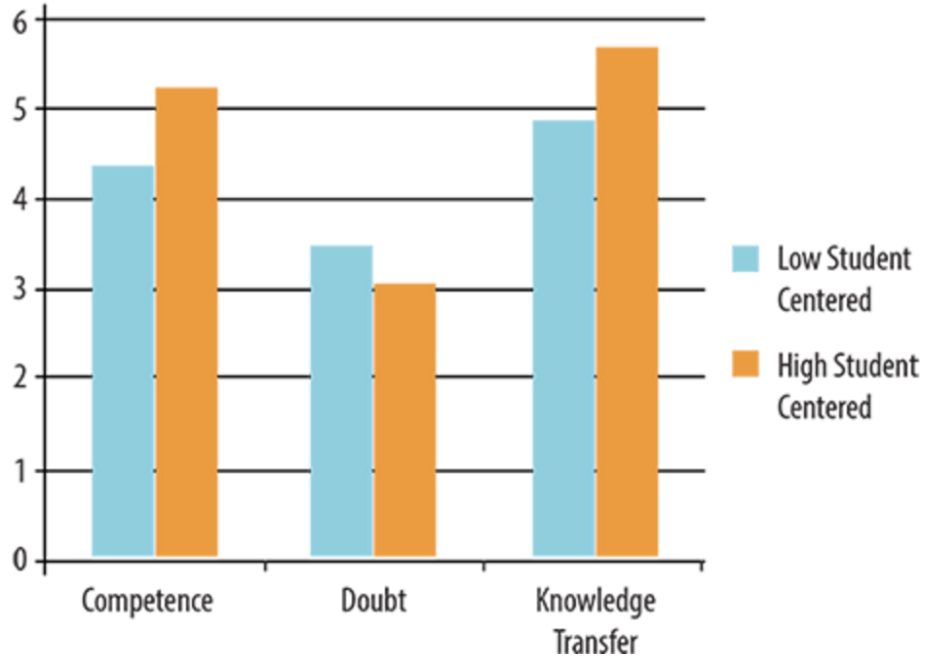
Our Mission

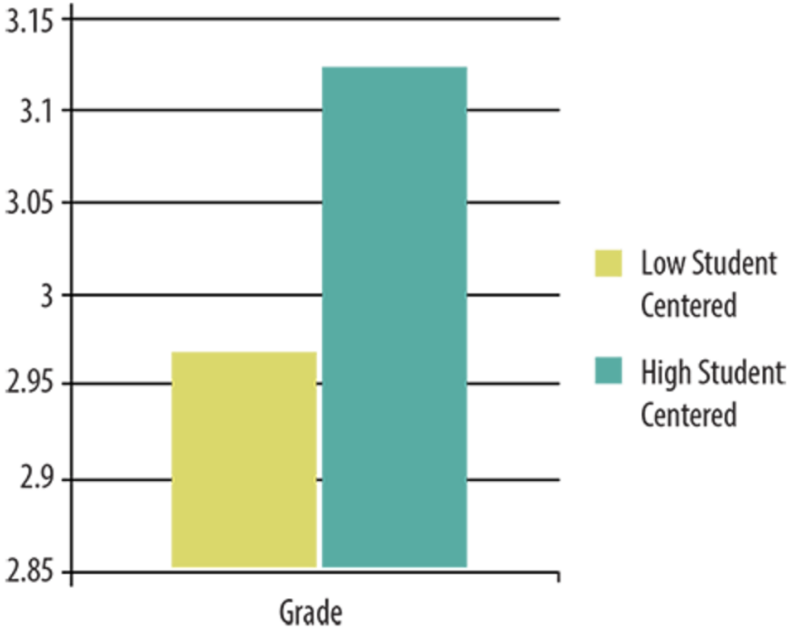
Redesign foundational courses by using research findings to create student-centered teaching and learning environments.

Our Goals

These faculty-led redesigns will foster student learning gains, student success, development of positive skills and attitudes, student well-being and institutional cultural change.

http://www.purdue.edu/impact/





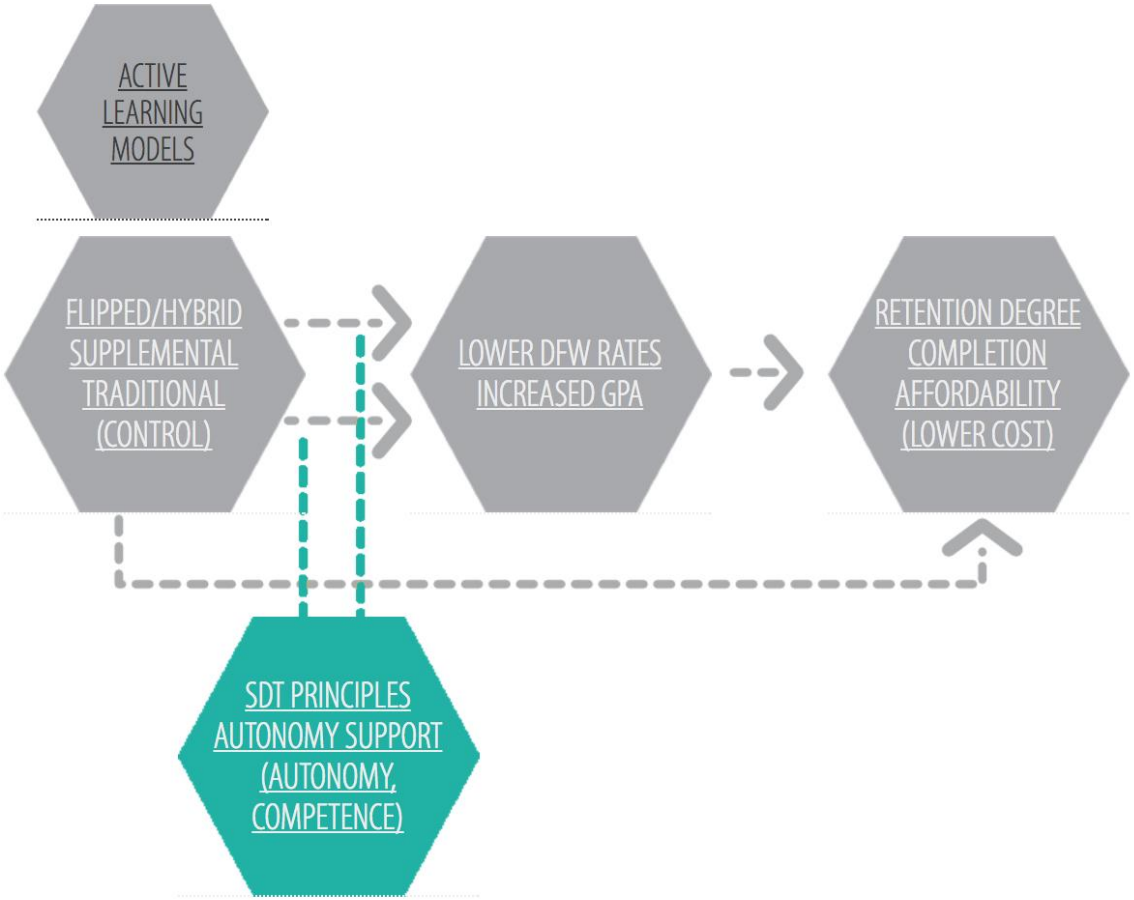
INTRO

REDESIGNING EDUCATION

IMPACT

Self-Determination Theory (SDT; Deci & Ryan, 1985; 2000)

- According to self-determination theory, self-determined motivation is motivation guiding behaviors that are valued and chosen volitionally (identification)
- Student-centered learning environments satisfy the needs for competence, autonomy (choice), and relatedness.
- Autonomy-supportive environments student motivation becomes more self-determined leading to improved learning outcomes.



FIND OUT MORE: http://www.purdue.edu/impact

Traditional Classroom

Learning to Program can be Difficult

World-wide, only 2 in 3 students enrolled in CS1 are successful

- Bennedsen and Casper (2007)
- Watson and Li (2014)

Moving Towards Student Centeredness

Structured, informal cooperation during computer labs

"**Informal cooperative learning** consists of having students work together to achieve joint learning goals in temporary, ad-hoc groups that last from a few minutes to one class period."

- Johnson et al. (2002, 2006)

The Experiment Was Done in CNIT155

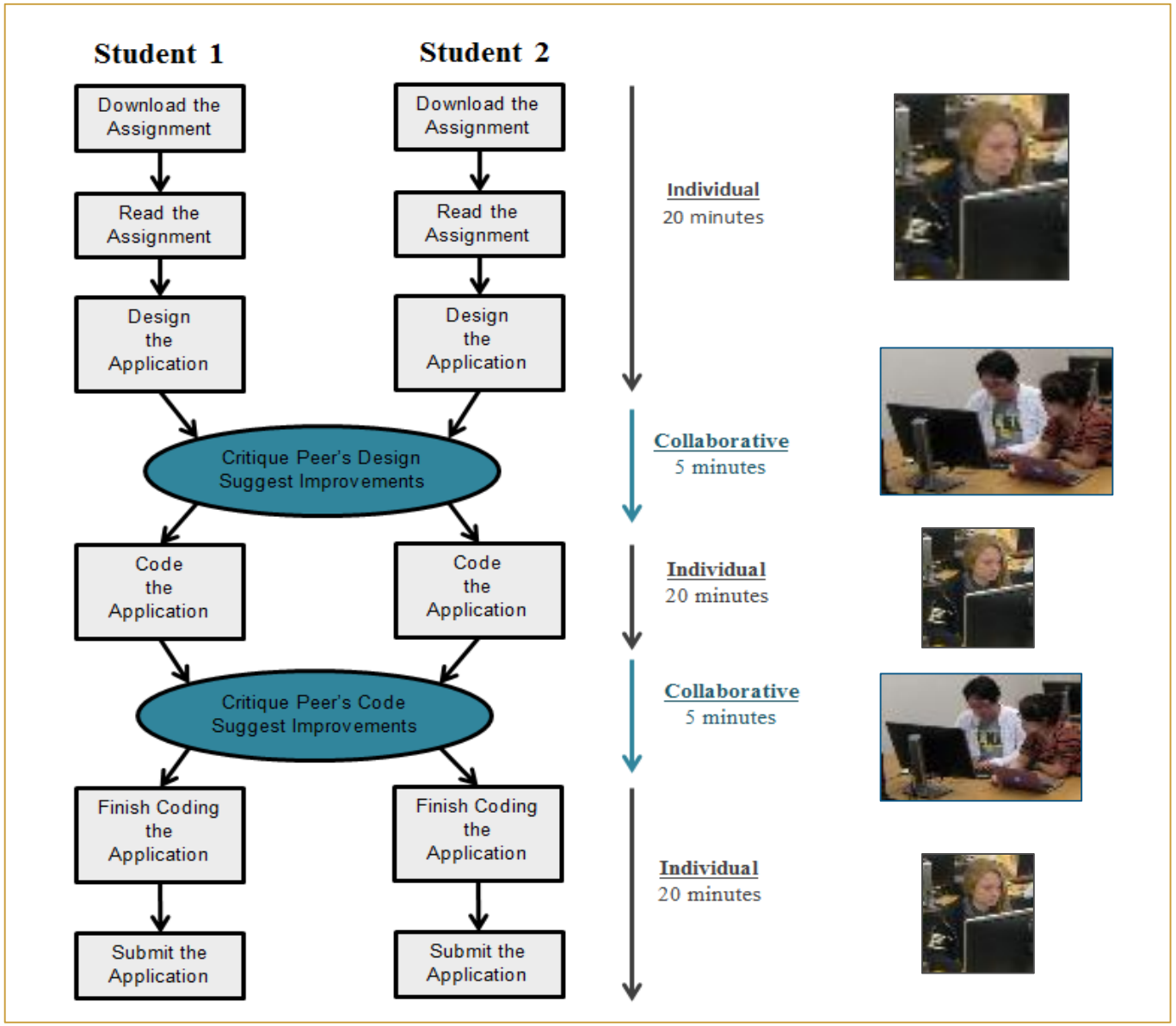
- The change was incorporated in the laboratory portion of CNIT155 “Introduction to Software Development Concepts”.
- CNIT155 is the first programming course required for students pursuing a degree in CIT (Computer Information Technology) at Purdue Polytechnic.
- The course is structured as two 50 minutes lecture and one 110 minutes lab per week.
- Students normally take this course during their Freshman Year.
- There is no prerequisite for this class.
- Most of the students are first time programmers.

Traditional Lab Structure

Learning Environment

- Most students (~ 70%) have little or no programming experience
- Avg. class (laboratory) size is 22 students
- Students work individually throughout the lab session.
- When in doubt, they raise their hand and ask the TA for assistance.
- The wait time to get help can be long.

The Collaborative Process



Students briefly work in pairs (i.e., collaborate) at strategic points during their lab session.

Hands On Activity

Let’s try this together …

- Sit in a group with few other participants.
- Work on the given problem individually (5 min.)
- Collaborate with the adjacent person to review, evaluate, discuss each other’s solutions (5 min.)
- Finally, share your solution with other people at the table.

Word Problem

- Assume there are 9 identical balls.

- One of the balls is heavier than the others.
- There is a scale to weigh the balls.

- What is the minimum number of times you have to use the scale to identify the heavier ball?

Discussion

- **What do you think?**
  - Did working together enhance your solution?
  - Did you enjoy working with others?
  - Did you feel more motivated?

Results

Learning Outcomes

Grades

Comparison Fall 2015 vs. Fall 2016
Comparison Spring 2016 vs. Spring 2017

Students self-perceptions

Comparison Pre vs. Post Fall 2016
Comparison Pre vs. Post Spring 2017

Results

Learning Outcomes

Grades

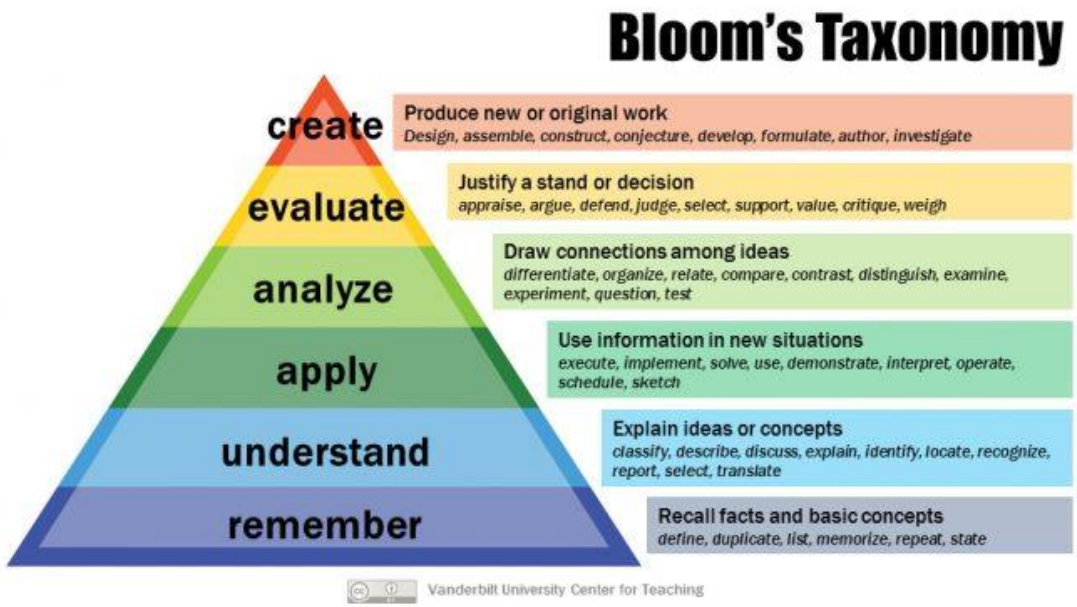
Comparison Fall 2015 vs. Fall 2016
Comparison Spring 2016 vs. Spring 2017

Students self-perceptions

Comparison Pre vs. Post Fall 2016
Comparison Pre vs. Post Spring 2017

PROCEDURE – IDENTIFYING LEARNING OUTCOMES

Learning Outcome	Description	Bloom's
LO1	Be able to employ critical thinking and problem solving – Basics of OOP – GUI Objects	1 & 2
LO2	Be able to manipulate numeric and textual data (Variable & Data Types, Input / output, arithmetic	2 & 3
LO3	Be able to interpret and employ different coding structures: Sequential, Selection, Repetition	3 & 4
LO4	Be able to modularize the program to make it more manageable (Writing helper methods to do a task).	2, 3, 4
LO5	Be able to manipulate large amount of data in the program (1-D Arrays & Files)	2, 3, 4, 5



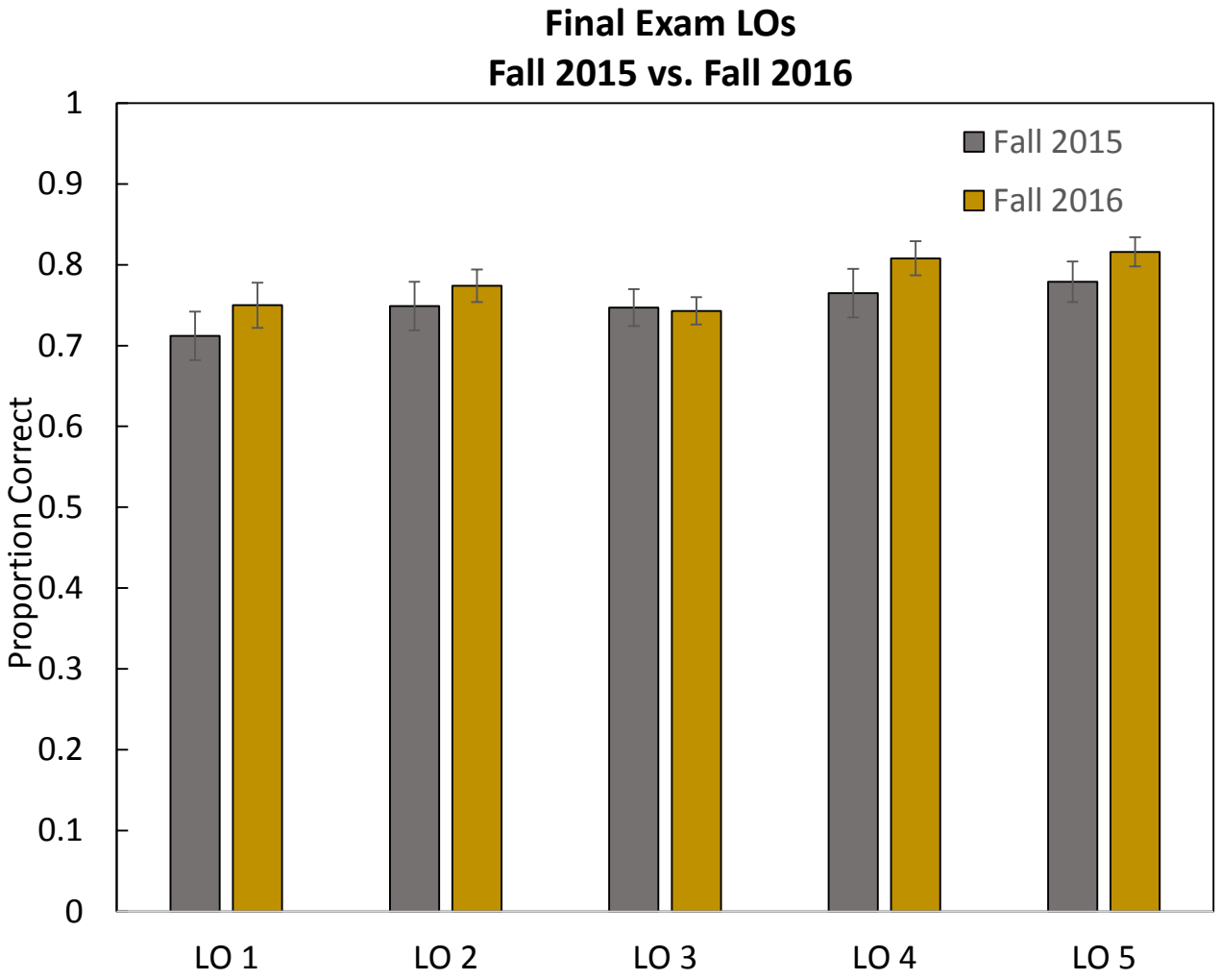
1	Remembering – Ex: Naming standards
2	Understanding – Ex: What is a Textbox
3	Applying - Ex: Calculate something
4	Analyzing – Ex: What coding structure should be used to ….
5	Evaluating – Ex: Measure the efficiency of an algorithm
6	Creating – Ex: Develop an original software

Note: Despite mapping the LOs into Bloom's levels we did not analyze gains in Bloom's because of the overlapping between levels.

PROCEDURE – MAPPING EXAM QUESTIONS INTO LEARNING OUTCOMES

No	Questions	L. Outcome
1	The extension of the source file is --------.	1
2	If a program runs without generating any errors, but produces wrong results, it probably contains -----	2
3	1. int.parse( ) is used to:	2
4	1. Which data type provides the most decimal place precision?	2
5	Which is the best data type to store the number of books	2
6	1. Which one is a comment in C#?	1
7	Which of the following statements will clear the *listbox* ?	3
8	In a C# Windows application, what happens if you delete	1
9	What value is stored in variable *answer* after executing the	2
10	What will be displayed if the user clicks on the button 5 times	4
11	Which of the following IS a valid name for a variable?	2
12	What is the 3 letter prefix for naming a ComboBox?	1
13	What value will be stored in *ans* by the following statements?	2
14	What is stored in `num` by executing the following code?	2
15	Which statement will perform a real division? Assume:	2
16	Which property of the Radio Button indicates if it has been selected?	3
...	...	...

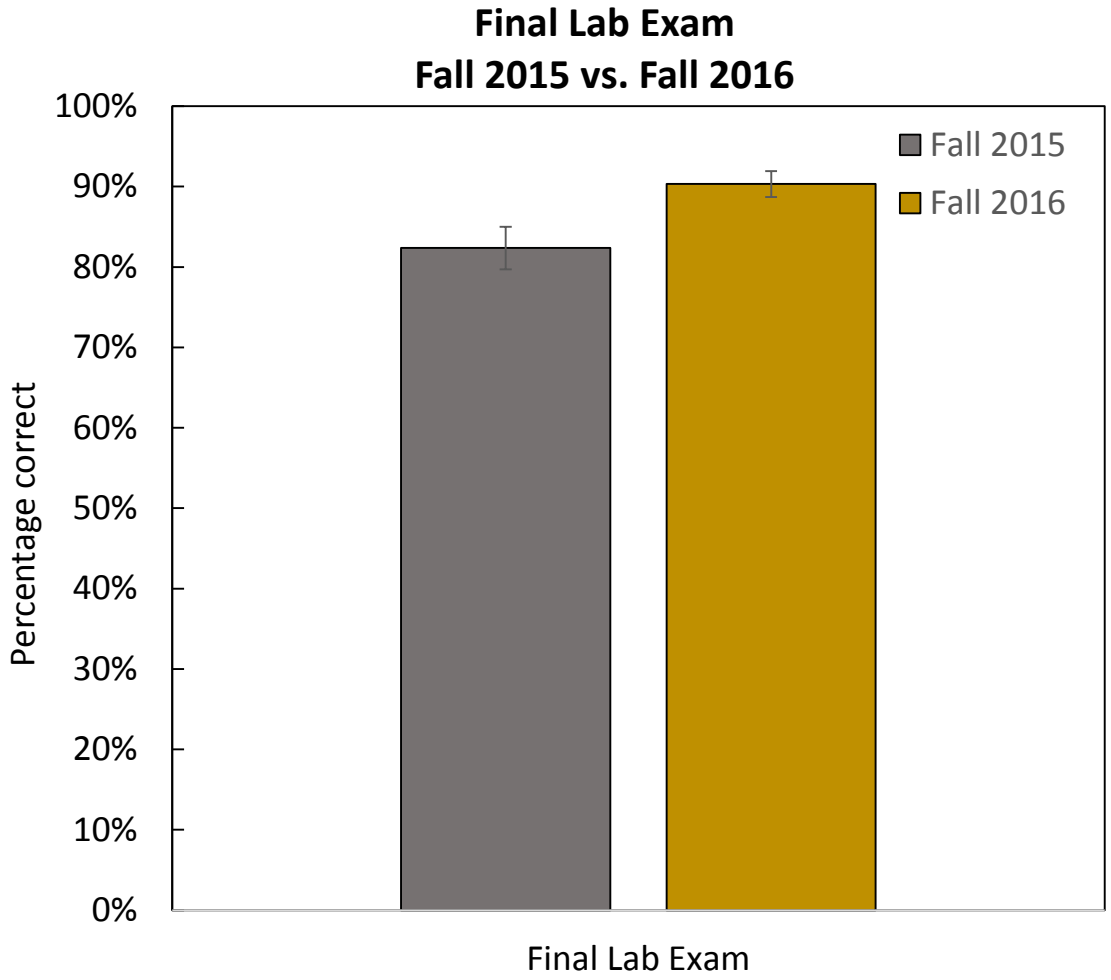
FINAL EXAM – Fall 2015 vs. Fall 2016



Learning Outcome	Description
LO1	Be able to employ critical thinking and problem solving – Basics of OOP – GUI Objects
LO2	Be able to manipulate numeric and textual data (Variable & Data Types, Input / output, arithmetic
LO3	Be able to interpret and employ different coding structures (Sequential, Selection, Repetition) - Data Validation
LO4	Be able to modularize the program to make it more manageable (Writing helper methods to do a task).
LO5	Be able to manipulate large amount of data in the program (1-D Arrays & Files)

- No significant gains between Fall 2015 and Fall 2016
- Numerical gains in all outcomes (except LO 3)

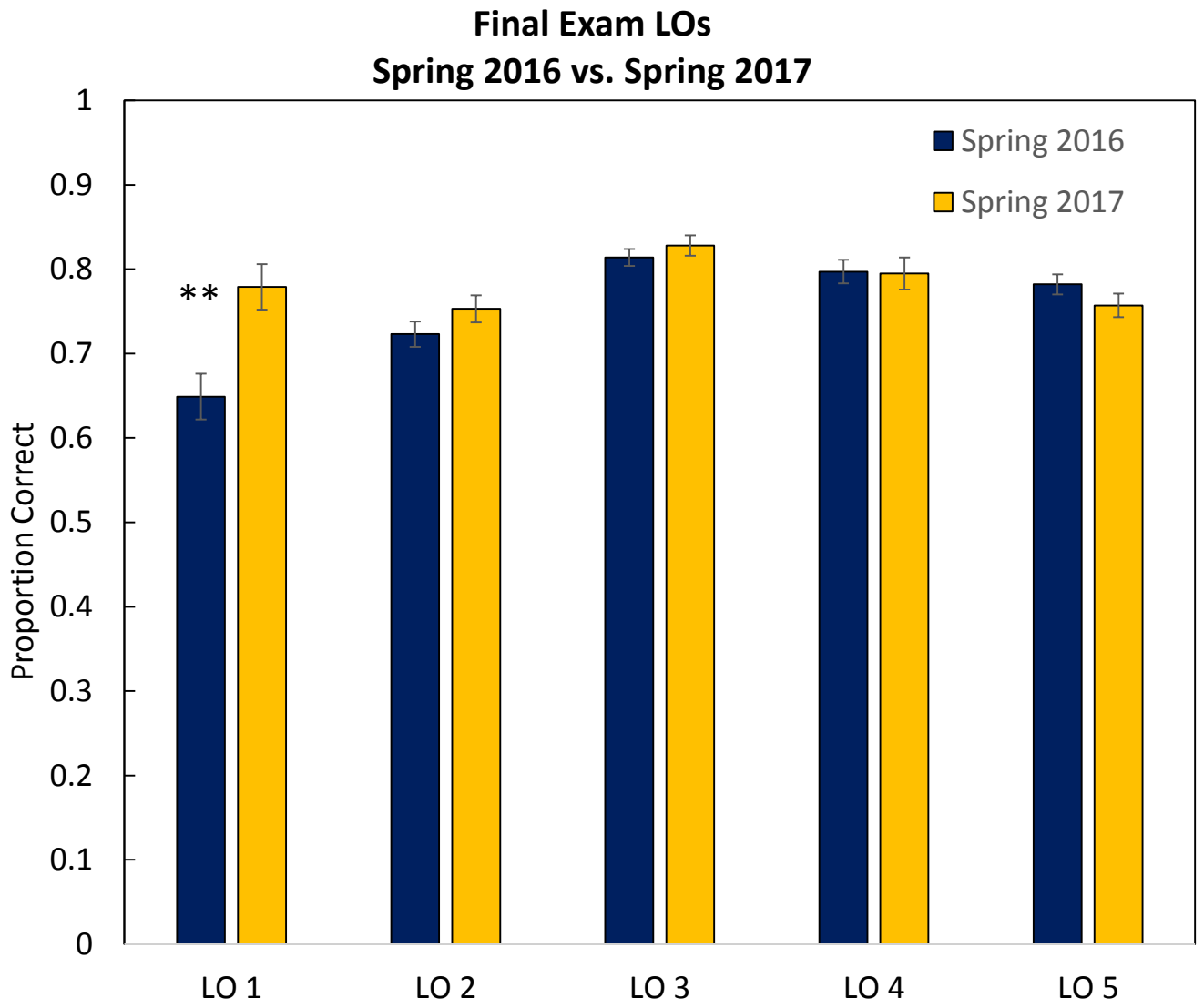
LAB EXAM – Fall 2015 vs. Fall 2016



- Significant gains in the lab exam between Fall 2015 and Fall 2016

  $t(93) = 2.703$, $p = 0.008$, $d = 0.56$

FINAL EXAM – Spring 2016 vs. Spring 2017

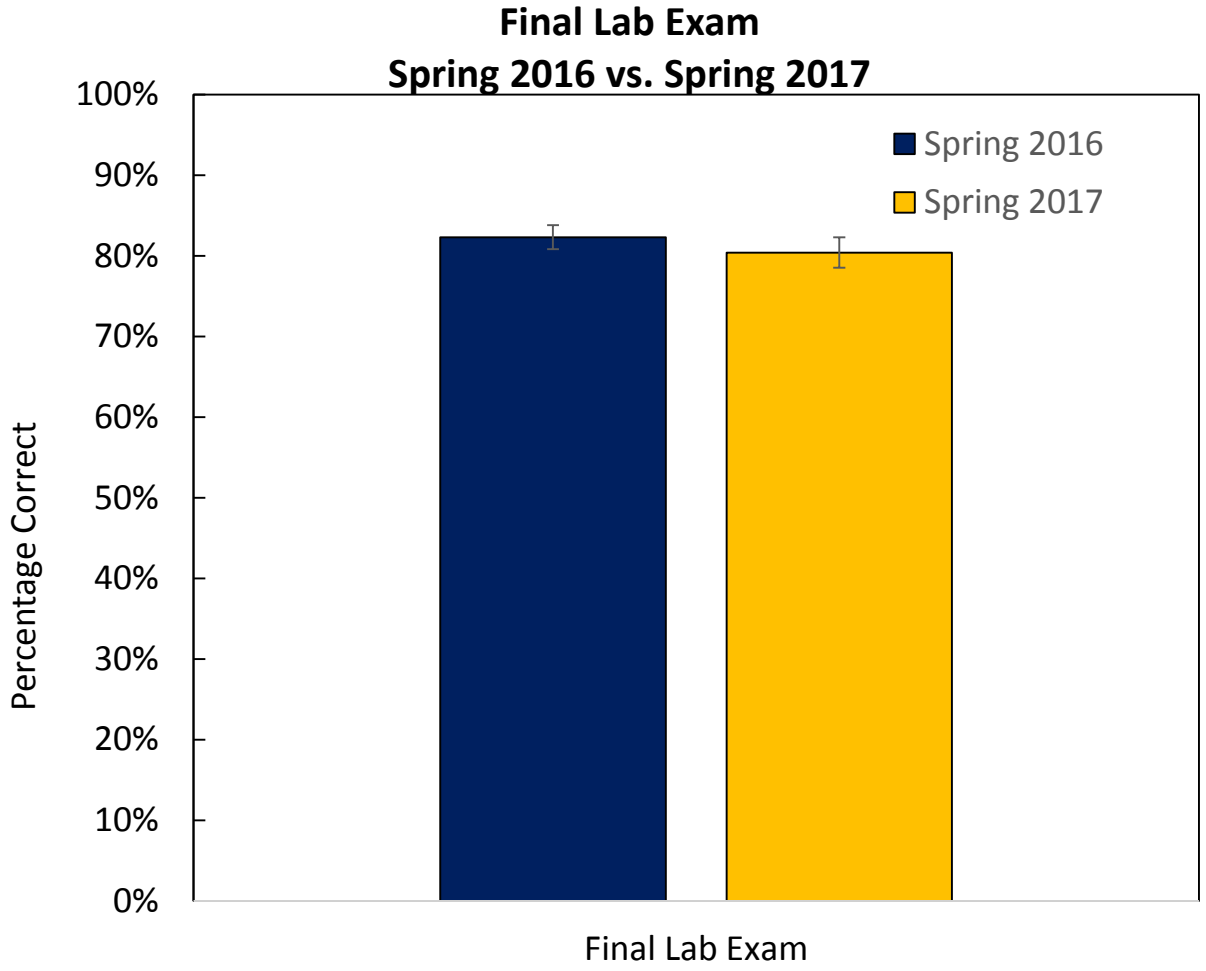


Learning Outcome	Description
LO1	Be able to employ critical thinking and problem solving – Basics of OOP – GUI Objects
LO2	Be able to manipulate numeric and textual data (Variable & Data Types, Input / output, arithmetic
LO3	Be able to interpret and employ different coding structures (Sequential, Selection, Repetition) - Data Validation
LO4	Be able to modularize the program to make it more manageable (Writing helper methods to do a task).
LO5	Be able to manipulate large amount of data in the program (1-D Arrays & Files)

- Numerical gains in all outcomes (except LO 4 and 5)
- Significant gains in LO 1 between Spring 2017 and Spring 2017

LO 1: $t(250) = 3.335$, $p = 0.001$, $d = 0.42$

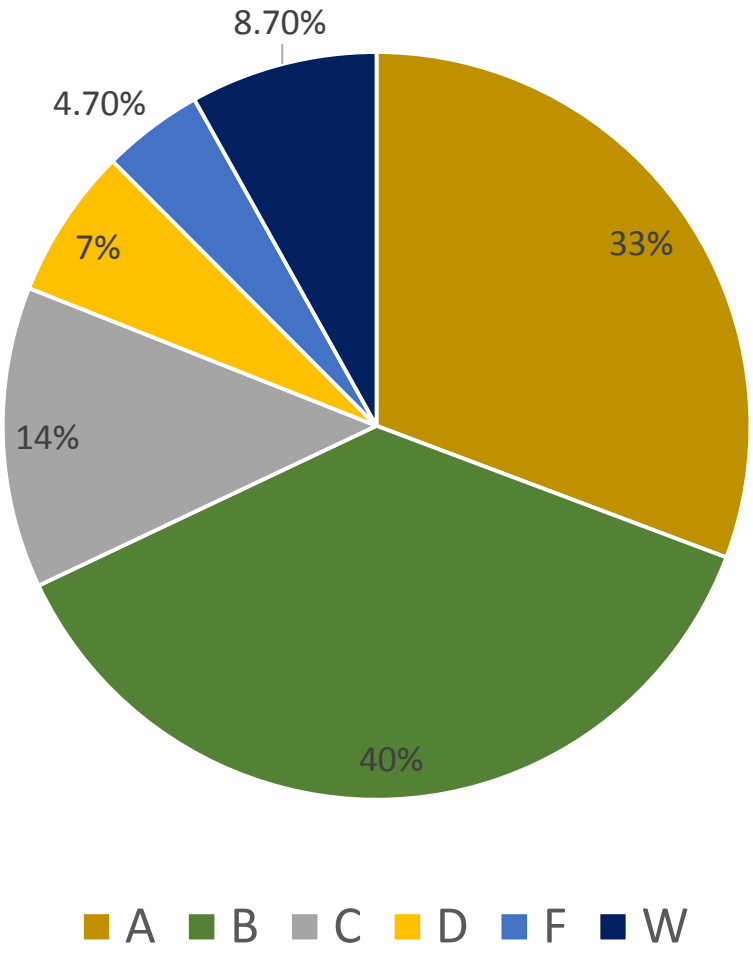
LAB EXAM – Spring 2016 vs. Spring 2017



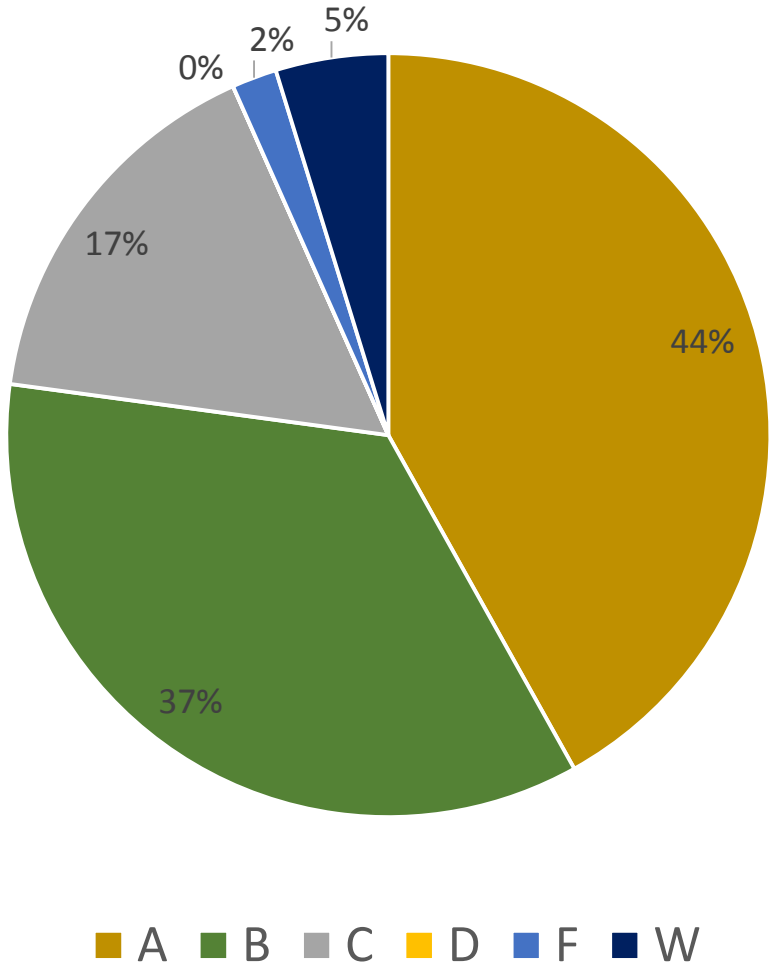
- No gains in the lab exam between Spring 2016 and Spring 2017

GRADE DISTRIBUTION – Fall 2015 vs. Fall 2016

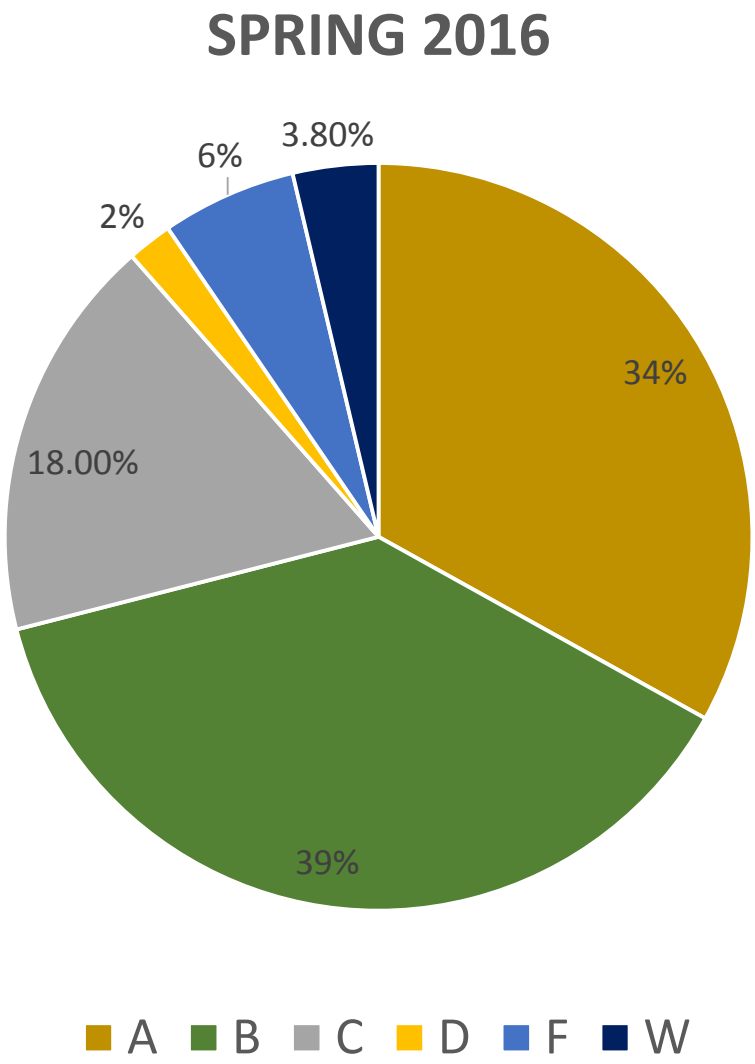
FALL 2015

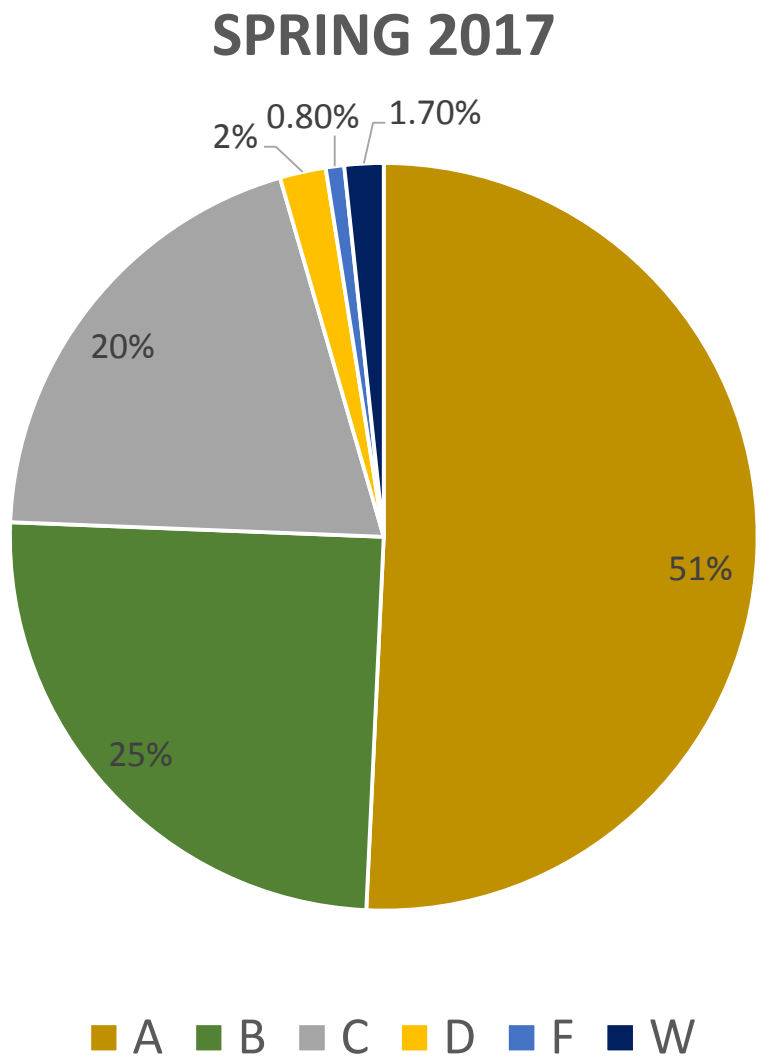


FALL 2016



GRADE DISTRIBUTION – Spring 2016 vs. Spring 2017



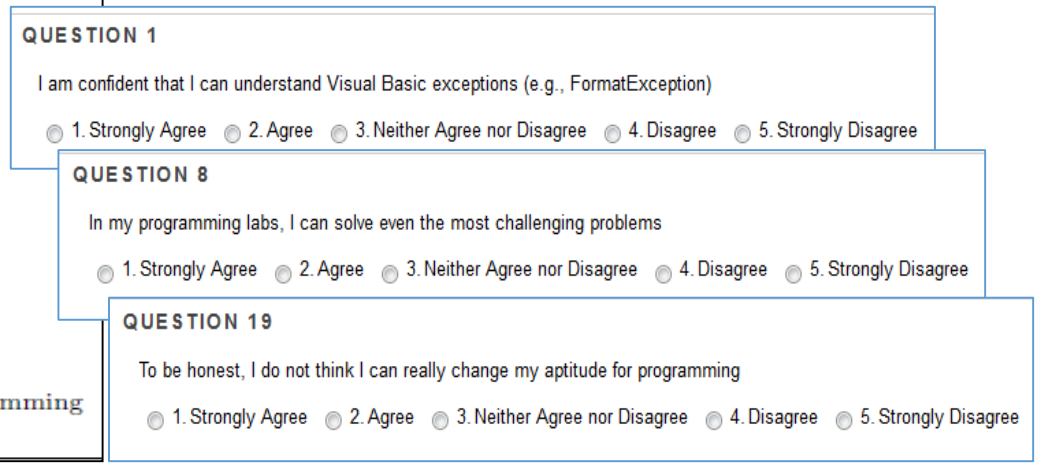


SURVEYS – STUDENT PROGRAMMING SELF-EFFICACY & SELF-BELIEFS

Item	Item Description
	Debugging Self-Efficacy
DSE1	I am confident that I can understand Java exceptions (e.g., NullPointerException)
DSE2	I am confident I can solve simple problems with my programs
DSE3	I am confident I can implement a method from a description of a problem or algorithm
DSE4	I am confident I can debug a program that calculates prime numbers
	Programming Self-Concept
PSC1	I am just not good at programming
PSC2	I learn programming quickly
PSC3	I have always believed that programming is one of my best subjects
PSC4	In my programming labs, I can solve even the most challenging problems
	Programming Interest
INT1	I enjoy reading about programming
INT2	I do programming because I enjoy it
INT3	I am interested in the things I learn in programming classes
INT4	I think programming is interesting
	Programming Anxiety
ANX1	I often worry that it will be difficult for me to complete debugging exercises
ANX2	I often get tense when I have to debug a program
ANX3	I get nervous when trying to solve programming bugs
ANX4	I feel helpless when trying to solve programming bugs
	Programming Aptitude Mindset
APT1	I have a fixed level of programming aptitude, and not much can be done to change it
APT2	I can learn new things about software development, but I cannot change my basic aptitude for programming
APT3	To be honest, I do not think I can really change my aptitude for programming

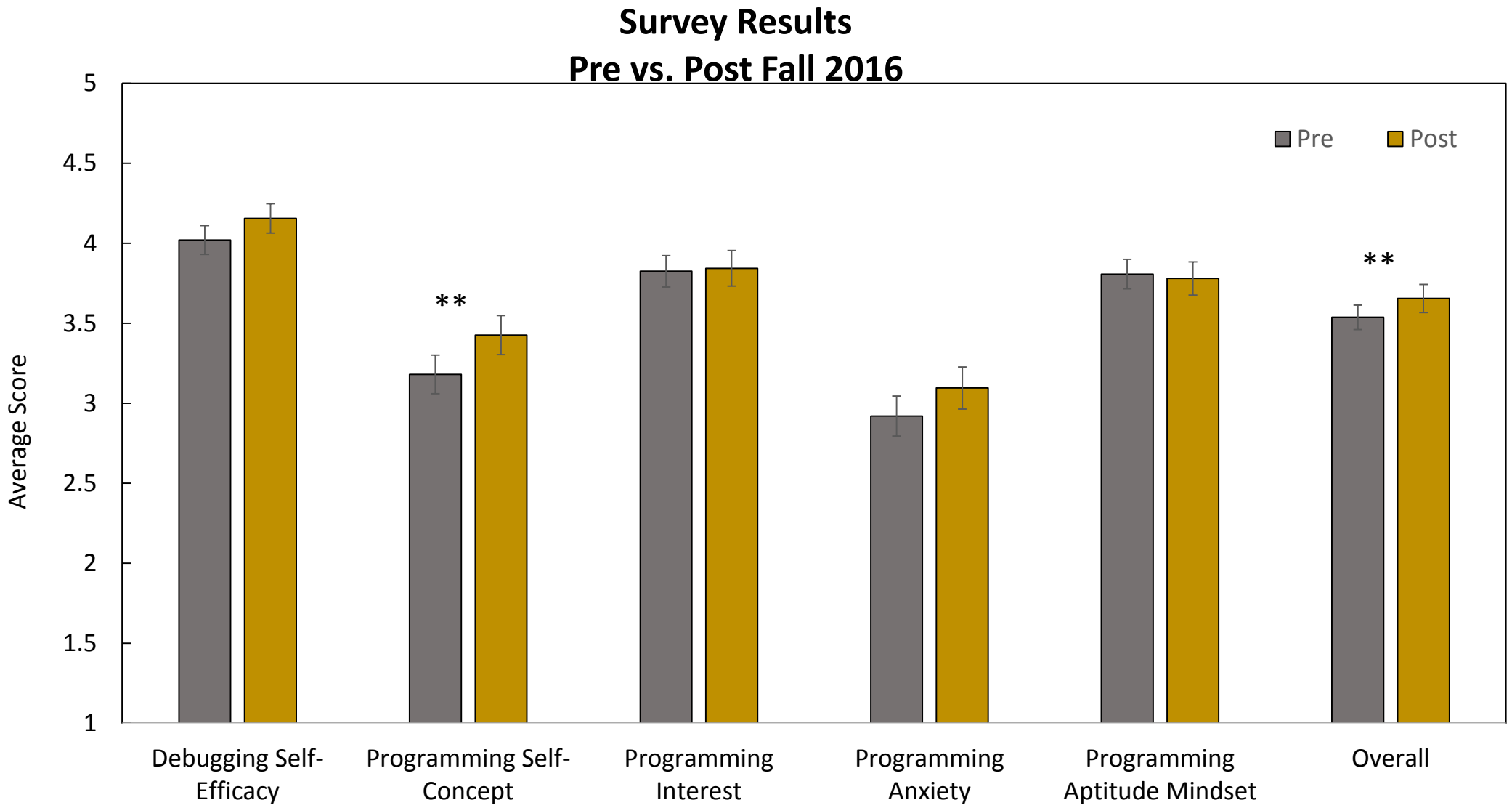
Scott & Ghinea (2014) instrument adapted for use in the specific context of this course.

Responses were given in 5 point scales and higher values mean more desirable beliefs (some items were reversed to reflect this).

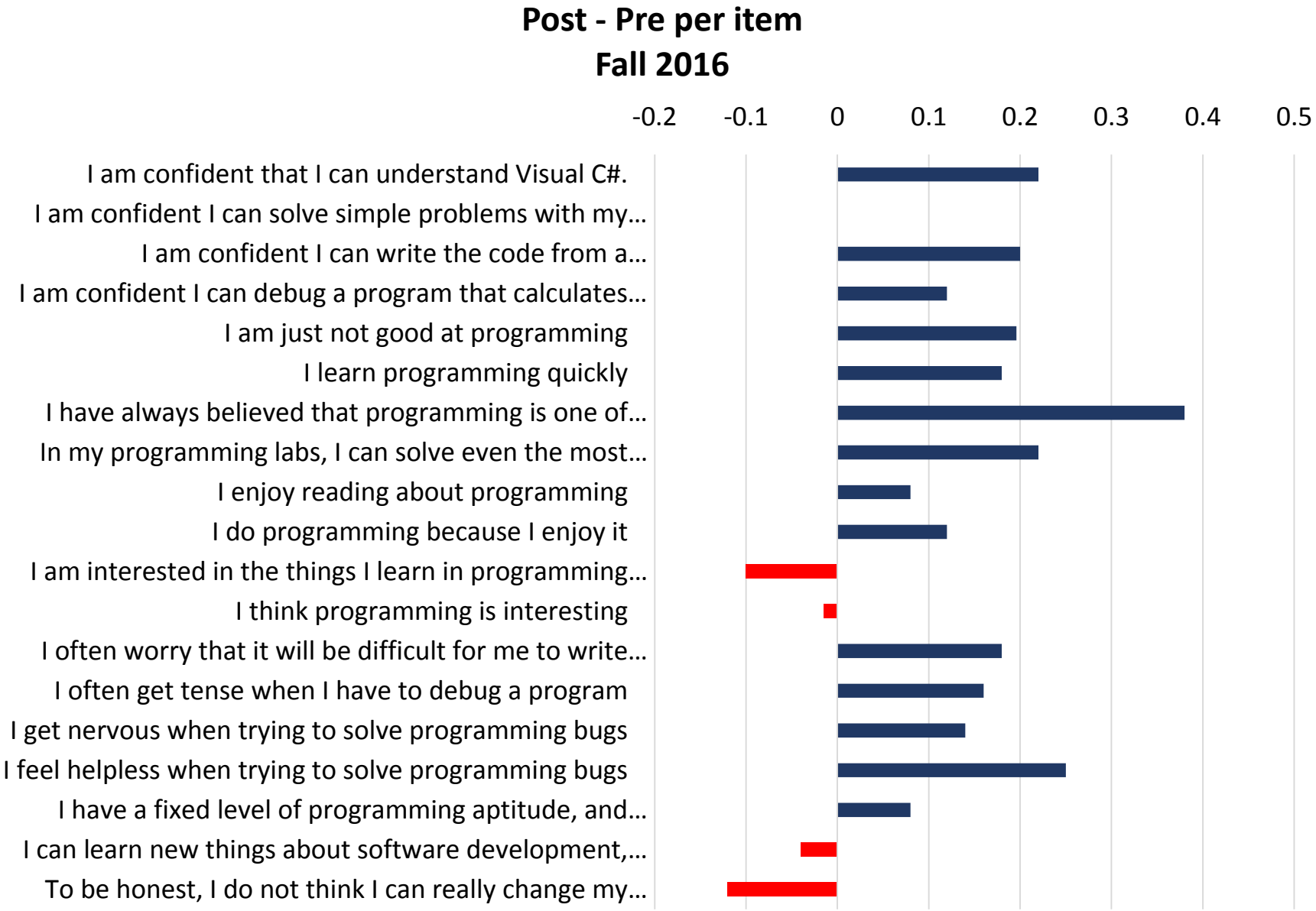


SURVEY – Pre vs. Post Fall 2016

PSC: t(49) = 3.074, p = 0.003, d = 0.44
Overall: t(49) = 2.096, p = 0.041, d = 0.30

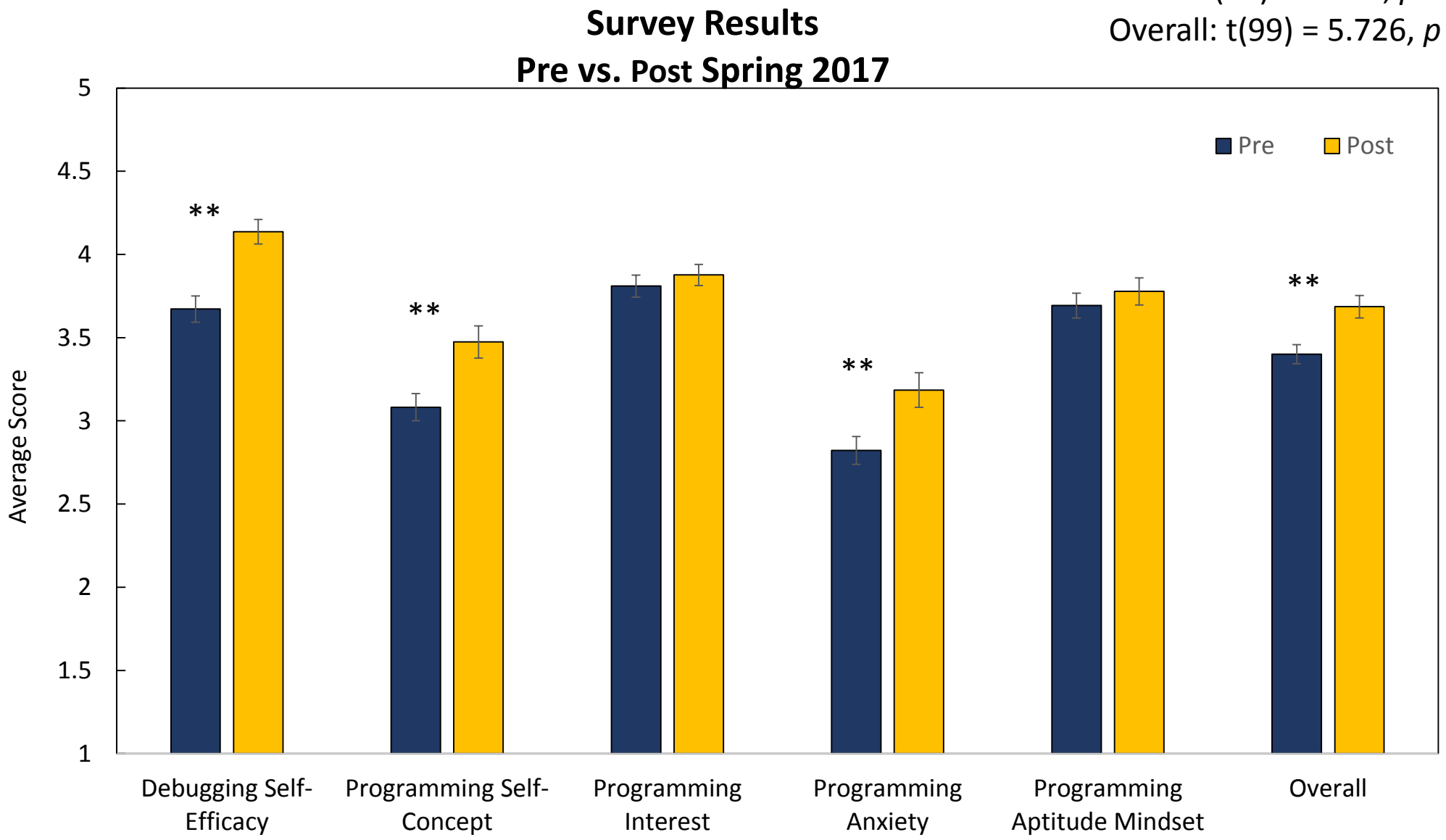


SURVEY – Gains from Pre to Post - Fall 2016

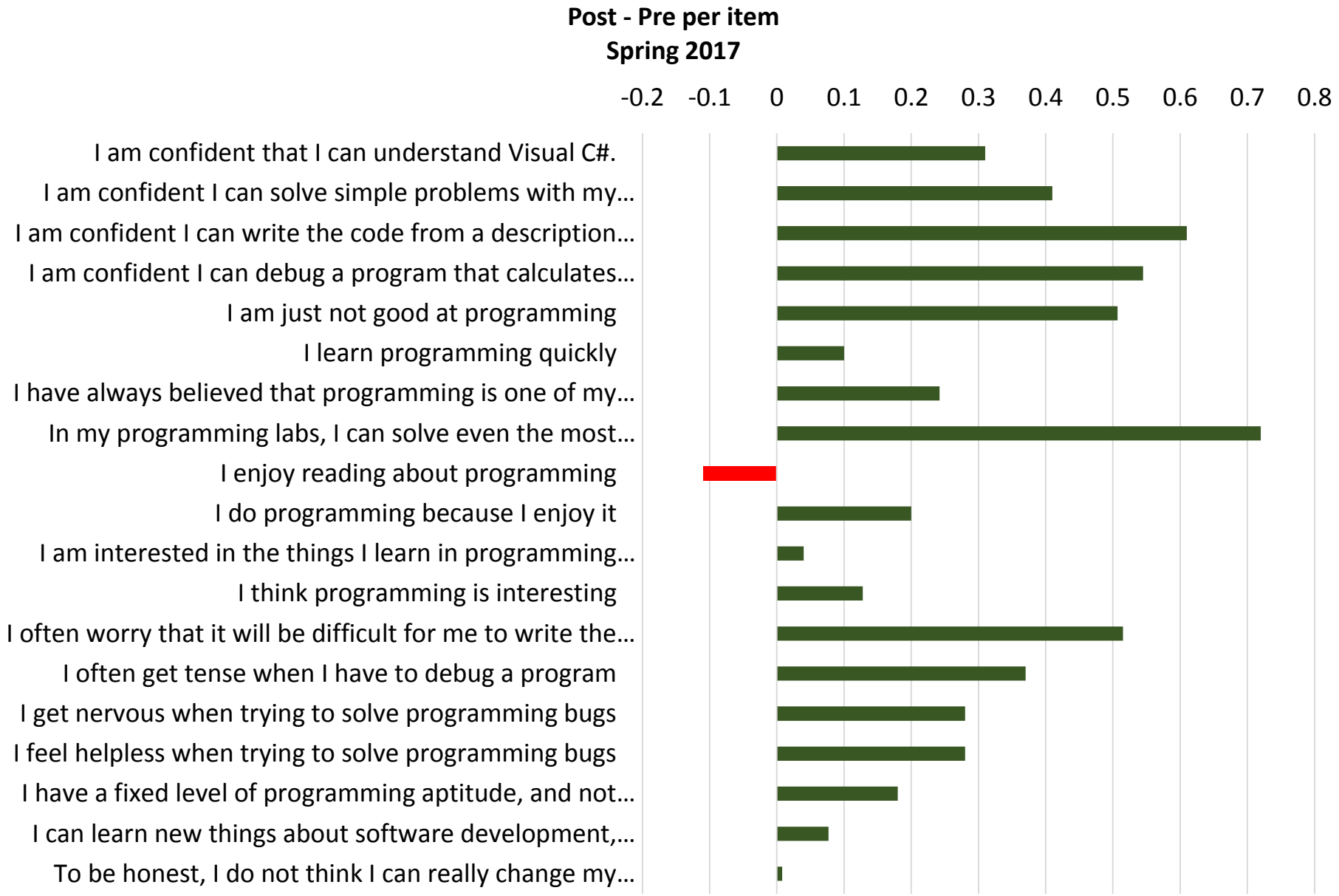


SURVEY – Pre vs. Post Spring 2017

DSE: t(99) = 6.604, p = 0.001, d = 0.66
PSC: t(99) = 5.262, p = 0.001, d = 0.53
PANX: t(99) = 4.195, p = 0.001, d = 0.42
Overall: t(99) = 5.726, p = 0.001, d = 0.57



SURVEY - Gains from Pre to Post - Spring 2017



Focus Groups

Focus Groups

Themes

- Forcing students to cooperate helped them meet their peers. It “gave them permission” to interact with their peers, something traditionally prohibited in their experience.
- Student confidence increased when they were able to help someone else
- Student confidence increased when they realized others were experiencing the same difficulties they were
- Seeking peer assistance was faster/easier than seeking instructor assistance
- By the end of the semester, students no longer followed the prescribed schedule. They sought assistance from their peers whenever they needed it.
- Students reported that it became natural to assist each other in their other classes. That is, the cooperative relationships they formed in this class transcended this class.

Results - Summary

- Students performance significantly improved in some LOs, although findings were inconsistent. Numerical gains were replicated though.
- Letter grades D, F, W decreased and As increased
- Importantly, students self-beliefs about programming improved
- Programming interest does not seem to have improved
- However, students report increased confidence after engaging in collaboration with their peers and overall enjoyed the lab format.

Instructor’s Observations

Based on our study, we observed that students benefit from informal collaboration. Some of the benefits are:

- Reduced anxiety
- Sense of community
- Higher self belief
- Sense of enjoyment while programming
- Less dependence on the teacher

Discussion

Per NSF report, the number of women graduating from CS discipline decreased from 28% to 18% between 2002 and 2011.

- Would methods like ours improve women's retention and success in Computing fields?

- What else can be done to increase women's success in CS?

Researchers

Guity Ravai
Purdue University
Clinical Assistant Professor
Computer and Information Technology
Email: guity@purdue.edu
Phone: 765-430-2573

Ronald Erdei
University of South Carolina Beaufort
Assistant Professor
Mathematics and Computational Science
Email: erdei@uscb.edu

Ludmila Nunes
Purdue University
Postdoctoral Research Associate
Center for Instructional Excellence
Email: nunes@purdue.edu

Sahithya Kodam
Purdue University
Graduate Student
Computer and Information Technology
Email: kodam@purdue.edu

References

- Bennedsen, J., & Caspersen, M. E. (2007). Failure rates in introductory programming. *ACM SIGCSE Bulletin*, *39*(2), 32–36. http://doi.org/10.1145/1272848.1272879
- Bransford, J. D., Brown, A. L., & Cocking, R. R. (2000). *How people learn : brain, mind, experience, and school*. Washington, D.C.: National Academy Press.
- Johnson, D. W., & Johnson, R. T. (2008). Cooperation and the Use of Technology. In J. M. Spector (Ed.), *Handbook of research on educational communications and technology* (3rd ed., pp. 785–811). New York: Lawrence Erlbaum Associates.

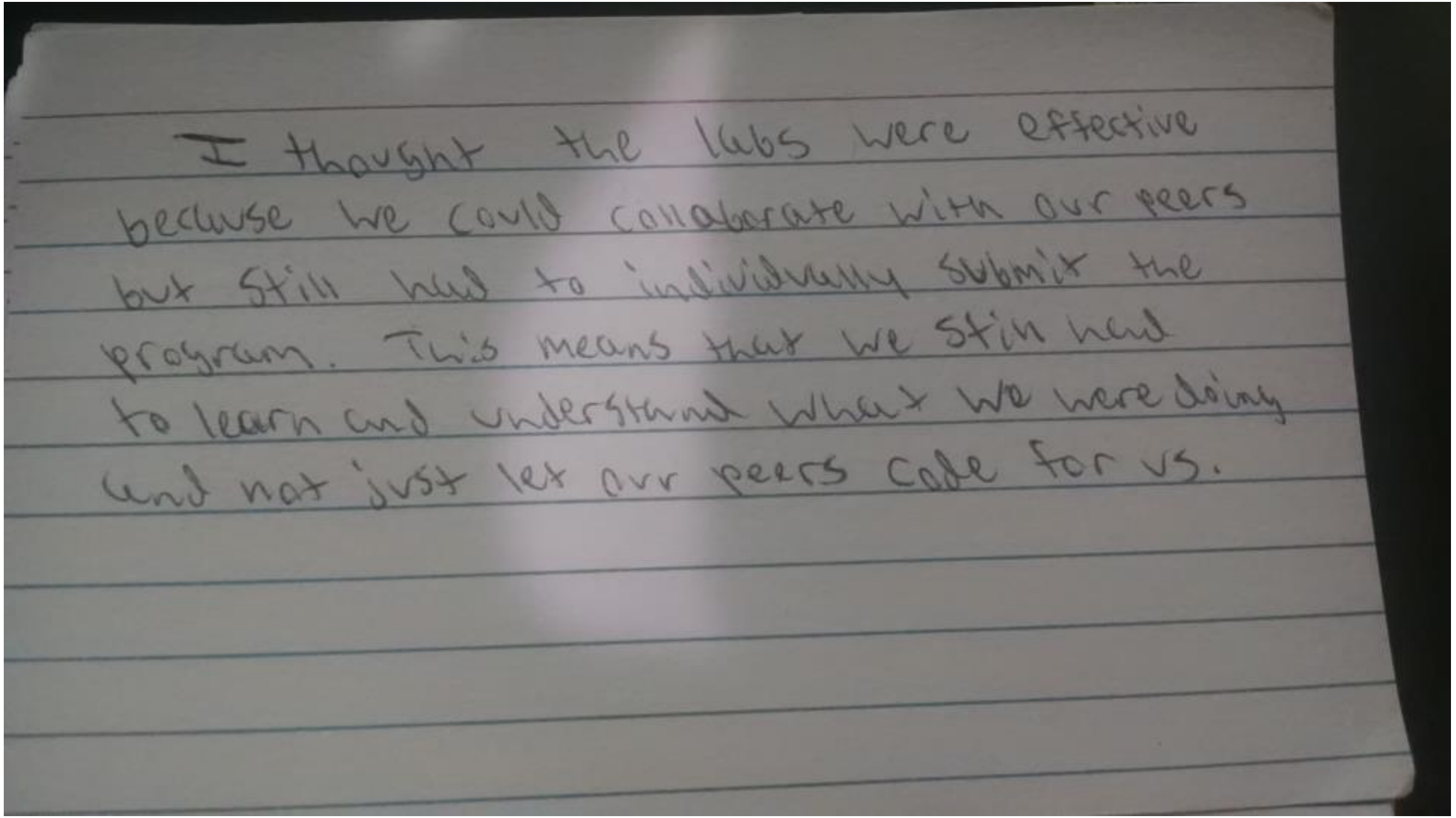
References

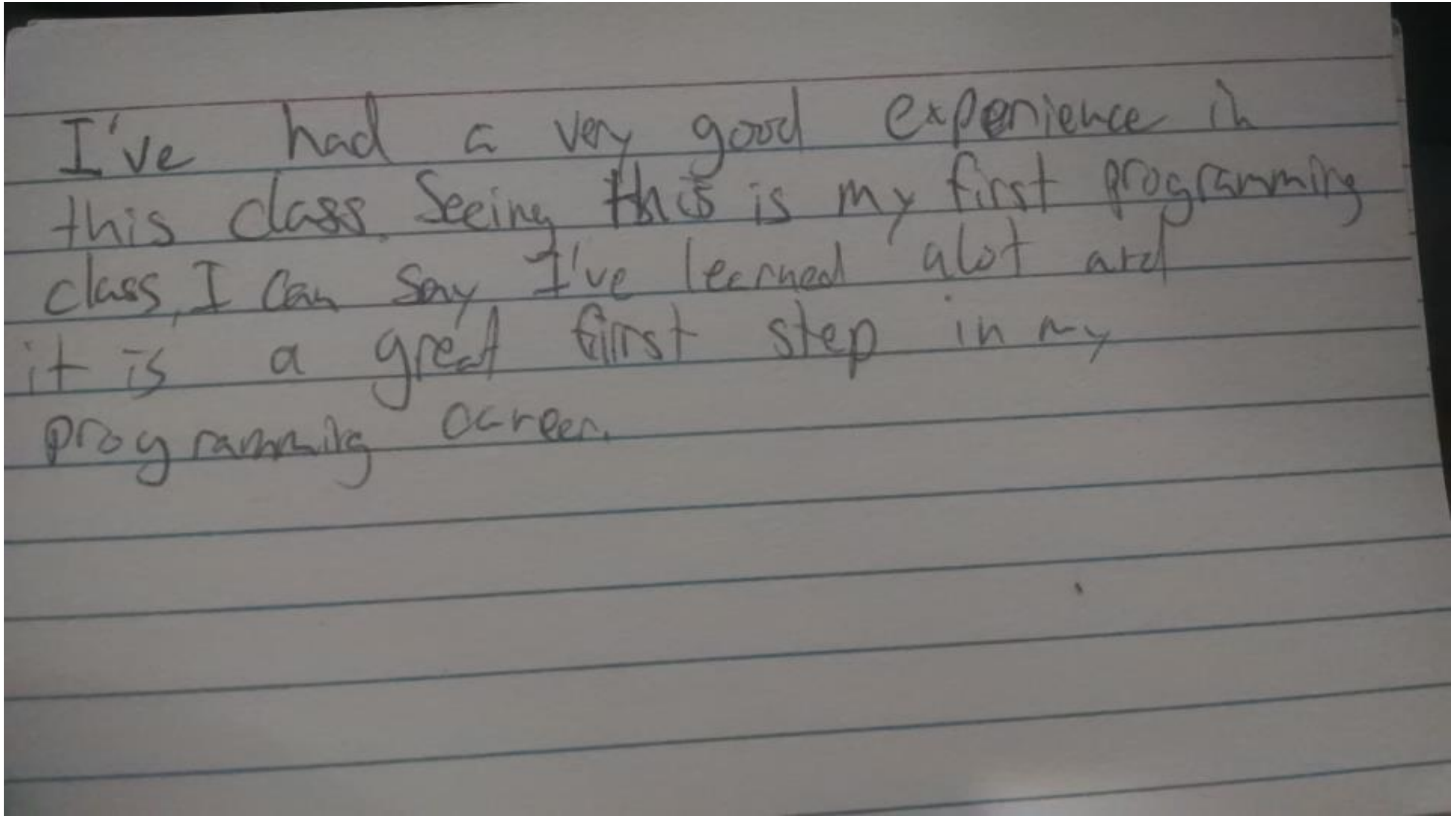
- Scott, M. J., & Ghinea, G. (2014). Measuring enrichment: the assembly and validation of an instrument to assess student self-beliefs in CS1. In *Proceedings of the tenth annual conference on International computing education research* (pp. 123–130). New York, New York, USA: ACM. http://doi.org/10.1145/2632320.2632350
- Watson, C., & Li, F. (2014). Failure rates in introductory programming revisited. In *Proceedings of the 2014 conference on Innovation & technology in computer science education* (pp. 39–44). ACM. http://doi.org/10.1145/2591708.2591749

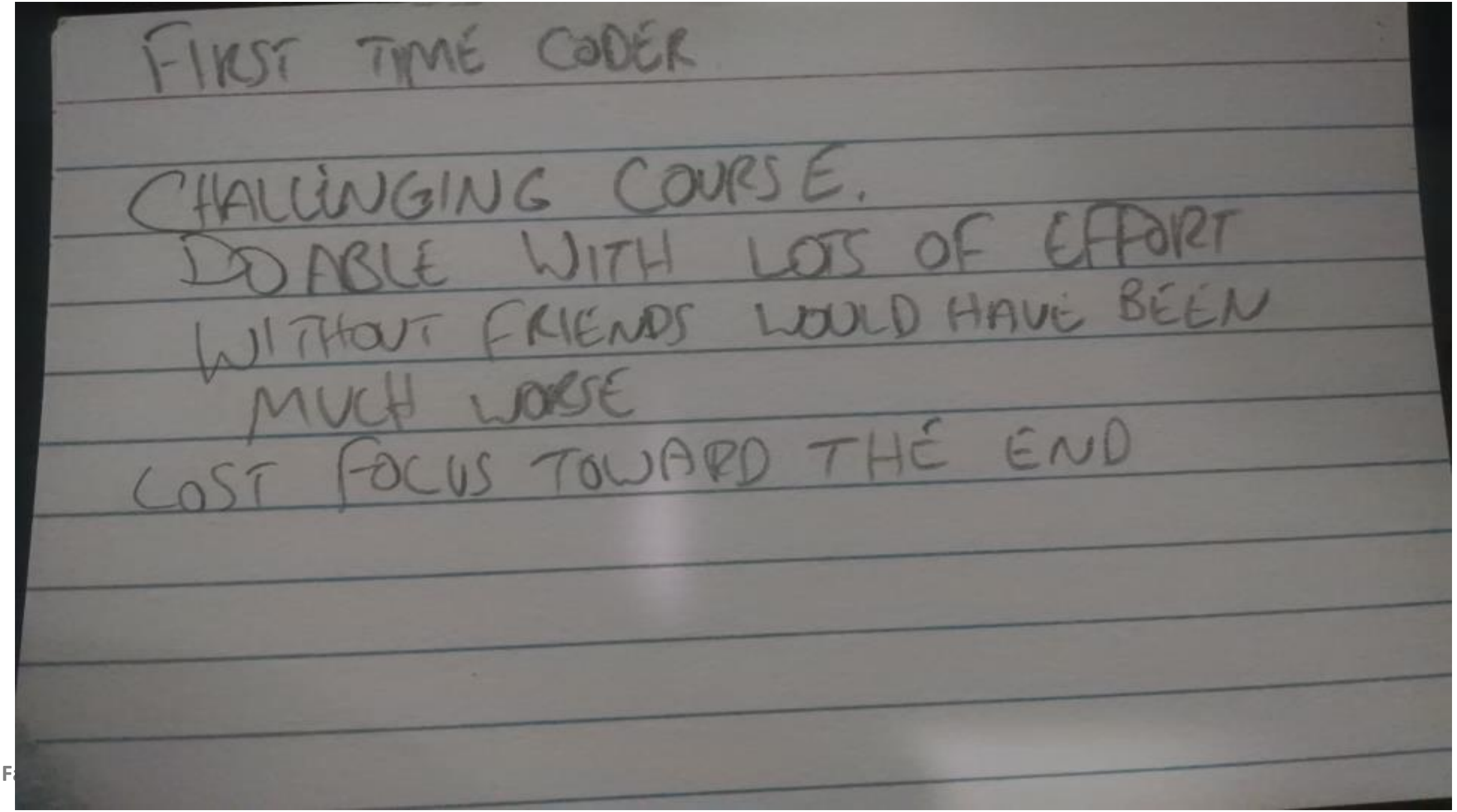
Gallery: Students working Individually

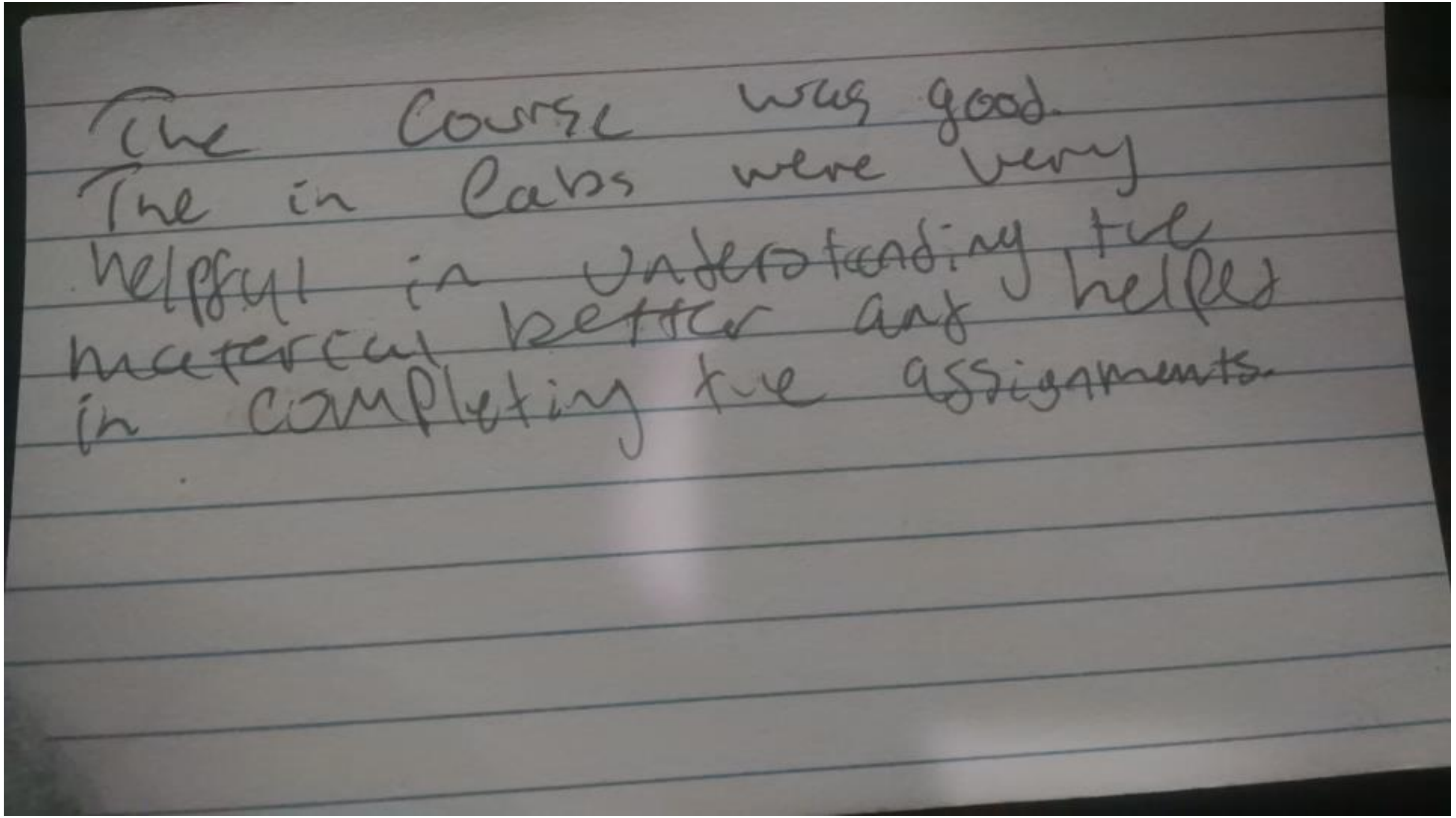
Gallery: Informal Cooperation

Gallery: Lab Instructor Assistance


I thought the labs were effective
because we could collaborate with our peers
but still had to individually submit the
program. This means that we still had
to learn and understand what we were doing
and not just let our peers code for us.


I've had a very good experience in this class. Seeing this is my first programming class, I can say I've learned alot and it is a great first step in my programming career.


FIRST TIME CODER
CHALLINGING COURSE.
DOABLE WITH LOTS OF EFFORT
WITHOUT FRIENDS WOULD HAVE BEEN
MUCH WORSE
LOST FOCUS TOWARD THE END


The Course was good.
The in Labs were very
helpful in understanding the
material better and helped
in completing the assignments.