

Mar 4th, 11:15 AM - 11:45 AM

# Learning Computational Thinking Using Open-Source Hardware-based Programming

Rami Haddad

Georgia Southern University, rhaddad@georgiasouthern.edu

Salman Siddiqui

Georgia Southern University, ssiddiqui@georgiasouthern.edu

Follow this and additional works at: <https://digitalcommons.georgiasouthern.edu/stem>

---

## Recommended Citation

Haddad, Rami and Siddiqui, Salman, "Learning Computational Thinking Using Open-Source Hardware-based Programming" (2017). *Interdisciplinary STEM Teaching & Learning Conference*. 36.  
<https://digitalcommons.georgiasouthern.edu/stem/2017/2017/36>

This event is brought to you for free and open access by the Conferences & Events at Digital Commons@Georgia Southern. It has been accepted for inclusion in Interdisciplinary STEM Teaching & Learning Conference by an authorized administrator of Digital Commons@Georgia Southern. For more information, please contact [digitalcommons@georgiasouthern.edu](mailto:digitalcommons@georgiasouthern.edu).



**STEM**

SCIENCE TECHNOLOGY ENGINEERING & MATH

Interdisciplinary STEM Teaching & Learning Conference



**GEORGIA  
SOUTHERN  
UNIVERSITY**

Georgia's large-scale, small-feel research university

# Learning Computational Thinking Using Open-Source Hardware-based Programming

Rami Haddad, PhD, Salman Siddiqui, PhD

*Department of Electrical Engineering*

*Georgia Southern University*

# Outline

- Motivation
- What is Computational Thinking?
- Computational Thinking Misconceptions
- Problem Solving vs. Computational Thinking
- Proposed Hypothesis
- Study and Results
- Conclusion

# Motivation



We do not acquire technical skills simply from the use of technology any more than engineering skills evolve from using automobiles or aeronautical engineering skills from flying.



Georgia's large-scale, small-feel research university

Robert Tinker, Alvaro Galvis, and Andrew Zucker  
The Concord Consortium

# Motivation

Not everyone should be a programmer, but everyone should learn computational thinking to succeed as an engineer.



Georgia's large-scale, small-feel research university





"EVERYBODY IN THIS COUNTRY SHOULD  
LEARN HOW TO PROGRAM A COMPUTER,  
BECAUSE IT TEACHES YOU HOW TO THINK."

STEVE JOBS



Georgia's large-scale, small-feel research university

# What is Computational Thinking?

- Computational Thinking is a fundamental analytical skill that everyone can use to help solve problems, design systems, and understand human behavior.



## Modeling a problem involves:

- formulating the problem,
- defining its inputs and outputs,
- dividing it into its basic components using Computational Thinking modalities.

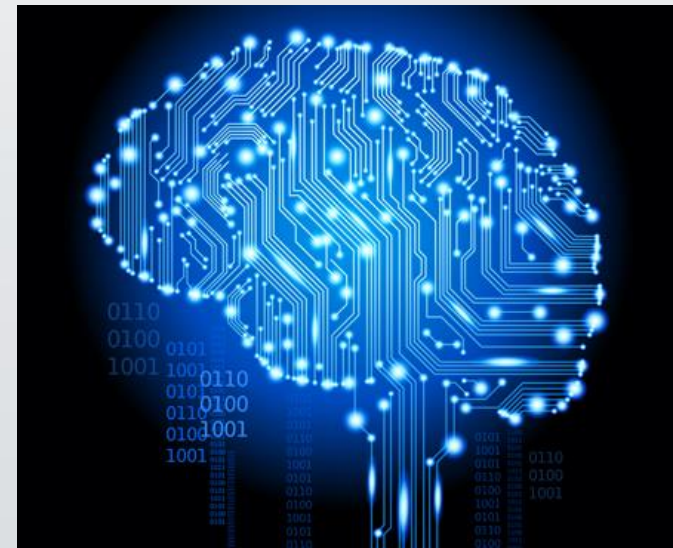


# Computational Thinking Misconception

- It's not just more technical details for using software.
- It's not thinking like a computer.
- It's not programming (necessarily).
- It doesn't always require a computer.

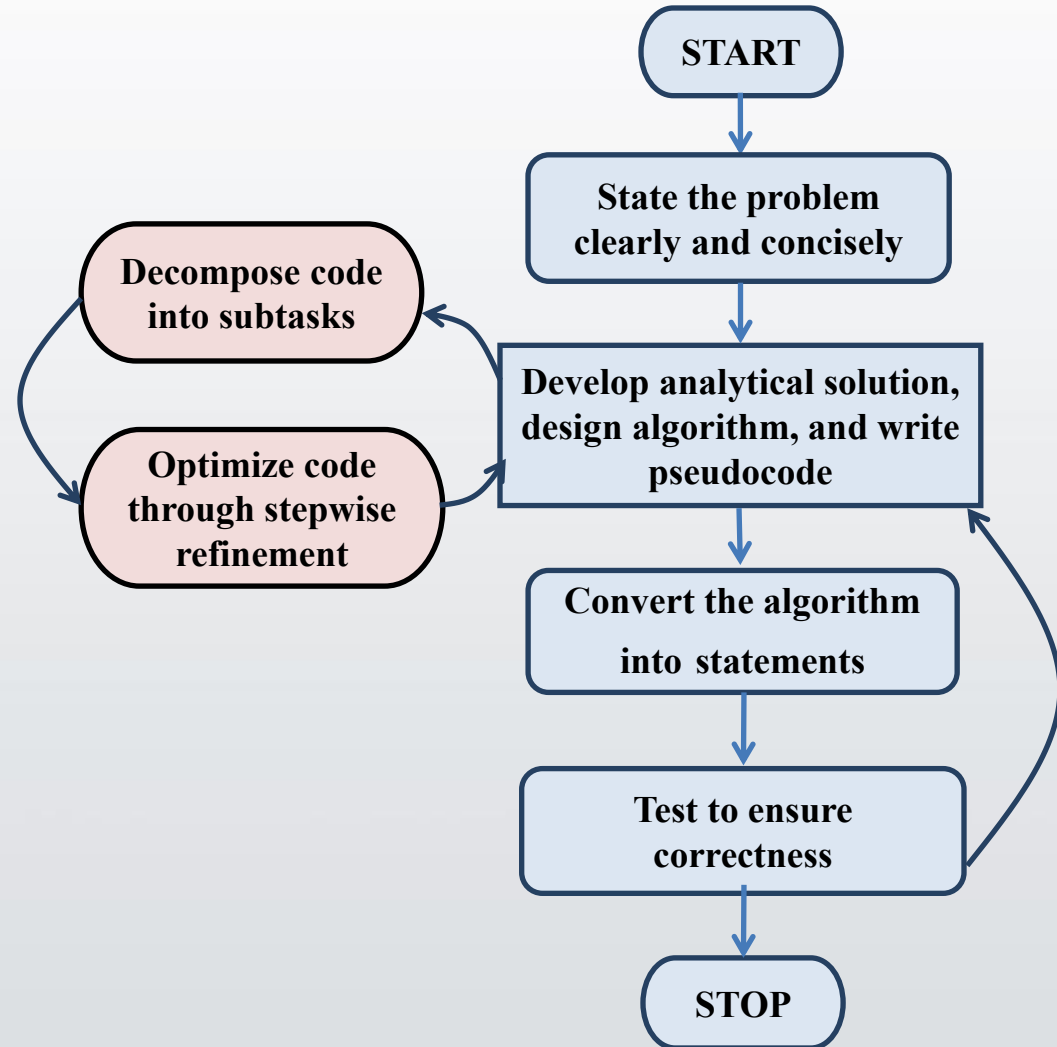
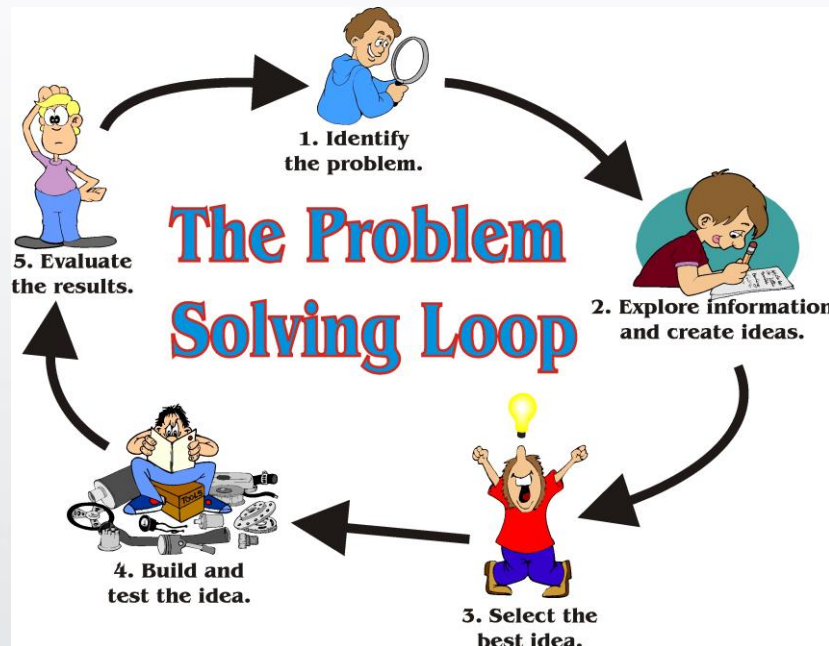


Georgia's large-scale, small-feel research university





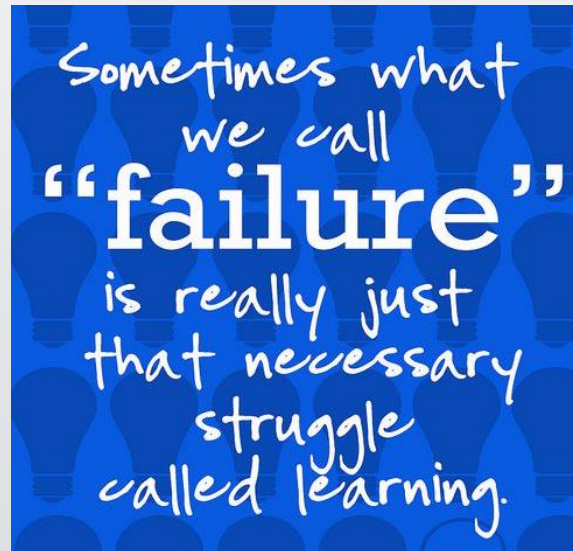
# Problem Solving vs. Computational Thinking



# Problem Solving vs. Computational Thinking

If I find 10,000 ways something won't work, I haven't failed. I am not discouraged, because every wrong attempt discarded is another step forward.

Thomas A. Edison  
1847 – 1931



# Proposed Hypothesis

If engineering is very dependent on creative problem solving and Computational Thinking is a structured process that facilitates this, then Computational Thinking should be introduced using an authentic learning process that involves engineering design process.



# Proposed Model Implementation

- We proposed using hardware programming of a microcontroller (Arduino) to engineer a practical system/product.
- The proposed authentic learning activity was implemented in an introductory freshman course “ENGR1731 - Computing for Engineers”.
- The students’ performance improvement was directly and indirectly assessed.
- Within the two months of the course, the traditional model was used, while the authentic learning model was used for the remainder of the course.

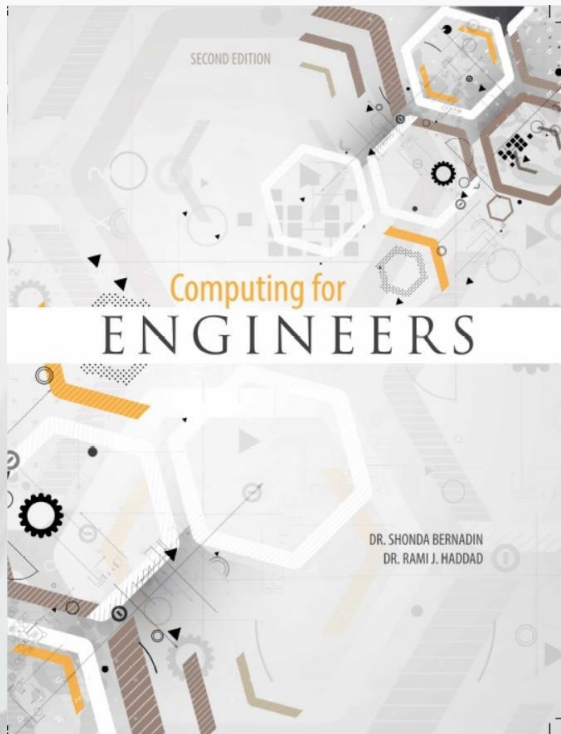


# Benefits of Proposed Model

1. Create an authentic learning environment where students can create real-world products.
2. Bridge the gap between the virtual/abstract nature of programming and the hands-on/applied nature of engineering students.
3. Address the basics of engineering principles and hands-on design at the freshman level.
4. Provides a unique opportunity for students to apply what they learned in a capstone-like project to help solidify their understanding of the topics being covered.
5. Develop the students' communication skills early-on by conducting presentations and writing reports.
6. Increase the students' overall performance and success in the course.



# Course Material & Notes



## Lectures

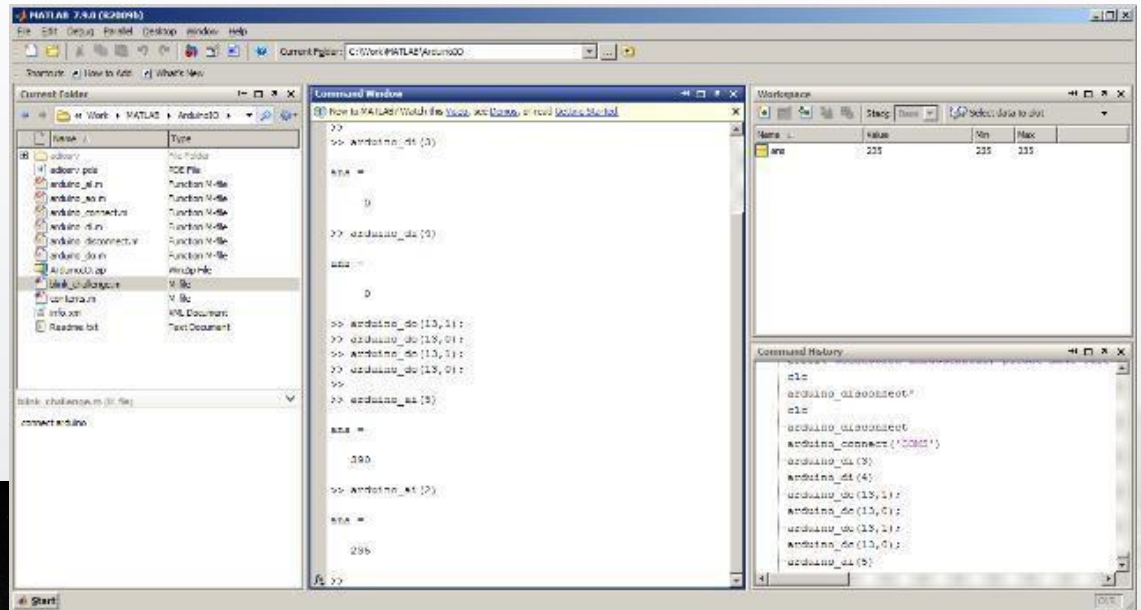
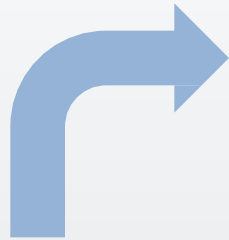
Lecture 1 - Engineering Essentials .....	1
Lecture 2 - Engineering Ethics .....	5
Lecture 3 - Engineering Communications .....	9
Lecture 4 - Introduction to Problem-Solving .....	15
Lecture 5 - Introduction to Matlab Environment .....	19
Lecture 6 - Matlab Basics (Arrays & Variables) .....	27
Lecture 7 - Array Operations and Built-in Functions .....	33
Lecture 8 - Introduction to Plotting and Debugging .....	41
Lecture 9 - Logical Data Types .....	49
Lecture 10 - Conditional Statements .....	55
Lecture 11 - More on Plotting in Matlab .....	63
Lecture 12 - Structured Programming .....	71
Lecture 13 - Data Modeling using Linear Regression .....	77
Lecture 14 - Introduction to Matlab Functions .....	83
Lecture 15 - Other types of Matlab Functions .....	87
Lecture 16 - Numerical Techniques (Random Number Generation) .....	91
Lecture 17 - Cell Arrays .....	97
Lecture 18 - Structure Arrays .....	103
Lecture 19 - Structure Array Examples .....	107
Lecture 20 - Input/Output Functions .....	111
Lecture 21 - Image Processing .....	115



Georgia's large-scale, small-feel research university

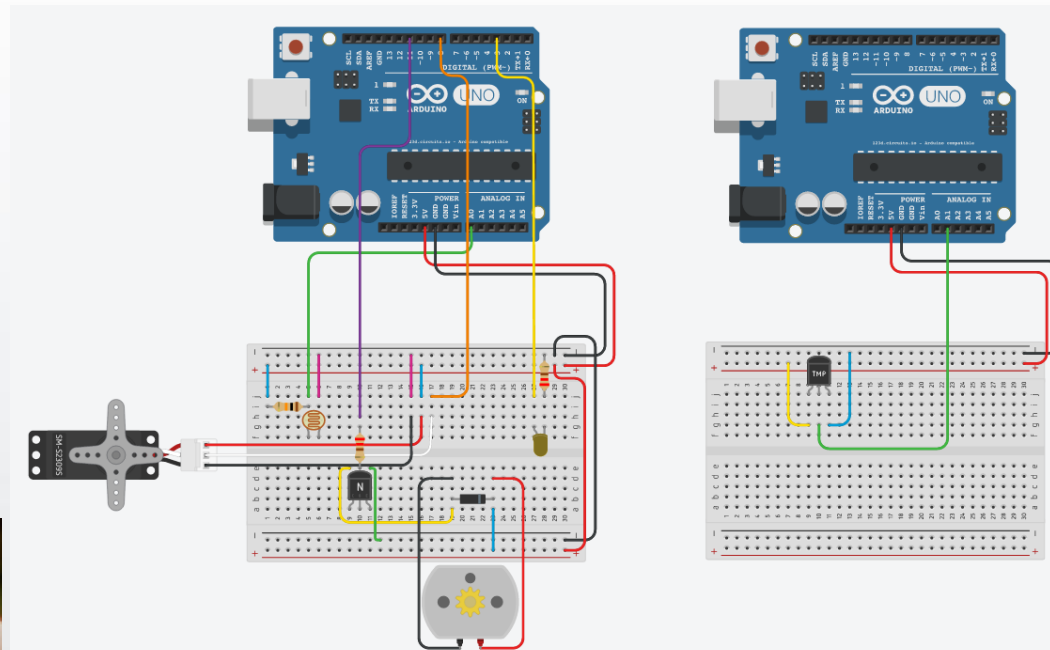
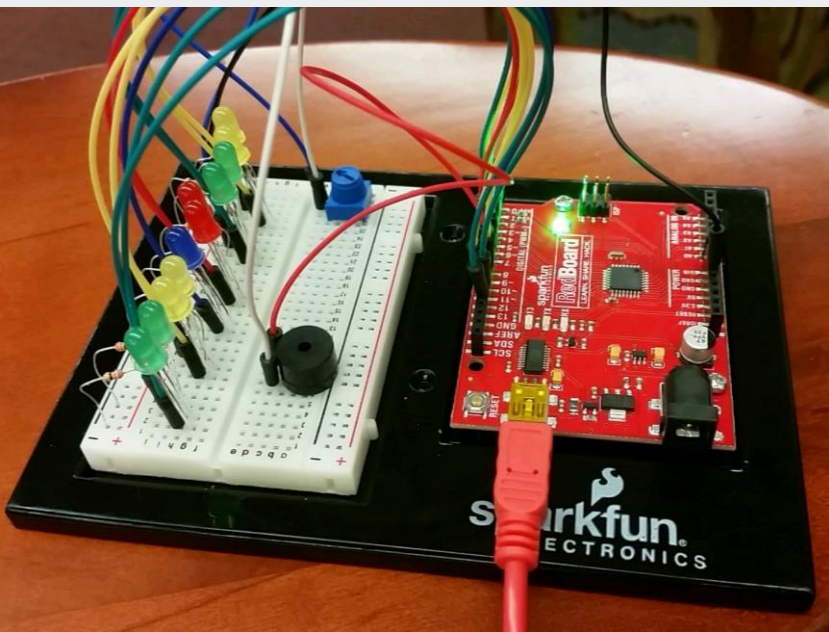
# Hardware Programming using MATLAB

Arduino board answers to Matlab via USB



MATLAB sends a command or request to the Arduino board via USB

# ENGR 1731 Students' System Designs



Georgia's large-scale, small-feel research university



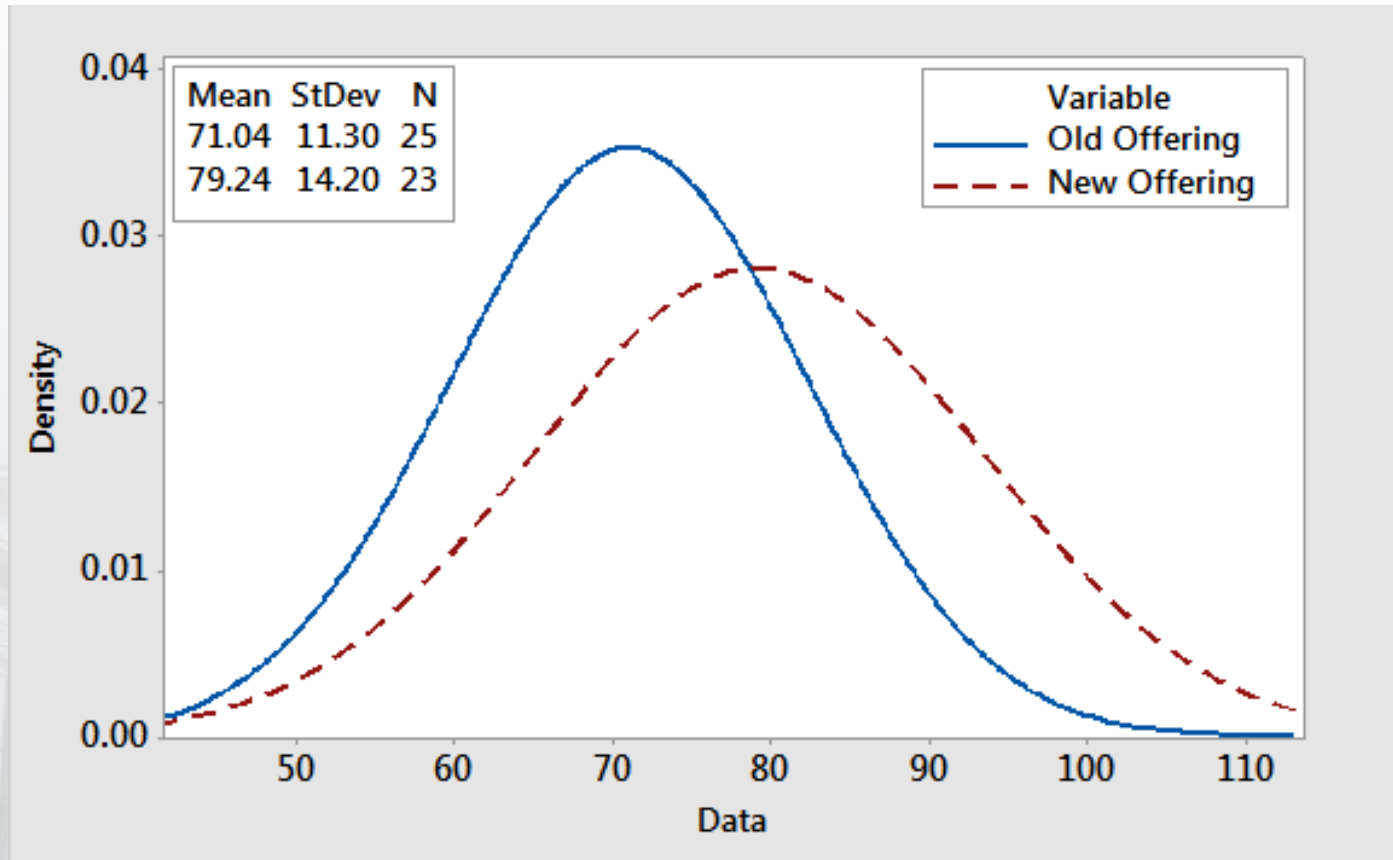
# Study Details

## To test our hypothesis,

- a quantitative analysis was conducted to compare 2 offerings of this Computing for Engineers course (with and without Hardware programming).



# Normal Fit of Data



**The new offering with hardware programming resulted in a higher final exam mean.**

# Statistical Analysis Results

## General Linear Model: Final Exam Grade versus Offering

Method

Factor Information

Factor	Type	Levels	Values
Offering	Fixed	2	Proposed, Traditional

Analysis of Variance

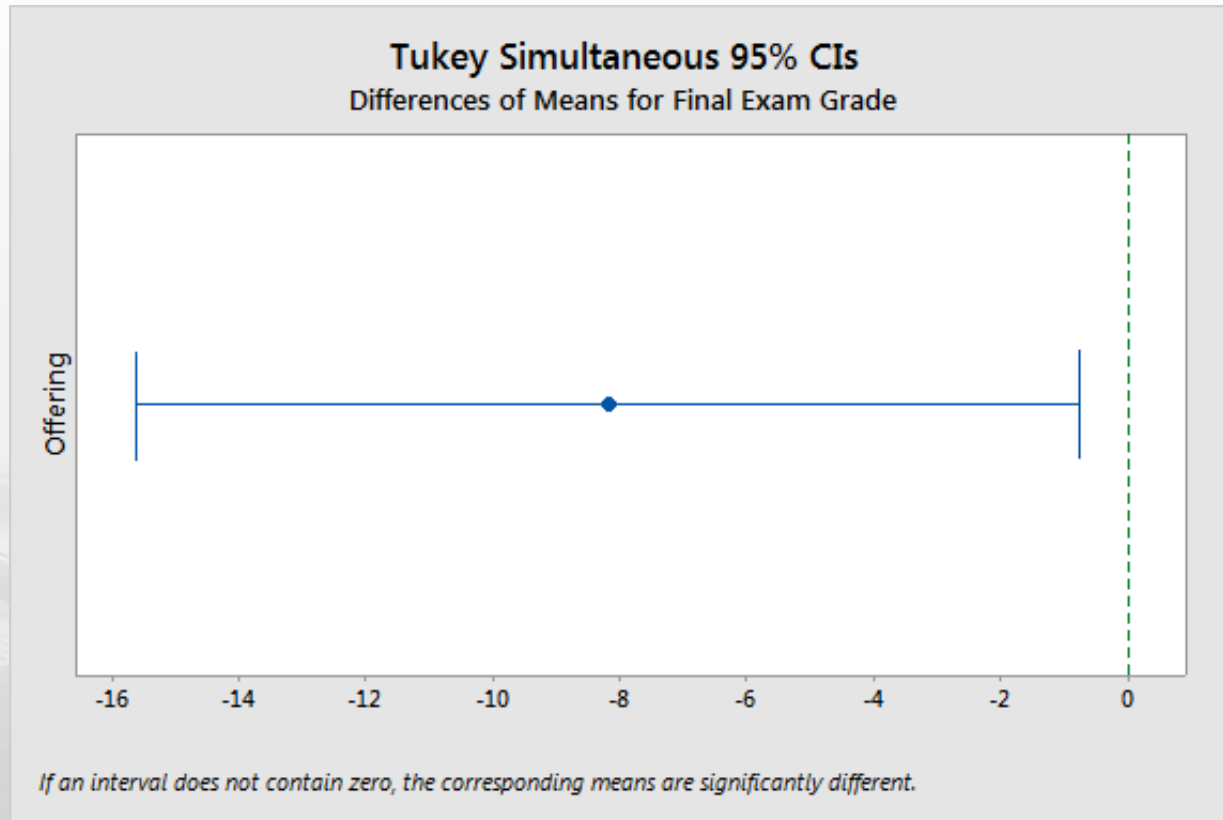
Source	DF	Adj SS	Adj MS	F-Value	P-Value
Offering	1	805.3	805.3	4.94	0.031
Error	46	7499.3	163.0		
Total	47	8304.6			



Georgia's large-scale, small-feel research university

We achieved statistical significance with a confidence level of **96.9%**

# Pairwise Comparisons - Course Grades



**Engineering students' academic achievement represented by their Final grade highlight the effectiveness of hardware programming**

# Qualitative Assessment

*"I liked how useful the material is and how many helpful resources were available to learn the material."*

*"I liked learning coding and interfacing with hardware like Arduino. Allows me to get ahead and learn more things."*

*"The Arduino project was fun and the labs are good too."*

*"The course itself is easy to be engaged in because it is fun to learn MATLAB and its uses (applications)"*

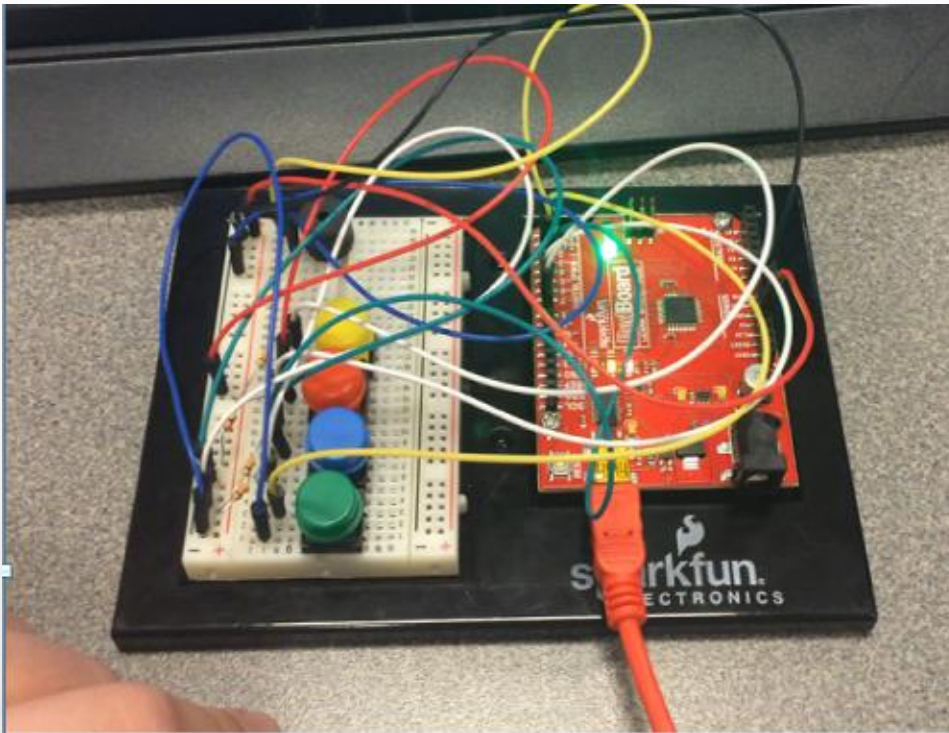
*"We got to explore practical applications of what we have learned so far with sensors and Arduino board."*

*"The work was challenging but enjoyable."*

*"I liked being able to apply knowledge in real world."*



# Demonstration of Sample Designs



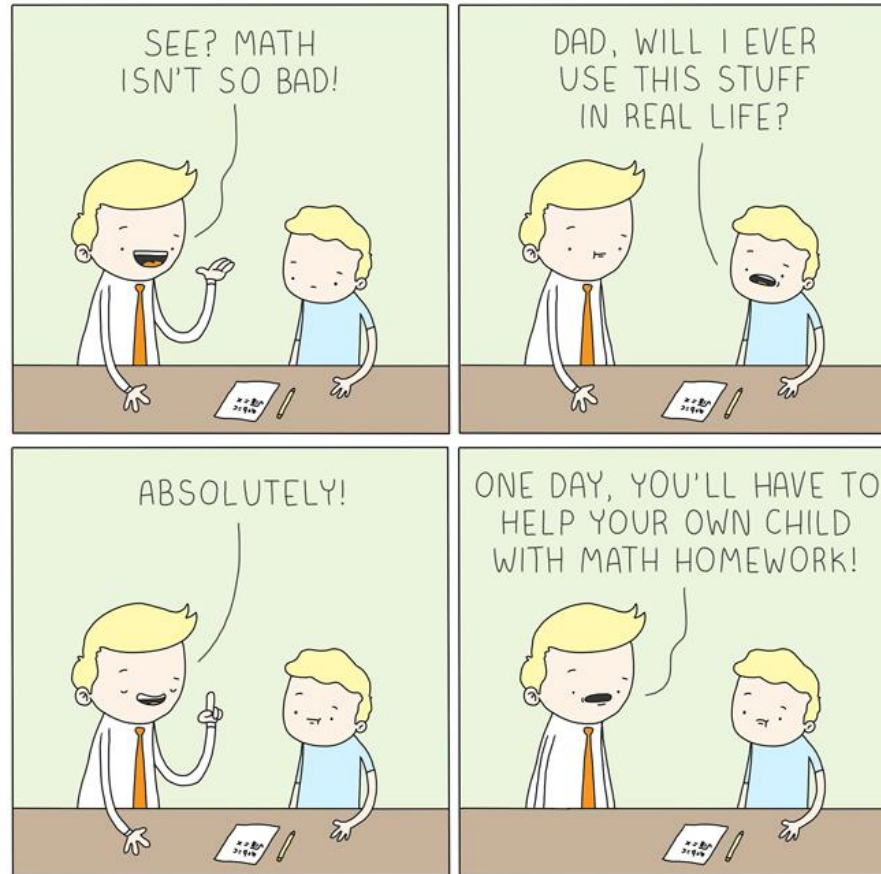
Georgia's large-scale, small-feel research university

# Conclusion

- A Computational Thinking authentic learning experience using hardware programming was presented.
- Hardware programming helped students better grasp Computational Thinking in a freshman course.
- We concluded that students' performance is statistical difference with over 96.9% confidence level when comparing 2 offering with/without hardware programming.



# Questions?



THEGENTLEMANSARMCHAIR.COM

## Contacts:

**Rami Haddad** ([rhaddad@georgiasouthern.edu](mailto:rhaddad@georgiasouthern.edu)),

**Salman Siddiqui** ([ssiddiqui@georgiasouthern.edu](mailto:ssiddiqui@georgiasouthern.edu))



Georgia's large-scale, small-feel research university