2023 Annual Conference & Exposition
Baltimore Convention Center, MD | June 25 - 28, 2023
The Harbor of Engineering
Education for 130 Years
Paper ID #37281

Comparing First-Year Student Programming Confidence Perceptions Between Different Hands-On Projects

Dr. James E. Lewis, University of Louisville

James E. Lewis, Ph.D. is an Associate Professor in the Department of Engineering Fundamentals in the J. B. Speed School of Engineering at the University of Louisville. is research interests include parallel and distributed computer systems, cryptography, engineering education, undergraduate retention and technology used in the classroom.

Mr. Nicholas Hawkins, University of Louisville

Nick Hawkins is an Assistant Professor in the Engineering Fundamentals Department at the University of Louisville. He received his B.S. (2016), M.Eng. (2017), and Ph.D. (2020) in Electrical and Computer Engineering at the University of Louisville. His res

Dr. Brian Scott Robinson, University of Louisville

Brian Robinson is an Associate Professor with the Department of Engineering Fundamentals at the University of Louisville. His primary research focus is in Engineering Education, with highest interest in first-year (and beyond) engineering retention & the effects of value-expectancy theory on student persistence.

Comparing First-Year Student Programming Confidence Perceptions Between Different Hands-On Projects

Abstract

Each first-year student attending the J. B. Speed School of Engineering (SSoE) at the University of Louisville (UofL), regardless of declared major, must complete a two-course sequence of introductory engineering courses. These courses, *Engineering Methods, Tools, & Practice I* (ENGR 110) and *Engineering Methods, Tools, & Practice II* (ENGR 111), introduce the fundamental tenets of the engineering profession. The first course in the sequence, ENGR 110, focuses on introducing a variety of fundamental engineering skills. The second course, ENGR 111, is a hands-on, project-based course housed in a 15,000 square foot makerspace that has students integrate and apply the skills learned in ENGR 110. One of the many skills taught in this sequence is basic programming.

The programming instruction presented in ENGR 111 is an extension of the programming skills learned in ENGR 110. However, ENGR 110 teaches programming basics in Python, whereas the ENGR 111 instruction utilizes Arduino microcontrollers for its programming curriculum. The programming instruction in ENGR 111 also forgoes standalone programming assignments for scaffolded modules that prepare students for an end-of-semester Cornerstone Project. Accordingly, students gain exposure to varying programming languages, and a wide introduction to software design concepts that help prepare them for the remainder of their academic and professional careers.

In this paper, two semesters of ENGR 111 with two different Cornerstone Projects will be compared. Project 1 took place during the spring of 2022 and was comprised of a windmill power generation system. Students constructed this windmill and used Arduino programming to interpret sensor data and calculate system performance. Project 2 took place during the 2022 summer semester and was comprised of a water filtration system. In this project, students utilized the Arduino to both observe system information and control its behavior.

At the end of each of these semesters, students took a survey in which they provided their perceptions of the programming instruction they received, in addition to expressing their confidence in programming. Results of these questions from Spring 2022 (Project 1) and Summer 2022 (Project 2) are compared in this paper.

Introduction

The J. B. Speed School of Engineering (SSoE) at the University of Louisville (UofL) has a required first-year two-course sequence that all incoming engineers, regardless of their engineering major, are required to complete. The courses in this sequence are *Engineering Methods, Tools, & Practice I* (ENGR 110), and *Engineering Methods, Tools, & Practice II* (ENGR 111). The first course in the sequence is ENGR 110 and this course is an introduction of the profession and fundamentals of engineering to first-year student. The next course in the sequence is ENGR 111. ENGR 111 is normally a makerspace-based course primarily focused on application and integration of the fundamentals learned in ENGR 110. Programming and circuitry have been identified as fundamental skills for all engineering majors. ENGR 111 exposes students to programming and circuitry to allow for a basic understanding regardless of engineering major. ENGR 110 is the first introduction to programming fundamentals to the students using the programming language Python. ENGR 111 follows with students applying their developed knowledge of programming, now integrated with circuitry.

During the two-course sequence development, the decision was made to use more than one type of programming environment. One motivator for this decision was to help the students understand that core programming concepts are the same regardless of the programming language used; i.e. a *for* loop logically works the same in all environments even though the syntax may differ. Arduinos and their associated Arduino Programming Language (APL) are the programming environment used in ENGR 111. ENGR 111 chose the Arduino Uno as the microcontroller for the course since it is an excellent tool for teaching basic circuitry, programming, and the interaction between the two.

The ENGR 111 course finishes with a team-based Cornerstone project that all students demonstrate and present near the end of the semester. The ENGR 111 course instruction, activities, and deliverables are designed to progress the students towards completion of their Cornerstone project. ENGR 111 is taught in a 15,000 ft^2 makerspace. There are individual classrooms as well as workstation areas in the makerspace. The Cornerstone projects are designed to be long term projects spanning multiple lab sessions, which is advantageous because it simulates how engineering functions in industry [1]. This involves hands-on learning and utilizes the presence of multiple instructors to assist the students [2] [3].

Scaffolded lessons are used in ENGR 111 to introduce the students to circuitry, programming on the Arduino, and interfacing between an Arduino and circuits. Each of these lessons begin with basic stand-alone circuits using breadboard, basic components, and wires. Arduino programming is next. Basic programming concepts and how to interact with an Arduino are the focus of these programming lessons. These programming lessons begin with a discussion regarding the similarities and differences of the APL to Python. Upon completion of the “Programming lessons”, there is a series of activities designed to help the students create circuit(s) and program(s) that interact with each other.

The programming and circuitry scaffolded modules prepare students for an end-of-semester Cornerstone Project. ENGR 111 currently has two different Cornerstone Projects. The Cornerstone Project is determined by the semester and year that the course is taken. The first Cornerstone Project (Project 1) is comprised of a windmill power generation system. Project 1 has students constructing a windmill and using Arduino programming to interpret sensor data and calculate system performance. The second Cornerstone Project (Project 2) is comprised of a water filtration system. Project 2 has students utilizing an Arduino to both observe system information and control the system behavior.

Project 1: Windmill System

As mentioned, Project 1 involves the construction and design of a windmill system. Project 1 requires the integration of a windmill, student-built AC motors, DC motors, circuitry, data acquisition, manipulation of the acquired data, and the display of the data results. Figure 1 shows a couple of images of ENGR 111 students "in action", and examples of respective windmill systems can also be seen.

Figure 1: Sample images of ENGR 111 students finalizing respective windmill (Cornerstone) systems for demonstration.

Project 1 involves measuring the rotational speed of the bench-scale windmill as it is powered by a fan to simulate wind. To obtain this measurement, students are given a proximity sensor and must use the Arduino to calculate the timing of the windmill's blades passing through the view of the sensor (thus simulating a basic tachometer). The windmill is mechanically coupled with a small DC motor which outputs to a basic electrical load. Students must utilize these measurements for a variety of calculations that determine the performance of the windmill system, including blade and motor efficiencies, and electrical power output. Students can even attempt to improve their system performance through adjustments in blade pitch and output load sizing.

Project 2: Water Filtration System

Project 2 involves the creation, control, and use of a water filtration system. Project 2 requires the integration of a water flow system, student-built filtration canisters, circuitry, data acquisition, using the acquired data to manipulate a pump and valve to control the water flow, and the display of the data results. The figure below shows a completed water filtration system Cornerstone project.

Figure 2: Visual representation of an ENGR 111 water filtration (Cornerstone) system, and a student adding water to the system.

Students working on Project 2 incorporate 2 sensors: a turbidity sensor for measuring water cleanliness, and an ultrasonic distance sensor for determining the water level in one of the tanks. The ultrasonic sensor is an integral component of a fully constructed system, as that water level is used to inform a variety of system states, such as controlling the downstream pump to only operate when there is enough water to be pumped and calculating tank volume and flow rate.

Arduino and the Cornerstone Projects

The microcontroller chosen for use with each of the aforementioned Cornerstone Projects was an Arduino Uno. The Arduino Uno microcontroller is illustrated below in Figure 3 [4].

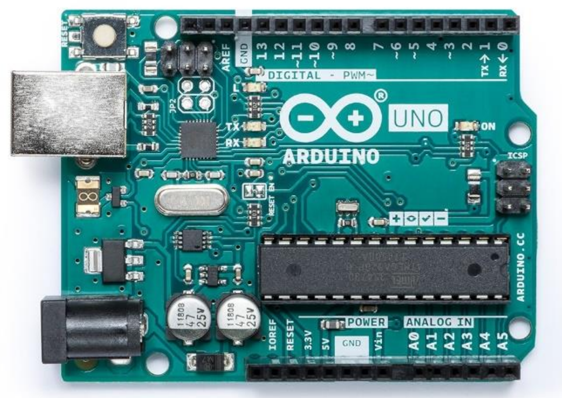

Figure 3: Visual representation of the Arduino Uno microcontroller used in ENGR 111.

The Arduino Uno was chosen because it is excellent for teaching basic circuitry and programming. The Arduino Uno also has easily accessible digital and analog input/output ports and uses a variant of the C programming language. This makes it simple for students to incorporate external measurements and utilize those values in their calculations, which is necessary for both projects used in this course.

Student Programming Surveys

At the end of each of these semesters, students took a survey which included questions related to their perceptions in programming confidence. Two quantitative queries, in which students self-reported their confidence in programming are stated below; both survey questions consisted of a 5-point Likert scale (Not Confident at All, Slightly Confident, Somewhat Confident, Very Confident, Extremely Confident):

- Q17: *Rate your current confidence level in programming with ARDUINO.*
- Q18: *Rate your current confidence level programming with ANY language.*

Results of these questions from Spring 2022 (Project 1) and Summer 2022 (Project 2) are compared in this paper. Project 1 (Windmill) had 39 students participate in the survey and Project 2 (Water Filtration) had 23 students participate.

Student responses for Q17 related to confidence in Arduino programming are shown in Figure 4 below.

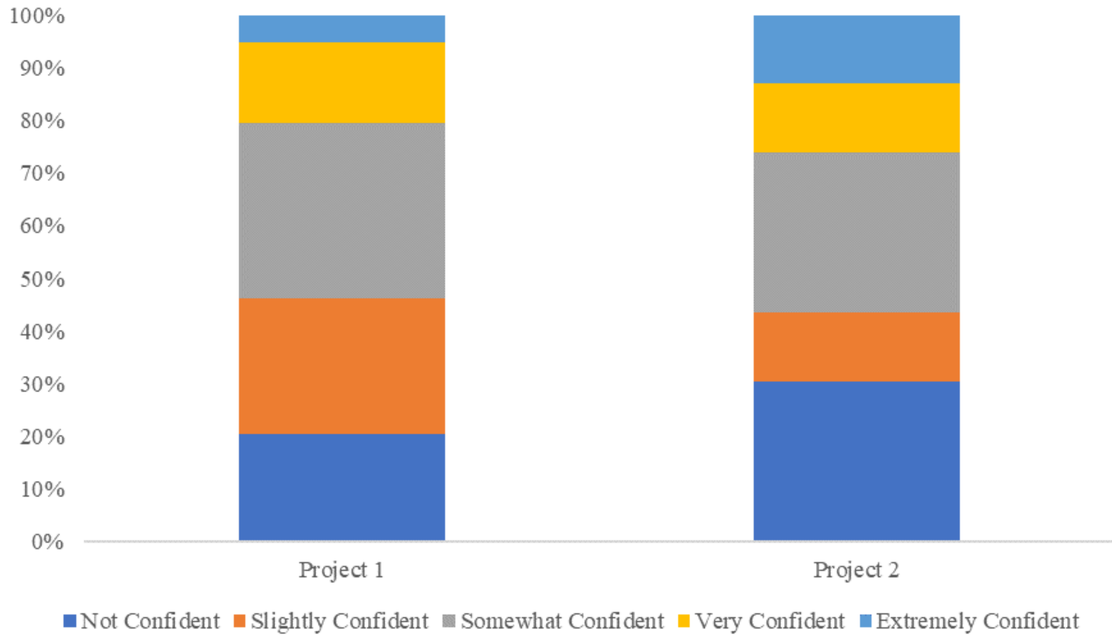

Figure 4: Student responses to Q17 regarding confidence level in programming an Arduino.

Results indicate that Project 2 is more polarizing for student confidence in Arduino programming, with a higher percentage of students choosing the lowest or highest levels than in Project 1. However, Project 2 has a very slightly higher percentage of students responding with at least "Somewhat Confident" (56.4%) compared to Project 1 (53.8%), though both being above 50% indicates both projects had a positive effect on their confidence overall.

Student responses for Q18 related to confidence in programming with any language are shown in Figure 5 below.

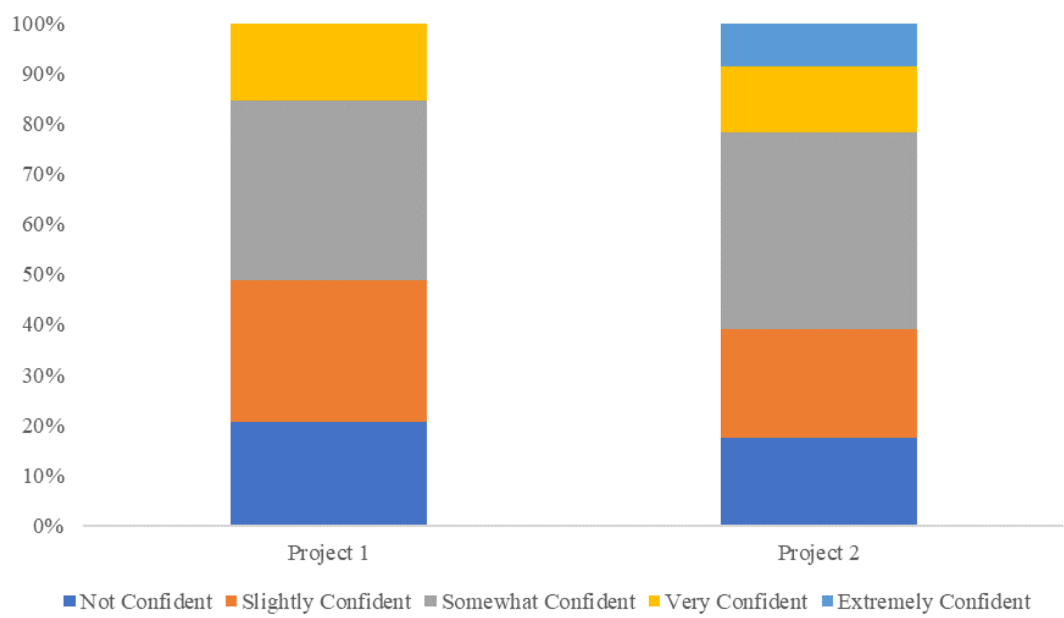

Figure 5: Student responses to Q18 regarding confidence level in programming with any language.

Notably, Project 1 has no students responding with "Extremely Confident", whereas 2 students (8.7%) in Project 2 did. The middle category "Somewhat Confident" dominates both sets of responses, with over 35% of students responding this way for both projects.

Additionally, student confidence across both questions were compared for each project. To accomplish this, each response has been given a numerical score as shown in Table 1. Then, the difference between a student's Q17 and Q18 score was calculated, and a summary of these differences is compiled in Table 2.

Table 1: Numerical scoring of each possible survey response.

Numerical Score	Survey Response
1	Not Confident at All
2	Slightly Confident
3	Somewhat Confident
4	Very Confident
5	Extremely Confident

Table 2: Comparison of student confidence levels between questions for both course projects.

Value	Project 1	Project 2
Average Score	-0.128	0.087
Standard Deviation	1.128	0.900
Number of Students More Confident with Any Language (total % of cohort)	10 (26%)	6 (26%)
Number of Students with Equal Confidence (total % of cohort)	17 (43%)	11 (48%)
Number of Students More Confident with Arduino (total % of cohort)	12 (31%)	6 (26%)

A negative average score for Project 1 indicates that students were overall more confident in Arduino programming than programming at large (Q18 – Q17). It is worth noting that Project 1 includes one student response with a -4 score ("Extremely Confident" in Arduino but "Not Confident at All" in any language), which could skew the results very slightly (as noted by the higher standard deviation). The distribution of confidence in programming languages between respective projects are very similar; for example, looking at row 3 in Table 2, 26% of both the Project 1 and Project 2 cohort(s) expressed higher confidence in any programming language versus exclusively Arduino language. A little under half of students expressed equivalent confidence in Arduino versus any language for both cohorts (43% of the Project 1 cohort and 48% of the Project 2 cohort), and close to a quarter of each cohort (31% for Project 1 and 26% for Project 2) expressed a higher level of confidence in Arduino programming versus programming in any language.

Discussion and Conclusions

One pertinent difference between the two projects within the scope of programming is the way students interact with their system. Both projects (Project 1 and Project 2) include some form of *measurement* and *calculation*. Each project involved a sensor collecting a measured value, which is sent to the Arduino, where student code can mathematically incorporate that data into various formulas. What separates these two projects is that Project 2 also included *control*. The code that students write in Project 2 exhibits a physical response to the system in the form of pumps activating and valves changing states. This difference in programming could lead to differences in the students' debugging "frustration" (a sizeable portion of programming microcontrollers), students found Project 1 more applicable to general programming concepts.

Initially, it may seem nonsensical that just over a quarter of students in both cohorts expressed a higher level of confidence in programming in any language versus Arduino only. In other words, one may assume that a certain level of confidence in "any" language would intuitively include at least the same level of confidence in the Arduino language, which certainly also falls into the "any" language category. This discrepancy is possibly due to students linking the "any" programming language category *exclusively* to the Python language; since these are the two languages introduced in the ENGR 110 (Python) and ENGR 111 (Arduino) courses, which for many first-year engineering students at SSoE represent their very first exposure to the subject of programming. Another key difference between the programming experience in ENGR 110 versus ENGR 111 is that, unlike ENGR 110, the programming experience in ENGR 111 (using Arduino) involves additional integration with circuitry, adding a new level of complexity to associated practice in programming and thusly resulting in a lower perceived level of programming confidence in Arduino.

It is interesting, and somewhat surprising, that the highest number of students, regardless of project type, expressed an equal level of confidence in all languages. It was expected that the largest number of students would select higher confidence in Arduino, since that is the primary platform utilized in ENGR 111 (during which the survey was employed). Yet a higher percentage choosing equal level of confidence is encouraging to course administrators that pedagogical efforts, across the entire ENGR 110/111 sequence, in building student confidence in programming across all languages are agreeable. Ultimately, course administrators are satisfied that the differences in programming project type seems to have little to no negative impact on the distribution of student confidence levels in programming. The current (2023) spring semester of ENGR 111 is employing Project 2 (Project 1 data has already been collected during another larger, spring iteration of ENGR 111), and the same survey(s) discussed in this paper will be presented to this cohort. Accordingly, another round of assessment identical to that conducted and discussed with the smaller cohorts in this paper will be conducted and comparatively assessed to confirm existing and/or reveal new trends using a more robust sample size.

References

[1] V. G. Agelidis, "The future of power electronics/power engineering education: Challenges and opportunities," IEEE Power Electron. Educ. Work. 2005, vol. 2005, pp. 1–8, 2005, doi: 10.1109/PEEW.2005.1567584.

[2] L. Guo, “Design Projects in a Programmable Logic Controller (PLC) Course in Electrical Engineering Technology,” Technol. Interface J., vol. 10, no. 1, pp. 1523–9926, 2009.
[3] R. V. Krivickas and J. Krivickas, “Laboratory instruction in engineering education,” Glob. J. Eng. Educ., vol. 11, no. 2, pp. 191–196, 2007.
[4] “Arduino.” [Online]. Available: https://store-usa.arduino.cc/products/arduino-uno-rev3 .