Engineering Educators Bringing the World Together
2025 ASEE Annual Conference & Exposition
Palais des congrès de Montréal, Montréal, QC • June 22–25, 2025
Paper ID #47613

# Bridging the Gap: A Project-Based Approach to Robotics Programming for First-Year Robotics Engineering Students

**Mahdi Agheli, Worcester Polytechnic Institute**

Mahdi Agheli is an Associate Teaching Professor and the chair of the Curriculum Committee with the Robotics Engineering Department at Worcester Polytechnic Institute.

**Gregory C. Lewin, Worcester Polytechnic Institute**

Dr. Lewin is an Associate Teaching Professor of Robotics Engineering at WPI.

**Andre Rosendo, Worcester Polytechnic Institute**

Andre Rosendo is an Assistant Teaching Professor with the Robotics Engineering Department at Worcester Polytechnic Institute.

**Vincent A Aloi, Worcester Polytechnic Institute**

Vincent Aloi is an Assistant Teaching Professor with the Worcester Polytechnic Institute. Their Bachelor of Science (2016), Master of Science (2021), and doctorate (2022) degrees were awarded from the University of Tennessee, Knoxville.

# Bridging the Gap: A Project-Based Approach to Robotics Programming for First-Year Robotics Engineering Students

## Abstract

In response to the evolving field of Robotics Engineering and the increasing demand for specialized programming skills within the discipline, the Robotics Engineering (RBE) department at Worcester Polytechnic Institute has introduced a tailored, first-year programming course designed specifically to address the needs of RBE students. This course was developed based on data collected through surveys from both students and faculty, which revealed a significant gap in programming proficiency impacting student performance in advanced RBE courses. Previously, RBE students were enrolled in programming courses offered by the Computer Science (CS) department; however, these courses often failed to meet the unique demands of robotics engineering, resulting in challenges for up to 30% of the students. The newly introduced course mitigates this issue through a project-based learning approach, where programming concepts are directly applied to real-world robotics projects. This paper presents a detailed analysis of the implementation process, course structure, and the pedagogical foundation of this newly developed programming course. We offer a comprehensive evaluation of the course's effectiveness, using both qualitative and quantitative assessments such as student feedback, performance metrics, and cohort comparisons. Our evaluation focuses particularly on the course's impact on students' readiness for subsequent robotics courses, demonstrating a marked improvement in their ability to apply programming skills in robotics-specific contexts. Additionally, we explore the broader implications of this course for the RBE curriculum and how it may serve as a model for other robotics engineering programs seeking to enhance student preparedness for advanced technical challenges. By demonstrating the impact of this educational approach, we aim to inspire other institutions to adopt similar strategies to address the rapidly evolving demands of the field.

## Introduction

The Robotics Engineering (RBE) program at Worcester Polytechnic Institute (WPI) emphasizes hands-on learning through a project-based, programming-intensive curriculum. The program is meticulously designed to not only impart theoretical knowledge but also to ensure experiential learning. Central to this curriculum are core courses such as Introduction to Robotics (RBE 1001), Robot Actuation (RBE 2001), Robot Sensing (RBE 2002), Robot Manipulation (RBE 3001), and Robot Navigation (RBE 3002). Each of these courses, characterized by their project-based and lab-heavy nature, demands a robust foundation in programming. This aspect of the curriculum is not merely about teaching programming as a skill; it is integrally woven into the fabric of the learning process, enabling students to apply these skills in real-world scenarios, thereby preparing them for the complex challenges of the robotics field.

Understanding the significance of programming in robotics, the RBE curriculum strategically incorporates 1 unit, equivalent to three courses, in computer science, ensuring that students are well-equipped with the necessary computational tools and techniques. These foundational courses set the stage for more advanced exploration and application in subsequent RBE courses. However, unlike many traditional educational models where prerequisites are a strict gatekeeper to course entry, the WPI's approach adopts a more flexible model. Here, while prerequisites are not rigidly enforced, a set of recommended backgrounds is provided for each course. Although this unique approach fosters a diverse and inclusive learning environment, it does present its own set of challenges. One such challenge is the varying levels of programming expertise that students bring to the RBE core classes. Given the absence of strict prerequisites, students' proficiency in programming can range widely, impacting their readiness to tackle the course material effectively.

In this paper, we examine how the addition of a tailored first-year programming course, designed specifically to address the needs of RBE students, enhances their readiness for advanced robotics courses and bridges gaps in their programming background.

## Background and Problem Context

To navigate the diversity in student programming backgrounds, we employ CATME (Comprehensive Assessment of Team Member Effectiveness) surveys [1] as a tool to gauge the students' existing knowledge and skills in programming and, accordingly, tailor the learning experience to meet the diverse needs of our students, ensuring that all participants can achieve the course's learning objectives. Prior assessments in Fall 2021 and Spring 2022 revealed a significant and systemic gap in programming readiness among RBE students. Surveys conducted across core courses, such as Robot Sensing (RBE 2002), Robot Manipulation (RBE 3001), and Robot Navigation (RBE 3002), revealed that up to 30% of students lacked adequate preparation in programming. This deficit significantly hindered their ability to engage effectively with the advanced robotics concepts and challenges presented in these courses. Additionally, the surveys identified a widespread lack of proficiency in programming languages critical to robotics applications, such as Python and C++. This gap underscored a systemic issue in the foundational programming education provided to RBE students.

Further analysis indicated that conventional programming courses offered by the CS department failed to address the unique and practical demands of robotics engineering. These courses typically emphasized general computing principles, which, while valuable, did not adequately prepare students for the specific challenges of applying programming within a robotics context. The resulting disconnect between the curriculum's focus and the practical needs of robotics students created an urgent need for a targeted intervention.

To address this, the RBE department conducted a comprehensive investigation into potential solutions. Figure 1 shows our approach pipeline. The process included a thorough review of peer institutions' curricula to identify best practices, detailed consultations with faculty members to understand pedagogical gaps, and extensive surveys of students to gather insights on their learning experiences and challenges. These efforts culminated in proposing four options as potential solutions, finally narrowing down into the development of a dedicated first-year programming course tailored specifically for RBE students, which was offered in Fall 2023 for the first time [3], followed by two offerings in Fall 2024. After, three offerings throughout two

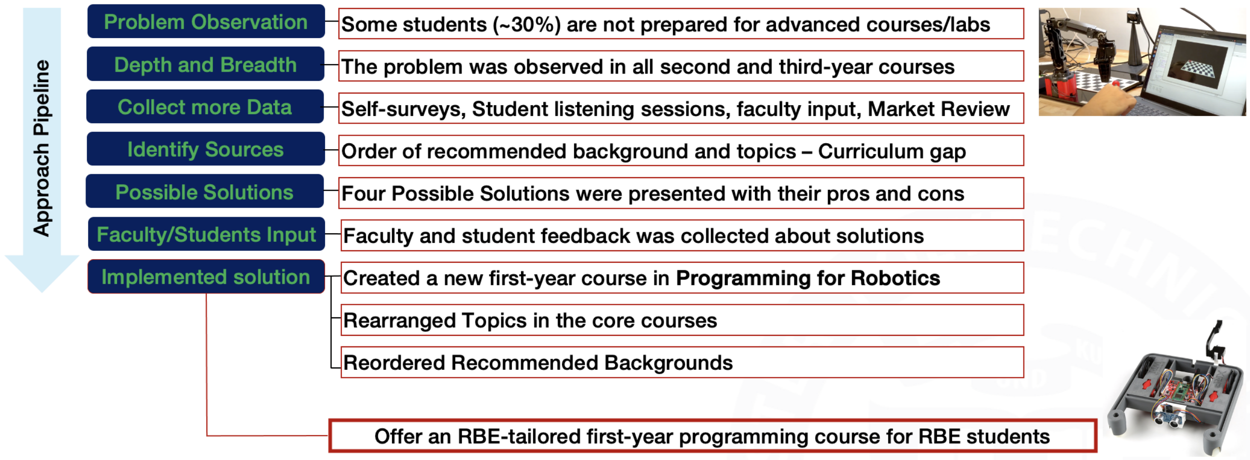


Figure 1: Our approach pipeline resulting in offering a new first-year RBE-tailored programming course, “Programming for Robotics”.

years, in this paper, we study the effectiveness of this course in the students preparation for advanced robotic courses. The authors will investigate the broader impact of this solution as students have progressed to their higher-level courses.

## Course Development and Implementation

The ”Programming for Robotics” course was meticulously designed to address the unique challenges faced by first-year RBE students, aiming to bridge the programming readiness gap through a project-based learning approach. By integrating programming instruction directly with robotics-specific applications, the course provided hands-on experiences that enhanced students’ technical skills and contextual understanding. This structure not only prepared students for advanced coursework but also equipped them with the practical competencies required for industry challenges, ensuring a seamless progression through their academic and professional journeys. Built on a well-defined syllabus and weekly schedule, the course incrementally enhanced programming skills while emphasizing robotics-specific applications. Grounded in the principles of Constructivism and Experiential Learning [2], the curriculum prioritized active participation, contextual learning, and hands-on engagement. Fundamental programming skills were taught with immediate applicability in robotics-specific contexts, effectively bridging the gap between theoretical knowledge and practical application. Replacing the general CS introductory programming course for RBE students, this tailored curriculum embedded robotics-specific content, providing a targeted foundation essential for subsequent RBE courses and advanced robotics challenges.

### *Course Description and Learning Outcomes*

The course introduces students to the fundamental principles of programming within the context of robotics. Topics include data structures, control flow, modularization, state machines, and event-based input/output. Emphasis is placed on implementing, testing, and debugging code for microcontrollers and robotic systems. Special focus is given to writing efficient and reusable code, providing students with an appropriate programming foundation for subsequent RBE courses. The learning outcomes are defined to be clear and comprehensive as shown in Table 1.

Table 1: Learning Outcomes of the *Programming for Robotics* Course

| # | **Learning Outcome** |
|---|---|
| 1 | Use fundamental programming concepts, including variables, control structures, functions, and loops, to solve robotic problems. |
| 2 | Write efficient, structured, and task-driven code. |
| 3 | Follow the best practices in coding, e.g., commenting, modularity. |
| 4 | Develop algorithms for fundamental problems using existing and custom libraries. |
| 5 | Analyze, debug, and adapt code written for robotic systems. |
| 6 | Describe the components of a robotic system and their interactions through code. |
| 7 | Interface with robotics actuators and sensors via code and microcontrollers. |
| 8 | Collaborate with peers to develop a robotic application that integrates multiple subsystems. |
| 9 | Apply concepts learned to make functional code that runs on hardware. |

*Weekly Schedule and Course Flow*

The course schedule spans 7 weeks, providing a structured and scaffolded approach to learning. Each week builds upon the previous one, progressively introducing more complex concepts and hands-on activities that are directly tied to robotics applications. Table 2 shows the details of weekly course schedule.

Table 2: Weekly Topics Overview

| **Week** | **Topics Covered** |
|---|---|
| 1 | Introduction to computers and programming concepts, covering Turing machines, binary systems, and variable types. |
| 2 | Structured programming concepts such as loops, conditionals, and functions, coupled with Lab 1 on basic robotic movement. |
| 3 | Data structures and algorithms, focusing on sorting, searching, and debugging techniques through practical robotics applications. |
| 4 | Simulation and visualization tools, including robot motion simulation using matplotlib and other libraries. |
| 5 | Advanced topics like state machines and event-driven programming, supported by lab exercises on navigation. |
| 6 | Graph representation and implementing graph algorithms, with a focus on robotic pathfinding strategies. |
| 7 | Integration of depth sensors for object detection and applying Dijkstra's algorithm in real-world robotic frameworks. |

This scaffolded approach ensures that students gain confidence and competence in applying programming concepts incrementally, culminating in a final project that synthesizes all course material. The iterative structure also provides opportunities for students to revisit foundational principles as they tackle increasingly complex robotics challenges.

*Course Components*

Key components of the course include:

- **Project-Based Assignments:** Students are immersed in projects that emphasize real-world applications and align closely with robotics engineering scenarios. These projects include programming robotic systems to execute tasks, developing algorithms for navigation using sensor data, and coding autonomous navigation routines. These assignments are structured to build incrementally, allowing students to master individual programming concepts before integrating them into comprehensive robotics solutions.
- **Interactive In-Class Activities:** Each week features dynamic, hands-on coding sessions through weekly labs where students collaborate on problem-solving exercises. These activities include live coding demonstrations, immediate application of newly introduced concepts in mini-projects, and peer programming sessions designed to foster teamwork and shared learning. Hardware-focused tasks, such as interfacing with sensors and actuators, provide students with practical experience that mirrors real-world robotics challenges.
- **Curriculum Design:** The course's curriculum is tailored to focus on essential programming principles such as control structures, modular design, data handling, and algorithm development. Each topic is introduced through robotics-centric examples, ensuring students understand the relevance of these concepts to their field. For example, loops and conditionals are taught in the context of controlling robotic actuators, while data structures are introduced through applications like pathfinding algorithms.

Through these innovative components, the course equips students with not only the technical proficiency required for advanced robotics courses but also the confidence to apply these skills in collaborative, real-world settings. This comprehensive approach ensures that students are well-prepared to meet the demands of both their academic progression and future careers in robotics engineering.

The instructors have used feedback from students and faculty and iteratively designed the course. Student feedback, collected through surveys and course evaluations, is analyzed to identify aspects of the course that are effective and areas needing improvement. Faculty feedback, particularly from those teaching subsequent courses, have provided insights into the long-term efficacy of the programming skills acquired. This feedback has informed annual course revisions, ensuring that the curriculum remains responsive to student needs and industry trends.

**Evaluation Methodology**

To assess the effectiveness of the "Programming for Robotics" course, a comprehensive and multi-faceted evaluation methodology was employed, blending both qualitative and quantitative approaches to gain a thorough understanding of the course's impact.

To this end, pre- and post-course surveys were administered to gather detailed insights into students' perceptions of the course's relevance and effectiveness and readiness for advanced courses. Questions explored their preparedness for advanced robotics courses, confidence in programming skills, and satisfaction with the hands-on learning approach. These surveys also included open-ended sections to capture nuanced feedback about specific aspects of the course.

Additionally, instructors from advanced RBE courses, who had first-hand experience observing students' performance through grades, project outcomes, and technical assessments, were interviewed to provide valuable insights into students' preparedness and programming capabilities. Faculty provided qualitative assessments on whether the course effectively bridged prior gaps in programming skills and how students performed in applying these skills to complex robotics problems. Anecdotal evidence from instructors further enriched the evaluation.

This methodology ensured a holistic view of the course's effectiveness, capturing both measurable outcomes and qualitative experiences to guide future iterations of the curriculum.

**Results and Analysis**

Preliminary results demonstrate that the "Programming for Robotics" course has had a transformative impact on student readiness and engagement. Key outcomes include:

- **Quantitative Gains and Enhanced Cohort Feedback:** The course demonstrated significant positive outcomes in student learning and engagement. Students who took the course in the first offering (Fall 2023) rated its quality at 87% on a scale of 1 to 5 (i.e. 4.35/5). Students who took the second offering (Fall 2024) similarly rated its quality at 85% (i.e. 4.25/5). Furthermore, they indicated that their learning in this course was 34% greater compared to their average learning in other college courses they have taken. These indications demonstrate the success of the course. This growth was evident not only in self-assessments but also in graded assessments.
- **Qualitative Feedback:** Qualitative feedback further reinforced these findings. Students consistently rated the hands-on, project-based approach as instrumental in bridging the gap between theoretical programming knowledge and practical robotics applications. Open-ended feedback highlighted specific elements such as live coding sessions and team-based projects as particularly beneficial. Many students expressed increased motivation and a sense of accomplishment when they could see the direct application of programming skills to robotics challenges, fostering deeper engagement and enthusiasm for the field. Students exhibited significantly higher confidence and competence in applying programming concepts to robotics challenges. Feedback from faculty in advanced courses further corroborates this observation, with instructors noting a marked improvement in students' ability to integrate programming into complex robotics projects. Anecdotal evidence includes students successfully applying concepts such as sensor integration and algorithm optimization in subsequent coursework.
- **Improved Readiness for and performance in RBE Advanced Courses:** Surveys conducted in the RBE advanced courses provided valuable insights into student readiness and how the programming course impacted their preparation for subsequent RBE core courses, such as *Introduction to Robotics*, *Robot Actuation*, *Robot Sensing*, *Robot Manipulation*, and *Robot Navigation*. These targeted surveys asked students about their prior programming experience, their confidence in applying programming concepts, and the challenges they faced during the course. For instance, students were asked to report whether they had taken recommended courses, such as CS programming course or the new

RBE-tailored programming course, and whether they felt the new course, if taken, adequately prepared them for more advanced robotics coursework.

*Introduction to Robotics:*

To evaluate the impact of the Programming for Robotics course on student preparedness in subsequent advanced RBE courses, we started with the Introduction to Robotics course, where the programming course was directly recommended as a foundational background. Surveys were conducted across two offerings of the Introduction to Robotics course in Fall 2024, each taught by different instructors. The first offering received 60 student responses, while the second received 57 responses. The primary goal of the survey was to assess how the programming course contributed to students' readiness for the Introduction to Robotics course.

Students were asked specific questions, such as whether they had completed the new programming course and if a lack of programming skills hindered their performance. Strikingly, in each offering, only one student who had completed the programming course reported that insufficient programming skills affected their performance in the Introduction to Robotics course. This evidence underscores the effectiveness of the programming course in equipping students with the necessary skills and confidence to excel in subsequent robotics courses. The results highlight the course's role in bridging foundational gaps and enabling students to meet the challenges of more advanced coursework seamlessly.

*Robot Actuation and Robot Sensing:*

To further evaluate the impact of the Programming for Robotics course, we extended our analysis to two subsequent courses, Robot Actuation and Robot Sensing, which build directly on concepts introduced in the Introduction to Robotics course. Since the Introduction to Robotics course has been revised to account for the inclusion of the programming course in the curriculum, this survey provided valuable insights into how the programming course, combined with curriculum reorganization, has influenced student readiness for advanced coursework.

Surveys were conducted across three offerings: at the end of Robot Sensing course in Spring 2024 with 83 participants, at the end of Robot Actuation course in Fall 2024 with 41 participants, and at the beginning of the Robot Actuation course in Spring 2025 with 49 participants. The surveys aimed to determine whether the new course has prepared them for these advanced courses or the lack of programming skills hindered student performance, and to measure the effectiveness of the new programming course in addressing foundational gaps. Remarkably, only two students across all three offerings reported a lack of the programming skills necessary for succeeding in these courses, highlighting the significant effectiveness of the programming course in addressing foundational gaps and equipping students with the necessary skills and confidence to excel in advanced robotics courses.

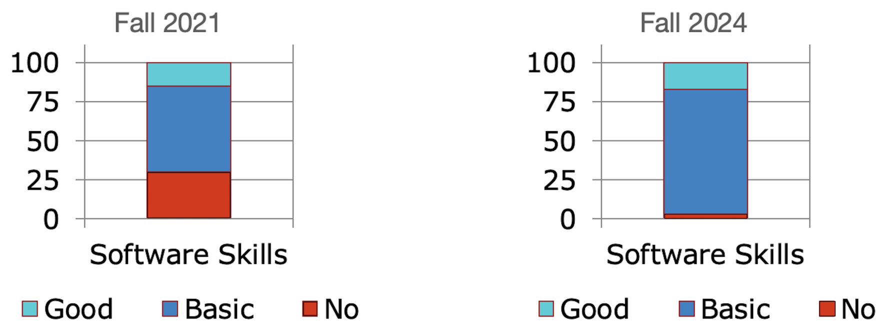


Figure 2: Comparison of the students readiness in Robot Manipulation course before (left) and after (right) adding the first-year RBE-tailored programming course to the curriculum. 'No' means insufficient readiness.

*Robot Manipulation:*

In Fall 2021, the CATME assessment process was implemented in the Robot Manipulation course, revealing a critical gap in programming readiness among students. The survey, conducted with 75 participants, showed that approximately 30% of students lacked the necessary software background, significantly hindering their performance. In Fall 2024, the same survey was conducted with 72 students who had the opportunity to take the Programming for Robotics course. The results demonstrated a dramatic improvement: fewer than 3% of students reported insufficient programming skills (Fig. 2). This stark contrast – from 30% in 2021 to under 3% in 2024 – highlights the transformative impact of the programming course and the curriculum revisions in equipping students with the foundational skills needed for success in advanced robotics courses.

These results and key findings are summarized in Table 3.

Table 3: Effectiveness of the Programming Course: Survey Results for RBE Core Courses

| Course | Survey Period | Participants | Key Findings |
|---|---|---|---|
| Introduction to Robotics | Fall 2024 | 60 | Only 1 student reported insufficient programming skills; the programming course significantly improved readiness. |
| | Fall 2024 | 57 | Similar findings, reinforcing the course's effectiveness in preparing students for advanced coursework. |
| Robot Actuation | Fall 2024 | 41 | Programming course effectively bridged foundational gaps, enabling seamless progression in robotics topics. |
| | Spring 2025 | 49 | Only 2 students across all offerings (Actuation & Sensing) indicated lack of programming skills. |
| Robot Sensing | Spring 2024 | 83 | Similar to Actuation, minimal students faced challenges due to lack of programming skills. |
| Robot Manipulation | Fall 2024 | 72 | Less than 3% of students reported insufficient programming skills, demonstrating the course's impact. |

## Broader Implications

The success of the ”Programming for Robotics” course underscores the transformative potential of discipline-specific education tailored to bridge the gap between foundational skills and practical applications. By directly aligning programming instruction with the demands of robotics engineering, this initiative has demonstrated that early interventions can significantly enhance student outcomes and engagement.

This new programming course serves as a replicable model for other robotics engineering programs aiming to address similar challenges in programming education. Its emphasis on project-based learning and contextual relevance ensures that students not only gain theoretical knowledge but also develop the practical skills necessary to tackle real-world robotics problems. Furthermore, the course's adaptability makes it suitable for a wide range of institutions, regardless of their existing resources or curricular frameworks.

The broader impact of this initiative highlights the importance of embedding discipline-specific elements into foundational courses. Early exposure to relevant, hands-on applications fosters a deeper understanding of the subject matter and prepares students for advanced technical challenges. This approach not only enhances student preparedness but also contributes to their confidence and motivation, thereby reinforcing long-term academic and professional success. By demonstrating the effectiveness of targeted, early-stage interventions, this course sets a new benchmark for robotics education and inspires further innovation in curriculum design.

## Conclusion and Future Direction

This paper has presented the development, implementation, and evaluation of a tailored programming course designed specifically to address the needs of first-year robotics engineering students. By identifying and bridging a significant gap in programming proficiency, the course has enhanced student readiness for advanced RBE courses and mitigated challenges previously faced by up to 30% of RBE students enrolled in traditional CS programming courses. Through a project-based learning approach, the course has enabled students to directly apply programming concepts to real-world robotics projects, significantly improving their ability to meet the demands of advanced robotics coursework. The qualitative and quantitative evaluations have demonstrated the course's effectiveness in fostering student preparedness and confidence in robotics-specific programming contexts. Future efforts will focus on three key areas: (1) conducting longitudinal studies to track the long-term impact of the course on students' academic and professional trajectories, (2) refining the course structure based on ongoing feedback from students and faculty, and (3) incorporating emerging programming technologies and trends to ensure the course remains relevant and effective. The “Programming for Robotics” course represents a significant advancement in robotics education, with the potential to inspire similar initiatives across institutions, ultimately fostering the next generation of robotics engineers equipped to meet the challenges of a rapidly evolving field.

## References


[1] *Overview of CATME Features*. Available: https://info.catme.org/features/overview/. [Accessed: 1/15/2025].

[2] D.A. Kolb, *Experiential learning: Experience as the source of learning and development*, FT press, 2014.

[3] M. Agheli, G. Lewin, and M. Nemitz, “WIP: The Necessity of an RBE-Tailored First-Year Programming Course in the Robotics Engineering Curriculum,” in *2024 ASEE Annual Conference & Exposition*, 2024.