# WIP: The Necessity of an RBE-Tailored First-Year Programming Course in the Robotics Engineering Curriculum

**Dr. Mahdi Agheli, Worcester Polytechnic Insitute**

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

**Dr. Greg Lewin, Worcester Polytechnic Institute**

Dr. Lewin is an Assistant Professor of Robotics at WPI.

**Prof. Markus Nemitz**

Assistant Professor Äì Worcester Polytechnic Institute, 2020-Present Postdoc Äì Harvard University, 2020 PhD, MSc Äì The University of Edinburgh, 2018 BEng Äì Bochum University of Applied Sciences, 2012

I grew up in Germany where I studied electrical engineering in Bochum. I subsequently moved to Scotland to study electronics and robotics in Edinburgh. At Harvard, I focused on the development of embedded controllers for soft robots. Since 2020, I am an assistant professor of robotics engineering and electrical and computer engineering.

My research group (the Robotic Materials Group) focuses on the creation, modeling, and control of robotic soft matter. Instead of using soft materials for the design of manipulators, we are interested in robotizing otherwise passive elastomeric materials. The goal is to develop programmable robotic materials that are able to change their shape; insights from biological organisms will help to develop strategies for the control of morphology.

# WIP: The Necessity of an RBE-tailored First-Year Programming Course in the Robotics Engineering Curriculum


## Abstract

[This is a "Work In Progress."] In response to the evolving field of Robotics Engineering, Worcester Polytechnic Institute's (WPI) Robotics Engineering (RBE) department, renowned as a pioneer in Robotics education, has recognized the need for a tailored, first-year programming course within its curriculum to enhance students' success in subsequent years. This decision stems from collaborative surveys conducted with both students and faculty, revealing a gap in programming proficiency among RBE students as they progress into their second and third-year courses. Targeted surveys in these advanced courses further confirmed the significance of addressing this matter. A market review, informed by these insights, was conducted to shape the course description. RBE students at WPI traditionally undertake programming courses from the Computer Science (CS) department; however, up to 30% encountered difficulties due to the mismatch between their needs and the CS-taught curriculum. The key differentiator of the RBE-tailored programming course, in comparison to traditional CS courses, is its project-based nature, where students learn programming concepts and directly apply them to real robotics projects. This work primarily focuses on the imperative of introducing a dedicated first-year programming course into the RBE curriculum, designed specifically for robotics, while highlighting WPI RBE's pioneering role in robotics education and the project-based approach that sets it apart. We aim to enhance the educational experience and preparedness of our students, ensuring that they are well-equipped to meet the demands of the rapidly evolving field of robotics. The pedagogical theory and approach underpinning this course will be presented, and the expected outcomes will be discussed, along with methods of assessment to evaluate its effectiveness. This endeavor is an effort to further enhance our existing RBE curriculum's excellence and adapt to the changing landscape of robotics engineering education while inspiring existing and future RBE departments in their creation of a curriculum.


## Introduction

The Robotics Engineering (RBE) program at Worcester Polytechnic Institute (WPI) stands out as a leader of innovation and practical learning in the realm of engineering education. Renowned for its project-based and programming-intensive curriculum, the RBE program is meticulously designed to not only impart theoretical knowledge but also to ensure hands-on, experiential learning. Central to this curriculum are core courses such as Introduction to Robotics (RBE 1001), Actuation (RBE 2001), Sensing (RBE 2002), Manipulation (RBE 3001), and 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 approach adopts a more flexible model. Here, while prerequisites are not rigidly enforced, a set of recommended backgrounds is provided for each course. This unique approach, emblematic of WPI's commitment to fostering a diverse and inclusive learning environment, 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. To navigate this diversity in student backgrounds, some RBE faculty members employ CATME (Comprehensive Assessment of Team Member Effectiveness) surveys [1]. These surveys are designed to assess various dimensions of team dynamics and individual contributions in educational settings. These surveys typically include questions that probe into team roles, communication, problem-solving, and individual responsibilities. The average completion time for these surveys is approximately 15 to 20 minutes, ensuring a thorough assessment without causing respondent fatigue. Validation studies have supported the reliability and effectiveness of CATME surveys, making them a trusted tool in academic and organizational environments. CATME surveys serve a dual purpose: they not only gauge the students' existing knowledge and skills in programming but also assist in forming balanced groups that align with their skill levels. By doing so, the faculty can tailor the learning experience to meet the diverse needs of their students, ensuring that all participants can achieve the course's learning objectives. This thoughtful approach to group formation and skill assessment underscores the commitment to providing a holistic and inclusive educational experience, one that recognizes and addresses the varied academic backgrounds of the students.

In this paper, we delve deeper into the challenges and solutions pertaining to the programming preparedness of students in the RBE program at WPI. We explore the innovative strategies employed to ensure that every student can thrive and excel in this dynamic and demanding field.

**Problem Identification**

At WPI's RBE department, a critical aspect of course management and student assessment hinges on the effective use of surveys including CATME surveys. CATME surveys are strategically employed in the first week of each term, primarily serving two key functions: assessing students background and facilitating team formation. The team-making process, crucial in a curriculum that emphasizes collaborative and project-based learning, benefits significantly from the insights provided by these surveys. Moreover, the follow-up weekly CATME surveys, consistently administered throughout the term, offer a continuous evaluative framework, crucial for monitoring student progress and adapting teaching strategies to meet evolving educational needs.

In Fall 2021, the CATME assessment process was implemented in a third-year course on Robot Manipulation (RBE 3001). The study's sample consisted of 75 RBE students from the course, offering a representative cross-section of the RBE program's demographic and skill diversity. This sample size and composition provide a robust basis for understanding the programming skill variance within the cohort. The context in which these surveys were administered—during the initial phase of the course—ensures that the data reflects the students' current competencies and challenges. RBE 3001 traditionally expects students to possess a solid foundation in programming and serves as a critical juncture in the RBE curriculum, where the application of programming skills becomes increasingly complex and integral to course outcomes. However, a significant issue was promptly identified upon the survey results: a notable discrepancy in the programming preparedness among students. This variance in skill levels was not merely a marginal observation but a substantial one, with approximately 30% of the students indicating a complete absence of software background. Figure 1 shows the result of the survey taken by all 75 students in RBE 3001 where students indicate their level of knowledge in Software. This revelation was gleaned from the raw data of the CATME surveys, which highlighted the disparity in students' programming proficiency.

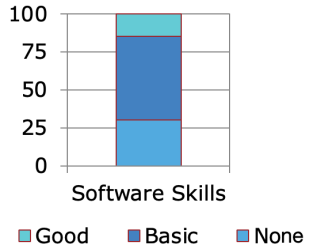


Figure 1: Data from CATME survey in RBE 3001 for all 75 students.

The knowledge levels in the survey were defined as follows: 'None' refers to little to no foundation; 'Basic' refers to familiarity with concepts; and 'Good' denotes a solid and operational level of proficiency. These categories were essential for stratifying students' programming skills, enabling targeted instructional strategies.

Compelled by these findings in the Manipulation (RBE 3001) course, we extended our inquiry to two additional core courses, "Sensing (RBE 2002)" and "Navigation (RBE 3002)," which have approximately the same number of students, to determine if this issue was isolated or prevalent across other courses as well. This time, in addition to CATME surveys, customized surveys that included more specific and targeted questions were employed. Instead of merely inquiring about programming skills as a whole, these surveys were designed to also gauge the specific backgrounds expected from students, providing a more nuanced understanding of their preparedness and areas for improvement. As shown in Figure 2, the results from these self-assessment surveys echoed the concerns raised in RBE 3001. Here, 14% of students in RBE 2002 and 27% of students in RBE 3002 indicated that they possessed little to no programming skills, despite having been through courses that involved a considerable amount of programming. In addition, 70% of the students in RBE 2002 and 50% of those in RBE 3002 indicated having scant experience in C++ and Python, respectively—skills considered beneficial or "good to have"

for their respective courses. This confirmation of the problem's existence across different core courses was a clear indication that the issue was systemic and not confined to a single course.

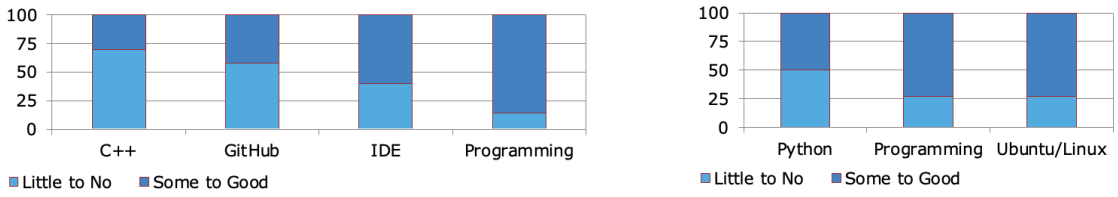


Figure 2: Survey results: RBE 2002 (left) and RBE 3002 (right) for all students in Spring 2022.

These revelations presented a red flag for the RBE curriculum, underscoring a critical gap in the programming preparedness of students. Such a gap poses significant challenges, not only for the students struggling to meet the course demands but also for the overall effectiveness of the RBE program. It became evident that addressing this inconsistency in student preparedness was imperative, not only for the success of individual students but also for the integrity and efficacy of the RBE curriculum as a whole. The following sections of this paper will delve into the methodologies adopted to investigate the root causes of this problem and explore the solutions and actions taken to bridge the programming skills gap among RBE students.

## Methodology

The analysis primarily leveraged processed output results from the CATME and the additional customized surveys, focusing on identifying trends and disparities in programming skills among the RBE students. The findings from the surveys revealed a significant disparity in programming skills, with a notable proportion of students displaying only basic or no proficiency. These results underscored the need for a curriculum intervention to provide a more solid programming foundation.

In light of these findings from the surveys, the necessity to address this disparity in student preparedness became a pressing priority for the RBE program. To address the problem of existing gap in students' background in terms of software and programming preparedness, we first planned to realize the sources causing the problem. This necessitated going beyond merely interpreting the conducted survey results. Therefore, a comprehensive methodology was adopted to execute a thorough review of the core curriculum by looking at recommended background for courses as well as the topics covered in the RBE core courses.

### *Recommended Background*

To identify the sources of the problem, we collected data related to recommended backgrounds in core courses through both student self-surveys and student listening sessions. We specifically aimed at finding answers to the following questions: 1) Are students taking recommended backgrounds as recommended?, 2) Are recommended backgrounds listed where they should be?, and 3) Are there any recommended background missing from the list? Figure 3 illustrates an example of a survey result related to the first question where some students indicated that they have not taken the recommended backgrounds as suggested.

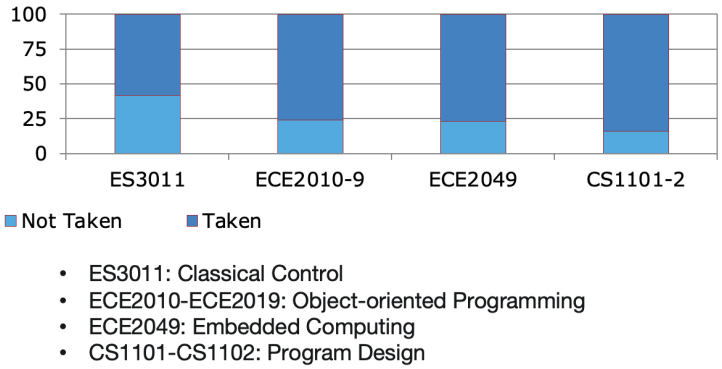


Figure 3: Survey results: Recommended Backgrounds for all RBE 3002 students in Spring 2022.

*Topics Covered in RBE core curriculum*

We collected inputs related to Course Topics from both faculty and students. We initiated a discussion among the RBE faculty by sharing course survey results with them soliciting their input. For students feedback, it was in the form of listening sessions. For this part, we specifically aimed at finding answers to the following questions: 1) Are all necessary Topics/Skills covered in the curriculum?, 2) Are there any gaps in the curriculum?, and 3) Are topics in the correct order?

**Findings**

Based on the collected data, we were able to find the sources of the problem of students not being prepared for courses in terms of programming and software engineering. Issue 1: The order in which Recommended Backgrounds are suggested contribute as part of the source of the problem. Issue 2: There is a gap in the curriculum related to programming topics/skills. Issue 3: Topics in early courses are not in an efficient order.

These findings suggested that there is a need to revise and improve our curriculum to address the identified challenges found in courses like Manipulation (RBE 3001) and Navigation (RBE 3002) in which the surveys revealed that a notable percentage of students had little to no programming background.

**Proposed Solutions and Actions**

In response to these findings, we first proposed several high-level solutions, including reorganizing recommended backgrounds to address Issue 1, adding programming topics to the core curriculum by either creating a new course or modifying existing courses to address Issue 2, and reorganizing course topics in RBE early courses to address Issue 3. A curriculum summit was held to discuss these solutions, leading to a rethinking and reorganization of the recommended backgrounds and course topics. At this point, it was obvious that while recommended backgrounds suggested for core courses needed to be revised and reordered, we also needed to either incorporate realized missing topics into the available RBE courses or create RBE specific courses.

**Market Review and Option Evaluation**

A market review of RBE programs at other universities was conducted. The objective was to get an accurate and realistic idea of what/how others are doing and to use the gathered data to boost our curriculum as necessary. We reviewed more than 10 universities that offer major or minor in Robotics Engineering across the globe. We have mainly reviewed the degree/major/minor requirements as well as the curriculum Themes i.e. Core Areas and Courses. Our study confirmed that the absence of a programming course in the first year was a significant gap in our curriculum.

**Implementation and Outcomes**

Several options were proposed, including having our students take an introductory programming course from the CS department such as Introduction to Programming (CS 1004) into the RBE curriculum, working with CS department to have their course (RBE 1004) tailored toward our goals for RBE students, creating a new RBE-specific programming course, and finally breaking one of our RBE introductory course namely Introduction to Robotics (RBE 1001) into two courses to incorporate programming topics throughout the two courses.

As can be seen in Table 1, Pros and Cons of each proposed solution was investigated. Throughout this thorough investigation, which was performed by anticipated instructors of the course together with the help of the members of the Curriculum Committee and other related faculty from both RBE and CS departments, several factors were taken into consideration, including resource needs, easiness of implementation and practicality, impact on other courses, simplicity, and efficiency. These considerations were imperative in understanding how effectively the solution will address the identified issues in the curriculum.

Table 1: Pros and Cons of the proposed solutions.

| Options | Pros | Cons |
|---|---|---|
| Option #1: **Use CS 1004** | ✓ No resources required from RBE<br>✓ Minimum change in Curr./RBs<br>✓ Easy to implement<br>✓ Independent of other options | X CS 1004 is too general<br>X The demand of CS 1004 may increase dramatically<br>X Cannot enforce as a prerequisite |
| Option #2: **Tailor CS 1004 Toward RBE** | ✓ No resources required from RBE<br>✓ Minimum change in Curr./RBs<br>✓ Useful for our purpose<br>✓ Makes option 1 more effective | X May not be fully practical<br>X Cannot enforce as a prerequisite |
| Option #3: **Offer our own RBE Course** | ✓ Having full control over the course<br>✓ Would directly address the issues<br>✓ Easier to enforce as a prerequisite | X Resources: Needs someone from RBE to teach it<br>X May become a service course |
| Option #4: **Offer RBE1001-A/B** | ✓ Students learn as they go<br>✓ Breaking a heavy course into two manageable courses<br>✓ No prerequisite issue | X Resources: Needs someone from RBE to teach it<br>X Significantly affect current RBE 1001<br>X Makes two courses dependent - cannot be waived |

Finally the chosen solution was Option #3 i.e. to develop a new experimental RBE programming course for the first year, “Programming for Robotics (RBE 100X)”. Here, “X” indicates an experimental course. This course replaced the CS introductory course (CS 1004) for RBE

students and delivered to students in Fall 2023 for its first time. Consequently, the recommended backgrounds for higher-level courses in the second and third years were updated to reflect the addition of the new programming course. Accordingly, the topics in some of the early courses such as RBE 1001, RBE 2001, and RBE 2002 were reorganized and reordered to ensure minimizing the overlap and maximizing the efficiency.

The introduction of a first-year programming course tailored for RBE students aimed to bridge the identified skills gap. This course, "Programming for Robotics," is designed with specific learning outcomes that align with the identified needs, focusing on practical and applicable programming skills in a robotics context. Its integration into the curriculum is anticipated to enhance student preparedness and engagement, thereby enriching the RBE program's overall educational efficacy.

The course design for "Programming for Robotics" is grounded in several key pedagogical theories, notably Constructivism and Experiential Learning. Constructivism suggests that students build knowledge through experiences and reflection, aligning with the course's hands-on, project-based approach. Experiential Learning Theory [2], emphasizes learning through experience, further supporting the course's design where students actively engage in programming tasks relevant to robotics, thus facilitating deeper understanding and skill acquisition. These theories inform the course structure, ensuring that students are not merely passive recipients of information but active participants in their learning journey.

Distinctively structured to meet the unique demands of RBE students, the course sets itself apart from traditional introductory programming courses in several key aspects:

- *Assignments:* The assignments in this course were specifically designed to integrate programming with robotics applications. For instance, students work on coding projects that involve controlling robotic kits, which directly apply programming skills to robotics contexts, unlike conventional assignments that may focus on general computing concepts.
- *In-Class Activities:* The course emphasizes hands-on learning and real-time problem-solving. In-class activities include live coding sessions, peer programming, and immediate application of concepts in mini-projects that involve hardware interfaces, sensors, and actuators. This practical focus is a departure from the more theory-centric approach often found in traditional programming courses.
- *Course Content:* While covering fundamental programming concepts, the course content is infused with robotics-specific examples and applications. This ensures that students not only learn programming but also understand how to apply these skills in the RBE context.
- *Collaborative Learning:* The course is aimed to promote collaborative learning through team-based projects that mimic real-world robotics programming challenges. This approach fosters a deeper understanding of both the individual and collective aspects of software development in robotics.

This tailored approach is to ensure that students gain not only programming proficiency but also an understanding of how these skills intersect with robotics, preparing them for advanced courses and real-world applications.

The effectiveness of the course was evaluated through a combination of formative and summative assessment methods. Formative assessments included weekly programming assignments and peer reviews, providing continuous feedback and opportunities for improvement. Summative assessments comprised a final project and a comprehensive exam, measuring students' mastery of the course objectives. Additionally, after the course was successfully offered for the first time in Fall 2023, student self-assessments and course evaluations were employed to gather feedback on the learning experience, which is instrumental in refining the course design. Students who took the course rated its quality at 87% on a scale of 1 to 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. The authors will continue to investigate the broader impact of this solution as students progress to their higher-level courses.

Moving forward, we will use feedback from students and faculty as it plays a pivotal role in the iterative design of 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, provides insights into the long-term efficacy of the programming skills acquired. This feedback informs annual course revisions, ensuring that the curriculum remains responsive to student needs and industry trends.

## Conclusion

During this process, we observed a problem in a third-year course led to investigating the depth and breadth of the problem. Data and evidence were collected and sources and needs were identified. Possible solutions with pros and cons were suggested and evaluated. faculty input was solicited to validate the solutions. We planned accordingly and implemented the solutions e.g. new course was added and topics and RBs were reorganized. The new course was offered in Fall 2023 for the first time, and its effectiveness was assessed through final course evaluations where students' responses indicated the success of the course. This initiative represents a significant enhancement to the WPI's RBE curriculum. By addressing students' programming preparedness, the program aims to provide a more robust educational foundation for future robotics engineers.

## Future Work

The impact of this solution is to be investigated as students move to their higher level courses and is a topic for another paper in the future as we are tracking the impact of offering this course on addressing the identified issues in our curriculum.

## Acknowledgements

The authors would like to extend their special thanks to the WPI RBE Curriculum Committee, RBE Undergraduate Program Committee, instructors of RBE core courses, CS-affiliated RBE faculty, and the CS curriculum committee for their invaluable contributions to this effort. Additionally, although the content of the references in the following section was not directly used, the authors found them both insightful and helpful and are therefore citing them here [3]-[7].

## References


[1] *Overview of CATME Features*. Available: `https://info.catme.org/features/overview/`. [Accessed: 3/31/2024].

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

[3] V. Sukackė, . A. O. P. de Carvalho Guerra, D. Ellinger, V. Carlos, S. Petronienė, L. Gaižiūnienė, S. Blanch, A. Marbà-Tallada, A. Brose, "Towards Active Evidence-Based Learning in Engineering Education: A Systematic Literature Review of PBL, PjBL, and CBL," in *Sustainability*, vol. 14, no. 21, 13955, 2022.

[4] C. Herodotou, M. Sharples, M. Gaved, A. Kukulska-Hulme, B. Rienties, E. Scanlon, D. Whitelock, "Innovative Pedagogies of the Future: An Evidence-Based Selection," in *Frontiers in Education*, vol. 4, pp. 113, 2019.

[5] E. Petraki, D. Herath, "Teaching and learning robotics: a pedagogical perspective," in *Foundations of Robotics: A Multidisciplinary Approach with Python and ROS,* Springer, pp. 43–62, 2022.

[6] M. Ciaraldi, E. Cobb, F. Looft, R. Norton, T. Padir, "Designing an undergraduate robotics engineering curriculum: unified robotics I and II," in *2009 Annual Conference & Exposition*, Paris, pp. 14–428, 2009.

[7] B. Ikeda, J. Hoelscher, R. Alterovitz, D. Szafir, "Guiding the Development of Undergraduate Educational Robotics," in *Proceedings of the ACM Conference on Global Computing Education*, vol. 1, pp. 71–77, 2023.