Engineering Educators Bringing the World Together
2025 ASEE Annual Conference & Exposition
Palais des congrès de Montréal, Montréal, QC • June 22–25, 2025
Paper ID #46891

# Rock paper symbols: Leveraging the spiral curriculum to teach coding in primary schools

**Dr. Brendan Jacobs, The University of New England**

Brendan Jacobs is an Associate Professor and Head of Department (STEM Education) at the University of New England, Australia. His research interests are in STEM education, explanatory animation creation, conceptual consolidation and digital scholarship.

**Solina Quinton, University of Waterloo**

Solina Quinton is an environmental engineering graduate from the University of Waterloo with a passion for education. Her experience in the field of engineering education began during her time in Australia working on The SILO Project, an integrative STEM education resource for primary students. Solina has continued this work to focus on the Ontario curriculum, and her goal is to continue to connect with and further inspire students' interest in STEM learning.

**Dr. Rania Al-Hammoud, University of Waterloo**

Rania Al-Hammoud is an associate professor teaching stream and the current associate chair of undergraduate studies at the civil & environmental engineering department at university of waterloo. Dr. Al-Hammoud has a civil engineering background with research focusing on materials and the rehabilitation of reinforced concrete structures. She also has passion for engineering education and has published widely in this area. She cares about the success and well-being of her students, thus always being creative with the teaching methods in the classroom. She also won multiple teaching awards.

# Rock paper symbols: Leveraging prior knowledge to teach coding in elementary schools

## Abstract

As technology becomes increasingly essential in society today, the next generation of rising professionals will need to be well-versed and fluent in the language of the future; coding. The problem at hand is the perception around this field. Coding can be difficult and complex to grasp, dissuading students from the field rather than empowering them to embrace the problem-solving aspects of it. This raises the question; how can we engage the students of today to be the problem-solvers of tomorrow, using coding as an instrument for success? The answer to this can be solved through the early introduction of coding in elementary schools.

This paper reports on a STEM integration project based around the co-creation of 28 integrated units of work, one for each term over the seven elementary years. Through the incorporation of hands-on and interactive activities to better engage learners in classroom activities, students are introduced and re-introduced to the concepts of coding using pseudocode, flowcharts, and conditional statements to progress from the familiar game of rock paper scissor, to a technological version that utilizes the block-based coding devices, Micro:bits. In the delivery of these lessons and activities, we observed increased levels of enjoyment and understanding of the concepts being taught from the students participating, solidifying their learning.

This hands-on approach can be applied to teach coding fundamentals while keeping students engaged, potentially growing interest in coding and STEM fields for the future. The intended outcome and assessment criteria of the task was embodied in the successful creation of the code to play the rock paper scissors game. Differentiation of learning was achieved by varying the complexity of the task from user-generated through to randomly generated icons.


## Introduction

Coding is becoming more prevalent as the technological world advances. Its applications are extending beyond the computer world and into the realm of education, with findings revealing that through teaching coding, students gain additional skills such as problem-solving, critical thinking, social skills, self-management and other academic skills relating to fields outside of math and programming [1].

Bers [2] described coding as another language in the context of teaching computer science in early childhood. Just as reading and writing are essential skills developed in the early years of education, coding will need to fall into this category as the field of education progresses to adapt to the world's technological changes, considered a "basic literacy" for the future [3]. In the context of this study, coding refers to the process of giving instructions in a language that results in an understandable output stemming from the directions received. This extends beyond what is commonly thought of as a coding language as it can even refer to hand symbols – such as rock, paper, and scissors. The use of symbols as a means of conveying information is a form of coding that can be used to engage learners in a more simplified approach to learning the language.

In elementary schools around the world, coding is being taken to a new level of importance. While certain countries have coding lessons mandated in classrooms, some rely on the initiative of teachers to incorporate these concepts and activities. Teacher confidence in the subject is an essential component of coding in classrooms. It has been found that a lack of resources exists for teachers to confidently lead engaging lessons in the subject – this lack of resources was noted as the primary selection in a survey regarding teacher's external apprehensions in the teaching of coding, as seen in Figure 1 [4].

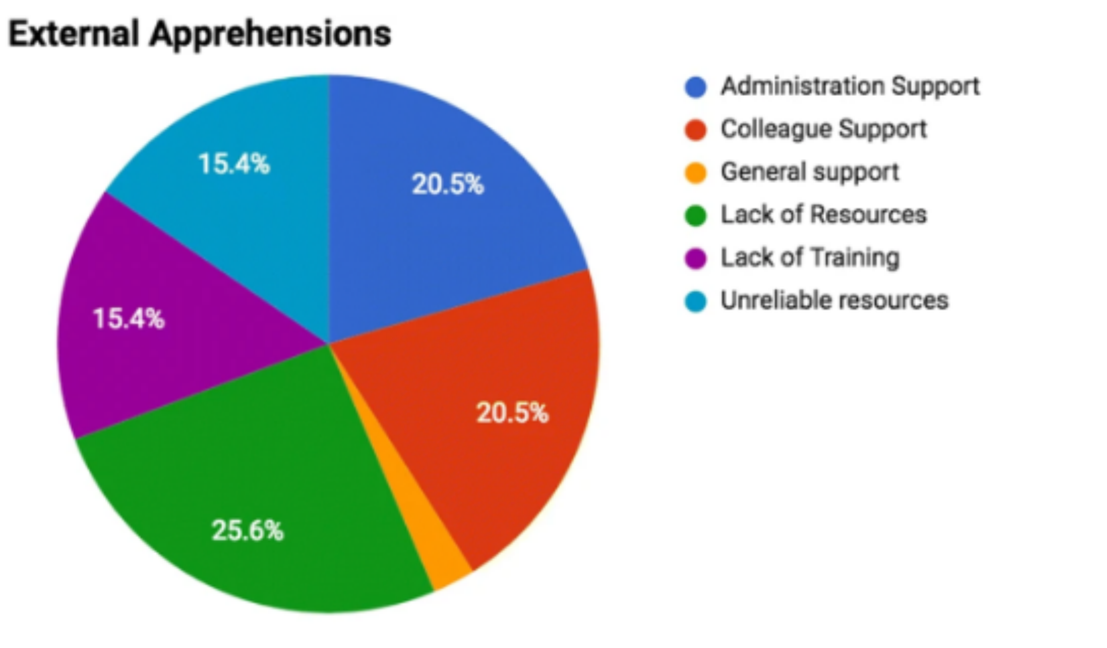


***Figure 1:*** *External Apprehension Teacher Survey Results, [4]*

One tool that has proven useful for both teachers and students in introductory coding in classrooms is the Micro:bit [5]. "The Micro:bit is a small computer designed for young people and students to learn how to code and make interactive projects" [6] for fun and engaging learning. Coding is normally done on the Make code website (https://makecode.microbit.org/) before it is transferred to a Micro:bit using Bluetooth or a USB cable. These small, programmable devices bring many benefits to the classroom, even beyond the standard realm of STEM. They act as a gateway to the world of coding that is opened through the device's ease of use and high level of engagement. In fact, by fostering an environment of creativity where students can use this coding platform to create and solve new ideas, students are more inclined to *want* to learn more about the tool, outside of the classroom as well as inside [5].

With the use of Micro:bits, there is an expansion of new lessons and activities that can be taught in the classroom to take the step to coding to the next level [5]. This is a beneficial extension of the block-based coding that has been used in schools increasingly around the world [7]. Block-based coding is a form of teaching computer science using appealing shapes and visuals to allow a drag-and-drop format of developing code through simplified instructions rather than trying to understand the complicated syntax [8]. This is an approach that is used through the Make code website connected to Micro:bits, which can be seen in the interface shown in Figure 2.

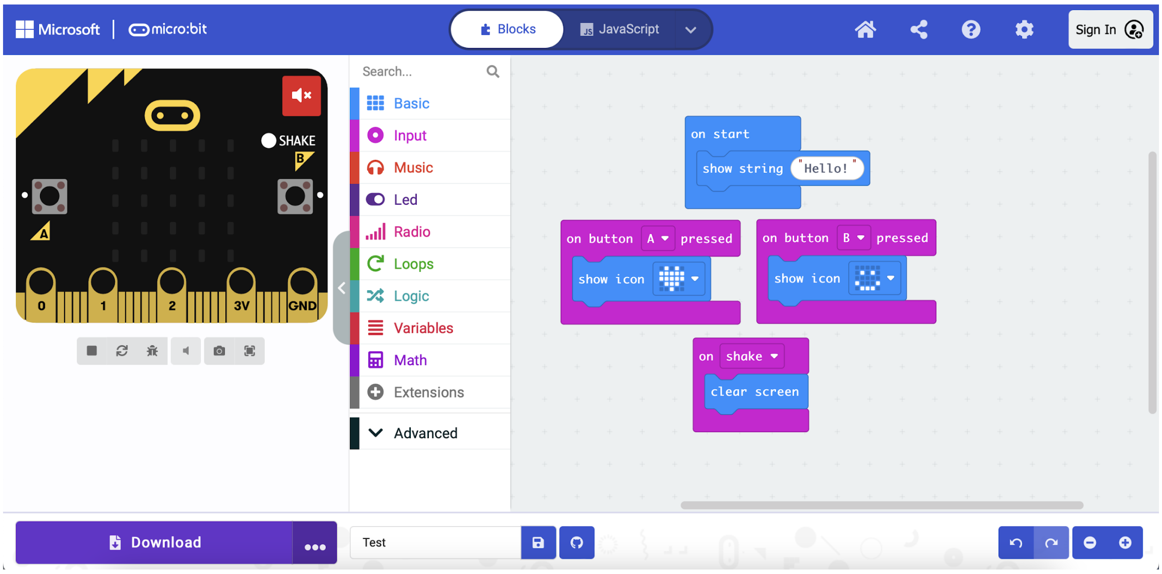


***Figure 2:*** *Micro:bit Website Interface, https://makecode.microbit.org/*

By toggling to the JavaScript tab, students have the option to view the actual syntax, making visual connections between the blocks and the code. This approach allows students to take on concepts of coding with confidence, aided by the visual ease and accessibility of this tool. Combining this approach with gamification is a recipe for success, giving students goals and challenges to keep them engaged [8].

This paper outlines a hands-on approach used with elementary school students in Years 1 to 6. The activities were intended to increase student's confidence in coding through transitioning from an already-understood game (rock paper scissors) to an electronic version of it using the Micro:bit, resulting in high levels of both engagement and motivation in the learners. Although it has become increasingly common for coding to be taught to children in elementary schools, it is generally presented as a way in which students can do new things using new coding knowledge and skills. The approach which we have taken is to start with something familiar to the students and then use this as the basis for skill development in coding. The following examples of coding in elementary schools are taken from a longitudinal study called The SILO Project, where the activities outlined in this paper have been created and implemented.

## The SILO Project

SILO is an acronym for Scientifically Integrated Learning Outcomes, but it is also a play on words because formal education is often criticised for teaching content in silos instead of using an integrated approach. The SILO Project is an Australian study which began in 2021. There are two aims of The SILO Project as follows:

1. To articulate a scope and sequence for elementary STEM education.
2. To develop 28 integrated STEM units and continue to refine them through implementation in elementary school classrooms.

The reason for having 28 units is to cover the seven years of elementary education over four terms (i.e., 7 x 4). The two research questions for The SILO Project are as follows:

1. What might an integrated STEM curriculum for K-6 students look like?

2. How can the co-design of learning sequences and activities between teachers and researchers be effectively undertaken to improve the quality and usability of project findings and recommendations?

The units being developed within The SILO Project are undergoing experimentation in two pilot schools. This involves classroom sessions which last for one hour each week. The participants in the study are the teachers in the two pilot schools, not the students. This is to address the second research question about classroom collaboration. The first research question is addressed through the SILO website which is updated on a daily basis at https://silo.edu.au/. All content on the website is made freely available with a Creative Commons CC BY-NC-SA 4.0 licence. No usernames or passwords are required to ensure that there are no access issues. The dynamics between new knowledge creation and the international emphasis on STEM education could be considered to be running in parallel due to the commonality of innovation, however, interventions in STEM education are often based around gains in student learning and engagement. The SILO Project takes a different approach at the methodological level through the creation of a new qualitative research methodology designed specifically to improve the quality and useability of project findings and recommendations without the need to quantify student achievement.

**Research methodology**

The research methodology used for The SILO Project is Provisional Multimodal Research (PMR) [9]. PMR is a qualitative methodology based on the constructionist principle that learning can be embodied in the creation of an artifact [10]. In our use of PMR, the central artifact is the SILO website in general and the 28 STEM units in particular (https://silo.edu.au/28stemunits.html). The explanatory strength of PMR is achieved by using version control to provide a chronology of both the product and the process. Figure 3 shows how the rationale for changes and the various iterations of artifacts creates a cycle for improvement. The key ingredient for the rationale for changes is professional judgment.

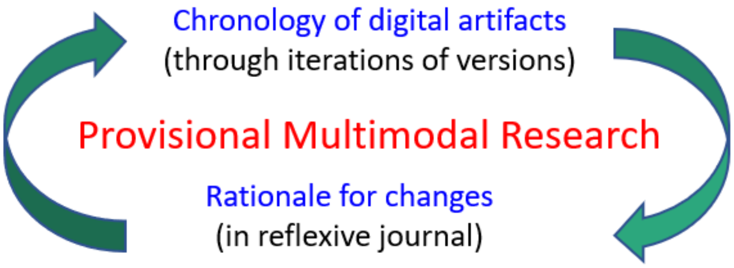


***Figure 3**: Provisional Multimodal Research*

PMR represents a paradigm shift in qualitative research as data is generated each time a learning sequence is articulated or enacted. The SILO Project is currently operating with only three researchers and around eight teachers across two schools, but the project is intrinsically scalable due to the multimodal approach. Each time a STEM unit (i.e., webpage) changes, the date is added to the old version and these HTML files are archived. Importantly, people who visit the SILO website are only ever presented with the latest version of each artifact and the reflexive journal is kept offline for the purposes of writing up the findings as they emerge.

PMR utilizes ‘referential chronology’ which is an extension of referential adequacy, formulated by Lincoln and Guba in [11]. PMR makes two important advances to referential adequacy. Firstly, the role of the researcher is quite different in PMR as they are the designer or co-designer of the learning artifact(s). The researcher’s reflexive journal is the primary mechanism to document data analysis because a rationale is provided for each iteration of an artifact. The rationale for these decisions is archived in the chief investigator’s reflexive journal rather than on the SILO website as teachers only want to see the latest versions of the STEM units. Confirmability is an interesting element within PMR due to the intrinsically subjective nature of design. Another researcher could look at the chronology of the evolving artifacts and confirm that they understand the logic behind the rationale for any changes made, but this does not necessarily mean that they would have made the same decisions. This also affirms Lincoln and Guba’s insight from 40 years ago when they noted that the “trouble with generalizations is that they don’t apply to particulars” [11]. In keeping with the dynamic nature of PMR, we have decided to weave any relevant pedagogical issues into the 'Data analysis and results' section to recount a chronology of development across the various coding activities.

## Data analysis and results

Data in PMR is seen as provisional because the objective is incremental improvement. Each version of an artifact is a discrete source of data which is analyzed in real time using professional judgment whenever changes are made. The following coding activities were conducted with various classes and age groups. These activities could be conducted in longer, single sessions but they are presented here as two separate sessions. Sequencing the activities in this manner is based on the spiral curriculum where Bruner [12] advocated the practice of building on the early introduction of concepts with increasing complexity.

### *Session 1: Micro:bit coding for user-controlled rock paper scissor icons*

An introductory problem-solving activity is to discuss how a rock paper scissors tournament could be organized. Based on the premise that losing a round of rock paper scissors will result in elimination, the question is posed about how to find a winner. There are many possible options for this including simply having students move around the room playing each other until a winner is found. Students who have lost should sit down so that the remaining players are still able to move around the room. A more suspenseful variation on the game is where each round happens at the same time as the students are then more focused as the number remaining of players is halved after each round. A graphic organizer for this approach is shown in Figure 4. Students should also be asked how to manage a tournament when the number of players is not 16. As this is an open-ended question there can be multiple solutions.

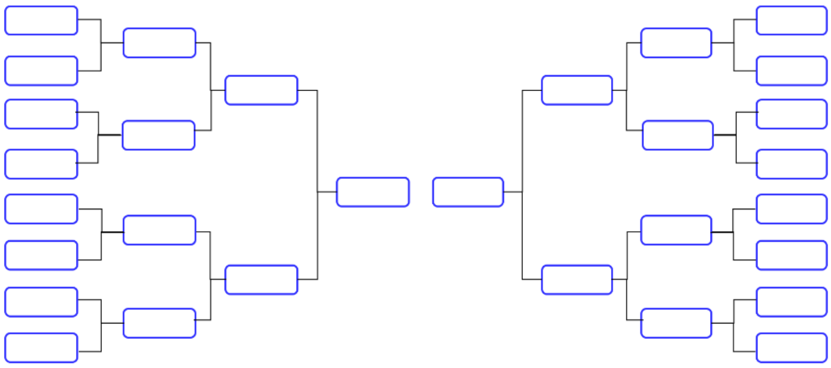

***Figure 4:*** *A Graphic Organizer for a 16-Player Tournament*

Coding the rock paper scissors game begins with students accessing https://makecode.microbit.org/ on their digital devices. The Micro:bit screens are an array of 25 LEDs so students can create their own designs for the three icons as shown in Figure 5.

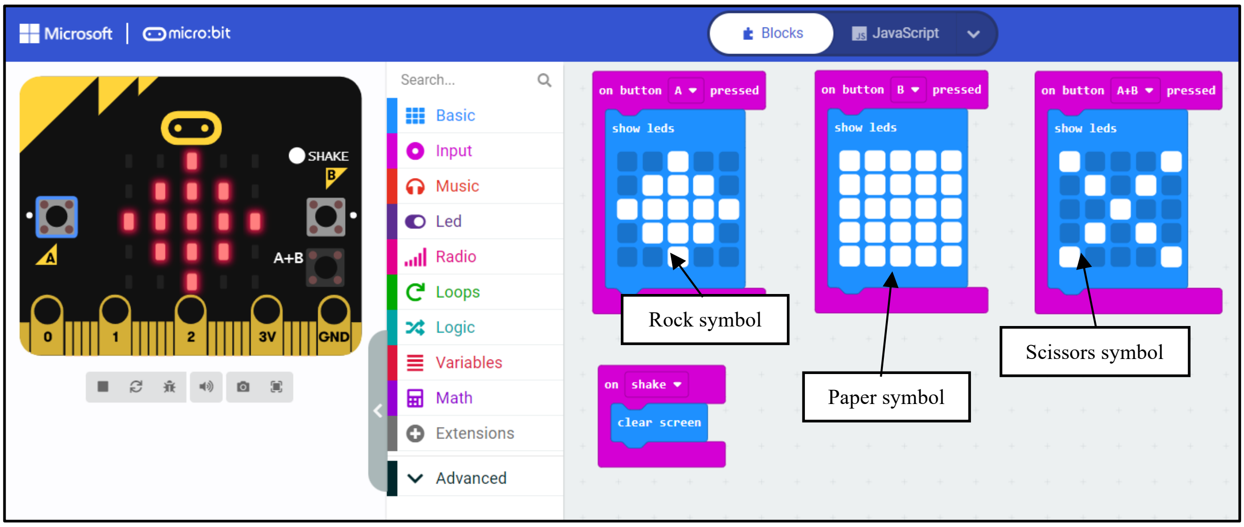


***Figure 5:*** *Coding Micro:bits for Rock Paper Scissors with Student Choice*

Once students have coded their devices, they can compete with one another in pairs. A discussion point relates to why the 'Shake' action is best used to clear the screen rather than generating an icon.

***Session 2: Micro:bit coding for randomly generated rock paper scissor icons***

Students are usually introduced to the basic symbols used for comparisons in Grade 1 (i.e., <, >, =). This knowledge can then be applied in the context of coding using pseudocode to explain how comparisons are commonly used in everyday devices. Pseudocode is an informal combination of coding concepts and ordinary language [13]. For example, an air conditioner can be set to start cooling at 23 degrees Celsius. The pseudocode for this is IF temperature is greater than 23 degrees THEN on. IF and THEN are conditional statements which can be readily understood by children by using familiar examples which might not even involve digital devices. For example, IF cold THEN put jacket on, IF the weather is sunny, THEN wear sunscreen, IF there are no cars coming THEN cross the road, and so on.

The following game builds on this knowledge in the context of coding. Figure 6 shows number cards and symbol cards. The numbers are between 1 and 100 but the symbols cards have = on one side and < on the other. Depending on the orientation of the < card it could be < or >. The game is played by giving each student a number card and a symbol card. Everyone then stands in a circle facing each other holding the cards in front of them so that they can be clearly seen. The number card must go in the right hand and the symbol card in the left hand. Depending on the number, each student must arrange their symbol card so that the sequence is correct. The teacher starts by stating their number followed by their symbol and the pattern continues moving clockwise around the circle. For example, "12 less than", "60 less than", "91 greater than", "15 less than…" and so on. Any errors in the symbol orientation should be fixed before moving on to the next person. When everyone in the circle has had their turn, the teacher counts 5,4,3,2,1 while the students shuffle into a new position in the circle, but they must keep the same two cards. This game was initially trialled with students as young as Grade 1 but it was found that this worked better with Grades 4, 5 and 6 as younger students often struggled to read their symbol cards upside down. Students should be encouraged to memorize their number cards as they keep the same number for each round of the game.

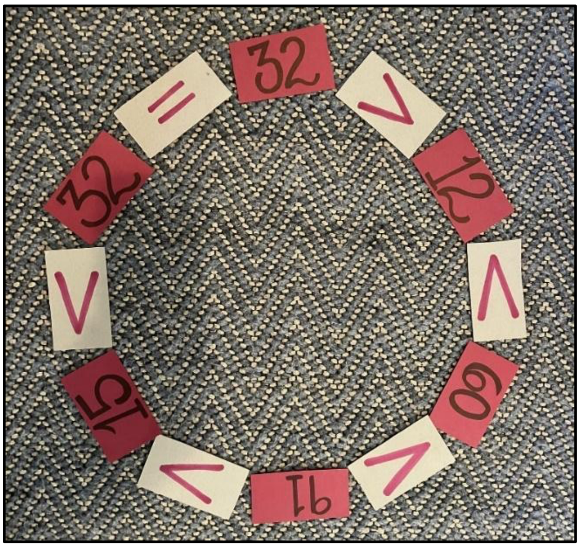


***Figure 6:*** *A Symbols Game Using Cards*

The complexity of the rock paper scissors game can be extended by coding the Micro:bits so that the device generates the icons randomly. Figure 7 is a variation on the code provided in a video provided by the Micro:bit Educational Foundation (https://www.youtube.com/watch?v=dIA06s9CZPw).

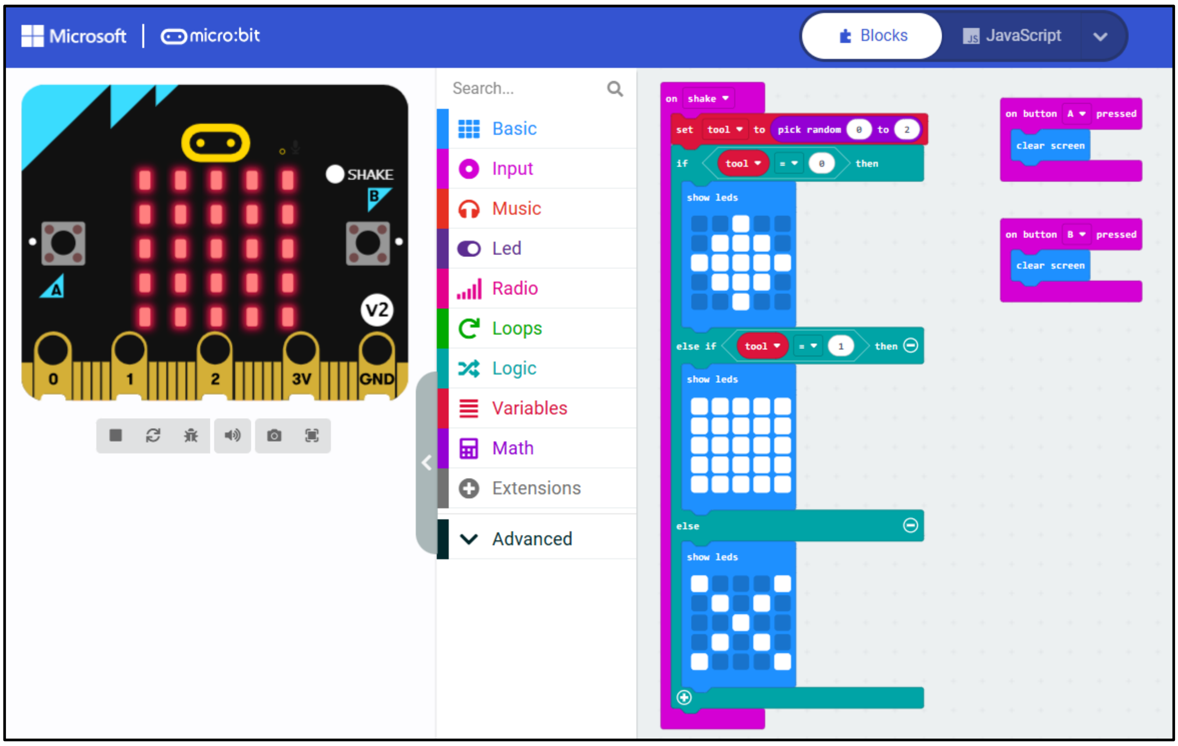


***Figure 7:*** *Coding Micro:bits for Rock Paper Scissors with Random Choice*

It is recommended that students watch this video as the presenter explains two important coding concepts as follows:

1. Although the Micro:bit could have been programmed to generate a random number between 1 and 3, the presenter chose to generate a number between 0 and 2 as a reminder that 0 is a number.
2. The use of ELSE introduces a new statement and is also a demonstration of how it is usually best to make the code as simple as possible. If the number is not 0 or 1 then it must be 2. This also saves time and creates a more elegant solution for the code.

Although exponents are not formally taught until high school, an opportunity to introduce exponents informally can occur by discussing the various combinations in the rock paper scissors game as shown in Figure 8.

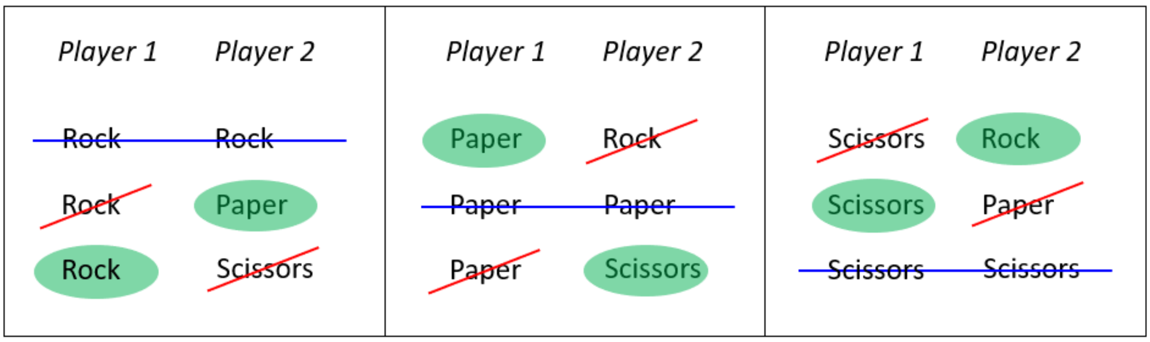


***Figure 8:*** *The Nine Combination for Rock Paper Scissors*

The teaching point is that for winning, losing or drawing, the probability is 3/9 which can be simplified as 1/3. The number 9 is evident from counting the options or from calculating $3^2$ (i.e., 3 options and 2 players).

An assessment rubric concludes each of the 28 STEM units but it is the same rubric for each topic. This is because each of the three rows in the rubric measures the stages of conceptual consolidation, namely, the use of correct terminology, identifying relevant components and then understanding how the components function together [9]. There was no comparison made between students who used the approach advocated in The SILO Project and those who did not. A rationale for this is evident in a new section about rocket science in SILO 3.2 'Satellites'. As rocket science is not part of the curriculum in elementary schools, it is self-evident that students exposed to rocket science activities will learn more about this topic than students who did not have this opportunity. This also affirms the enduring utility of Bruner's spiral curriculum [12] where concepts can be introduced at any time because they will be revisited with increasing levels of complexity.

**Conclusion and recommendations**

This paper proposed a novel instructional approach where teachers and researchers worked together to find out what is feasible in the classroom and then make this information publicly available on a website. The PMR methodology facilitated rapid prototyping where activities and refinements are often made within a matter of hours as the hallmark of PMR is incremental improvement through version control. This paper has also shown how prior knowledge can be used to give students confidence and clarity about the purpose of their coding task as all of the students were familiar with playing rock paper scissors. This enabled the actual coding task to be seen as a novelty because the end goal was in sight. Although a range of achievement was evident with the second coding challenge where the Micro:bits randomly generated the rock paper scissors outcomes, all students were able to successfully code the human-generated icons in the first challenge which meant that no child was left behind. Classroom experience showed that connecting Micro:bits to tablets using Bluetooth was sometimes problematic so we recommend using USB connections with laptops where possible. The simulator window within the https://makecode.microbit.org/ site can also be used to test the code independently of the Micro:bit devices.

## References


[1] S. Popat and L. Starkey, "Learning to code or coding to learn? A systematic review," *Computers & Education,* vol. 128, pp. 365-376, 2019. https://doi.org/10.1016/j.compedu.2018.10.005

[2] M. U. Bers, "Coding as another language: a pedagogical approach for teaching computer science in early childhood," *Journal of Computers in Education,* vol. 6, pp. 499-528, 2019. https://doi.org/10.1007/s40692-019-00147-3

[3] J. Miller, "STEM education in the primary years to support mathematical thinking: using coding to identify mathematical structures and patterns," *ZDM Mathematics Education,* vol. 51, pp. 915-927, 2019. https://doi.org/10.1007/s11858-019-01096-y

[4] P. J. Rich, S. F. Browning, M. Perkins, T. Shoop, E. Yoshikawa and O. M. Belikov, "Coding in K-8: International Trends in Teaching Elementary/Primary Computing," *TechTrends,* vol. 63, pp. 311-329, 2019. https://doi.org/10.1007/s11528-018-0295-4

[5] T. Q. Kieu, V. B. Nguyen and A. T. Nguyen, "Micro: bit in Science Education: A Systematic Review," *Jurnal Penelitian dan Pembelajaran IPA,* vol. 9, pp. 1-14, 2023. http://dx.doi.org/10.30870/jppi.v9i1.19491

[6] P. Maas and P. Heldens, *The Invent to Learn Guide to the micro:bit*, Constructing Modern Knowledge Press, 2023, pp. 2.

[7] R. B. Toma, "Measuring Acceptance of Block-Based Coding Environments," *Technology, Knowledge and Learning,* vol. 28, pp. 241–251, 2023. https://doi.org/10.1007/s10758-021-09562-x

[8] W. C. Choi and I. C. Choi, "Exploring the Impact of Code.org's Block-Based Coding Curriculum on Student Motivation in K-12 Education," in *2024 12th International Conference on Information and Education Technology (ICIET)*, Yamaguchi, Japan, 2024. pp. 93-97. doi: 10.1109/ICIET60671.2024.10542810.

[9] B. Jacobs, *Digital Scholarship in Education: Multimodality as a Window into Learning*, Taylor & Francis, 2024. doi: 10.4324/9781003449324-1

[10] I. Harel and S. Papert, *Constructionism: Research Reports and Essays, 1985-1990*, Ablex Publishing Corporation, 1991.

[11] Y. S. Lincoln and E. G. Guba, "The Only Generalization Is: There is No Generalization," in *Naturalist Inquiry*, Newbury Park, California, SAGE Publications, 1985, p. 110.

[12] J. Bruner, *The process of education*, Harvard University Press, 1960.

[13] T. Boz and M. Allexsaht-Snider, "How do elementary school teachers learn coding and robotics? A case study of mediations and conflicts," *Education and Information Technologies,* vol. 27, pp. 3935–3963, 2022. https://doi.org/10.1007/s10639-021-10736-4