A Leap for Mankind: Teaching Procedural Programming to Children

Matthew Bauer, Valentina Mosquera Reina, Ryan Cunico, Josiah Williams

Instructors: Dr. Cindy Robertson, Dr. Anca Doloc Mihu

Study Target / Focus / Purpose

This study addresses the ongoing need for more STEM professionals in the current workforce by encouraging a focused interest in information technology at an earlier age. This study specifically targets students and others that do not necessarily have a formal programming background.

Description of TAP program

The Technology Ambassadors Program is an interactive class focused on service learning. TAP provides the opportunity to build a project using different technologies and collectively deliver it through conferences, student involvement, workshops, and outreach events.

Methods

This study is inspired by the various simple racing games found on gaming websites. The project originally utilized a hand-tracking motion controller to control the direction, however this was removed in order to facilitate the workshop remotely during the COVID-19 pandemic.

The audience is introduced to procedural programming through an interactive activity where participants develop their own simple racing game using the Scratch programming language. Participants learn basic logical operators and statements, such as 'if' statements and 'while' loops.

The use of block code helps simplify coding concepts and makes them more accessible, creating more interest in STEM fields for students. Through the interactive drag-and-drop architecture of Scratch, students learn how certain algorithms are adopted to create the logic behind a game.

At the end of the activity, we hope to ignite a strong curiosity for the computer sciences and demonstrate that everybody is capable of learning how to program. We are encouraging the students to become lifelong STEM learners and begin their STEM journey at an earlier age.

The two technologies used in this project are the Scratch programming language and Leap Motion. Scratch is a drag-and-drop programming language developed by MIT, which aims to introduce children to programming and logical problem-solving in an interactive, easily digestible manner. Scratch also provides an online community where users can share interactive media such as stories, games, and animation with people from all over the world. Scratch is designed and maintained by the Lifelong Kindergarten group at the MIT Media Lab (Scratch, n.d.).

Leap Motion is a computer hardware sensor device that supports hand and finger motions as inputs. We use this technology to add functionality to the code and be able to move a car along a track by just moving your hand left and right without touching anything. We decided to remove the Leap Motion from our interactive activity in order to facilitate the workshop remotely during the COVID-19 pandemic.

Tap Expo
Target Audience: GGC college students and faculty
We will demonstrate the working game to Georgia Gwinnett students and faculty. Participants can visualize and understand the journey through video and presentation of our project via Zoom. Our project members will give a short description of the project to any curious event attendees, as well as answer any other technical questions that they may have.

Online Virtual Workshop
Target Audience: Middle School Students
Due to the COVID-19 pandemic, this workshop was facilitated online instead of in-person, utilizing the Zoom meeting platform. Participants were provided the required project files via a .zip file on Google Drive, as well as links to the ScratchX website.

This workshop began with an introduction to the finished game where participants could have fun with the racing car game and interact with the ScratchX interface. Students were able to play through the seven different courses with increasing levels of difficulty. The students were encouraged to share their final scores with each other to see who was able to navigate the courses the best.

Our group then led the audience of 10 middle school students through the interactive activity, which allowed them to develop their own simple racing game while incorporating their new knowledge of 'while' loops and 'if' statements.

Through this interactive workshop, participants demonstrated their ability to incorporate the learned programming concepts into their own program in a hands-on manner. During the workshop, these concepts were reiterated. The effects of those concepts became clear to students once they started manipulating their own code. After making a small change to the program's logic, students were encouraged to run the program to see how that change directly affected the game.

This provided immediate feedback, not only for the concepts they were learning, but also how a small change in the code's structure can have a large impact during gameplay. We were able to affirm that the participants were successfully educated and also thoroughly entertained by comparing the survey results that we collected before and after the workshop.

Throughout the workshop students had the opportunity to ask any questions or express any concerns. After the workshop, a number of parents of these middle school students contacted our adviser and expressed their content with the workshop, because their children had asked them if they could do the workshop again.

Online Virtual Classroom Workshop
Target Audience: Intro to Computing and Digital Media Students
Due to the COVID-19 pandemic, this workshop was unable to be held in person and was instead facilitated online, utilizing the Zoom meeting platform. Multiple workshops were held with groups of Georgia Gwinnett College students that are currently enrolled in Intro to Computing and Digital Media courses. The workshop was approximately one hour in duration.

This workshop began with a general introduction of our team, the TAP program, and our goals for the presentation. Participants were provided the required project files via a .zip file on Google Drive, as well as links to the ScratchX website.

Students were provided a pre-survey to measure their prior programming experience and familiarization with basic programming concepts, such as 'if' statements and 'while' loops. One of our team members demonstrated the working game and the students were then encouraged to play the game on their own devices. Students were able to play through the seven different courses with increasing levels of difficulty. The students were encouraged to share their final scores with each other to see who was able to navigate the courses the best.

Our group then walked the students through the development of their own racing game, focusing on the introduction of 'while' loops and 'if' statements, as well as how they are

used to create the logic that is needed for a racing game. Examples of this logic include detecting the finish line, sensing a car crash, or detecting a 'boost' and responding appropriately.

During the workshop, these concepts were reiterated. The effects of those concepts became much clearer once the students started manipulating their own code. After making a change to the program's logic, students were encouraged to run the program to see how that change directly affected the game. After the students coded and tested their own games, we summarized what they had learned. The students completed the post-activity survey and there was an open discussion, where students shared their thoughts and opinions regarding the workshop.

Individual Workshops

Target Audience: General

Due to the COVID-19 outbreak and stay-home orders, our group also conducted multiple individual workshops either at home or on Zoom.

These workshops allowed some participants to experience the racing game utilizing the Leap Motion controller, as originally intended. Pre and post surveys were provided to participants to measure the efficacy of the workshop.

During one of these individual workshops, the presenter experienced technical problems with the Leap Motion device, which caused her to continue the workshop in a similar format to the aforementioned workshops which were held online. This workshop still received favorable results, as the participant was able to successfully complete the activity and gained knowledge of the concepts being taught. This participant was also able to experience the game using the Leap Motion at a later date, and praised the easy and fun way to introduce programming concepts to people of all ages.

Challenges and Difficulties

The workshop had to be modified from its original format due to the unexpected COVID-19 outbreak and stay-at-home orders. By removing the Leap Motion from the presentation and incorporating keyboard controls, our team was able to quickly modify the existing instructions and program structure to conform to an online-only method of delivery, but there were various challenges that were encountered due to the online nature of the workshop.

Our team quickly learned how to troubleshoot common issues with the ScratchX platform, as well as with individual browsers and operating systems. The original workshop, intended to be in-person, would have relied upon a set of computers with the same operating systems, hardware specs, browsers and version, and browser

permissions. After moving the workshop online, the uncertainties and uniqueness of each participant's home computer environment quickly became apparent.

Some participants did not have Adobe Flash enabled in their browser by default, which was a large issue during the first online-only presentation. This led to the creation of step-by-step instructions for the students to combat this issue in the future. Students were also encouraged to utilize Google Chrome. During our first workshop we learned that students had been using various browsers, such as Safari, Internet Explorer, Microsoft Edge, or Mozilla Firefox. This led to compatibility issues with the ScratchX platform and introduced issues when troubleshooting any individual difficulties.

We learned that after changing the Adobe Flash permissions in Chrome, students would have to restart their browser. This was not a large issue when we hosted the workshop using Zoom, but when we used the browser-based BB Collaborate, students were forced to exit and rejoin the meeting room. We created a set of instructions for students to perform before the workshop, which included ensuring that the ScratchX website would load and that they could successfully retrieve the project files.

Some students found it difficult to download and extract the project files, which were in a .zip archive. We needed to become familiar with the process of extracting files from a .zip on both Mac and Windows, as well as where Google Chrome places the files downloaded from the internet on both operating systems. This allowed us to expedite our assistance to those students.

Since we were unable to see the participants' individual screens, we needed to become more familiar with the common errors that beginning Scratch users will make when trying to execute code. We adapted by becoming better at articulating exactly what the students needed to do and by slowing down to demonstrate it multiple times when necessary. Over time, we also became more familiar with common areas of confusion for the participants. This led us to preemptively address these pitfalls before they occurred, which led to a reduced number of issues.

We created a walkthrough document that we included in the project files, and this contained step-by-step instructions for the workshop, as well as screenshots and arrows to help clarify where each piece of code belonged. Students were able to reference this as their convenience and use it as a supplemental guide.

During the presentation, one of the presenters would act as the 'lead' presenter, sharing his/her screen and guiding the students through the walkthrough steps. The other presenter would continuously monitor the chat messages, helping to troubleshoot any individual issues, as well as alerting the lead presenter of any participant that needed further clarification on a concept.

The combination of these solutions ultimately led to a well-polished and smooth running workshop. Our team gained a wealth of knowledge that will continue to serve us during any future workshops that we host, both online or in-person.

Results

The results and success of the activities were measured by administering pre-activity and post-activity surveys. These surveys incorporated basic questions about the programming content and the individual's technological background, which let us gauge the participants' knowledge of basic programming concepts, both before and immediately after the activity.

We delivered our project to Intro to Computers, Digital Media, and middle school students using BB Collaborate and Zoom to adapt to the COVID-19 pandemic. Data was collected before and after the workshops via online surveys.

Our project invited all audiences to learn about technology and encouraged students to pursue a career in the STEM field. 90% of our participants were non-ITEC majors, which provided an opportunity to spark an interest in the STEM fields and introduce the material in a direct and simplified manner. Based upon the post-survey results, a large majority of participants found the workshop to be a well-structured initial introduction to these fundamental concepts of programming.
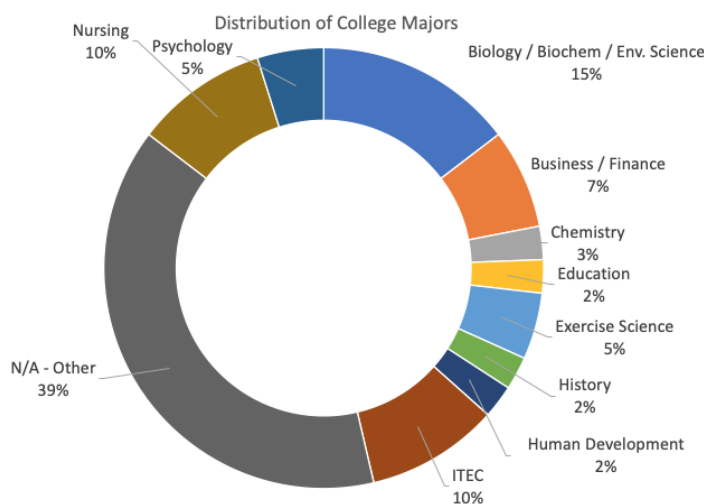

Fig 1. 90% of the workshop participants were not ITEC majors.

The ages and education levels of our participants varied greatly, which allowed our group to present the material to a wide variety of audiences. The participants' ages ranged from 12 to 63. A total of 27 participants, or 65.85%, were current college

students between the ages of 18 and 34. 10 participants, or 24.39% of respondents surveyed were current middle school students.
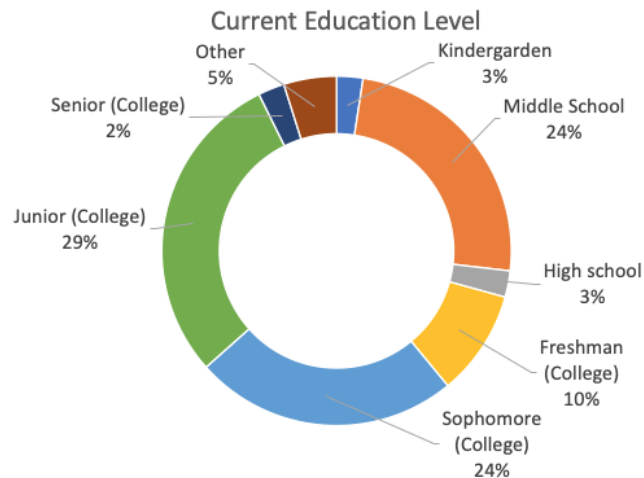

Fig 2. Breakdown of participants' current level of education

Out of the 41 participants, 29 students had no prior knowledge of 'if/else' statements and 32 had prior knowledge of 'while' loops. We were able to measure the remaining students who had limited or no prior knowledge of these concepts, to see if their understanding of these concepts improved as a result of the workshop. 100% of the participants with limited or no prior knowledge of 'if/else' statements saw an improvement, and 93.75% of participants with limited or no prior knowledge of 'while' loops saw an improvement.
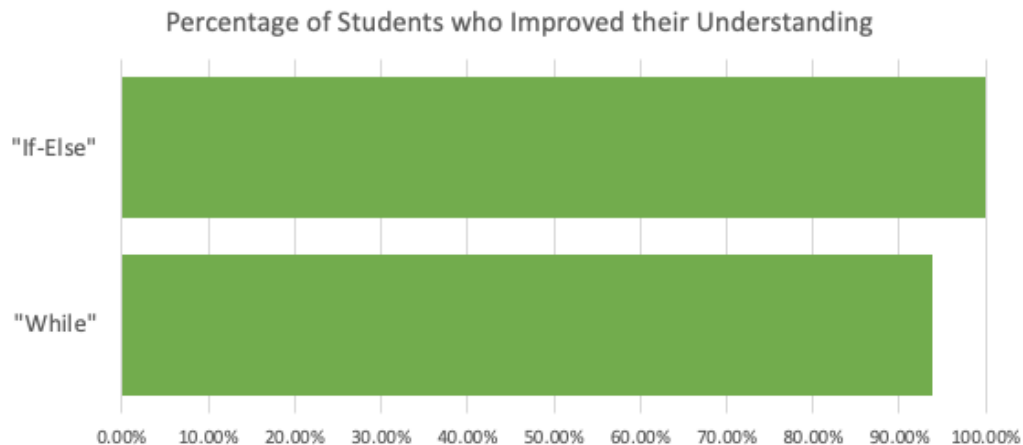

Fig 3. 100% and 93.75% of participants with limited or no prior knowledge improved their understanding of 'if-else' statements and 'while' loops, respectively.

Participants were asked about their understanding of "If" statements and "While" loops before and after the workshop. In both cases, the participants' understanding of these concepts improved dramatically after the workshop.

Participants were given relevant, simple-to-understand examples to explain these concepts. Students were given the following example of an 'if' statement: "If my hands are dirty, then I will wash them with soap and water". A similar example was given for 'while' loops: "While my hands are covered in soap and water, I will continue to scrub them". This helped introduce these concepts using a fun, real-world example that all participants could immediately relate to.
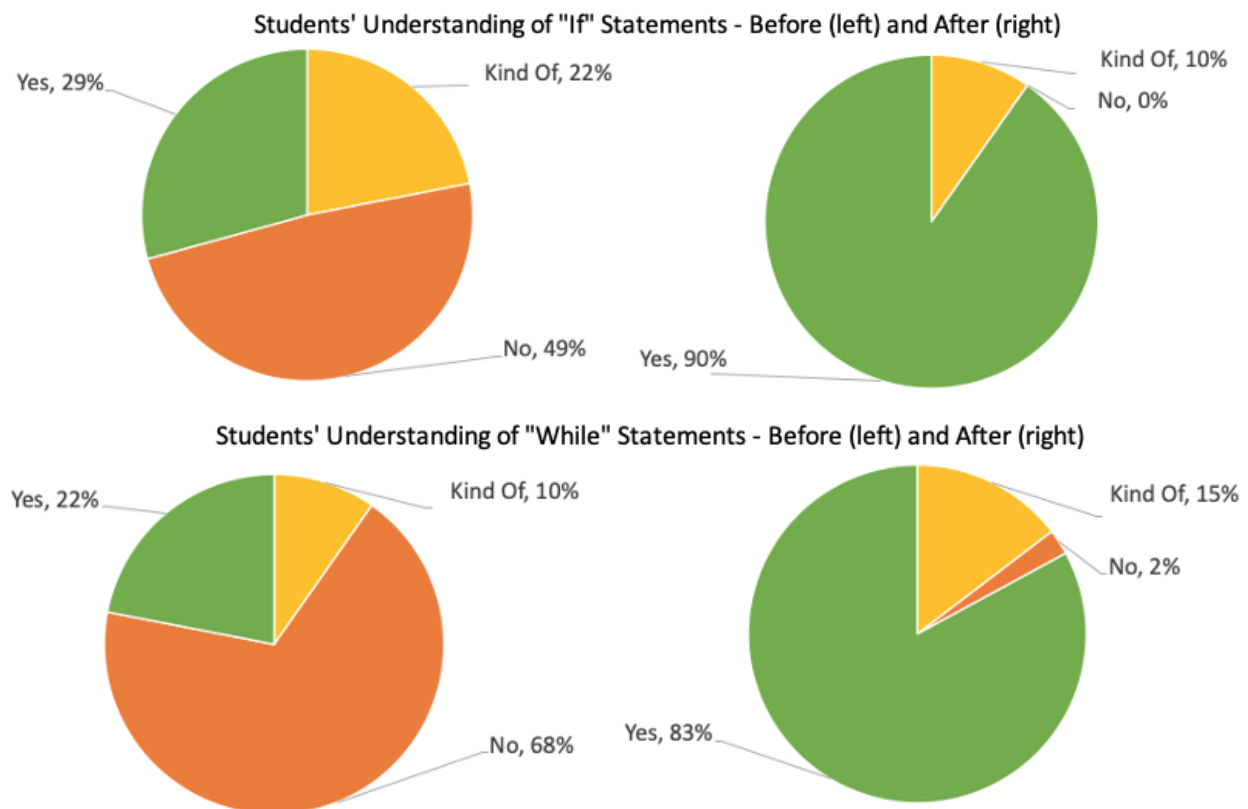

Fig 4. Out of the 41 participants, 12 had prior knowledge of 'If' statements and 9 had prior knowledge of 'While' loops.

Discussion and Conclusion

Our project's goals to introduce the general public to Scratch, teach and reinforce fundamental programming concepts to beginners, and to garner an increased interest in Information Technologies were all successful, based on the results of the pre and post surveys.

Our racing game showcases the importance of if-then statements and loops, taught in programming fundamental classes, without overwhelming the general audience with complex terminology. We used real-world examples to introduce the concepts, and

then allowed participants to write and modify code and see the immediate effects of those changes in the working code.

We have a continuous goal to motivate the GGC community, professionals, and various student groups to take part in science and technology activities and to learn more about the myriad of opportunities in STEM fields through our interactive and simple teaching techniques that introduce, engage, and excite the participants.

References

Scratch. (n.d.). Retrieved February 5, 2020, from https://scratch.mit.edu/

Ultraleap Tracking. (n.d.). Leap Motion Controller. Retrieved February 5, 2020, from https://www.ultraleap.com/product/leap-motion-controller/

Rosenbaum, E., Hanning, K., Dasgupta, S., Druga, S., Likhith, E., Aspeslagh, K., … Myriad Sensors. (n.d.). ScratchX. Retrieved February 5, 2020, from http://scratchx.org/

Scratch Extensions Browser Plugin. (n.d.). Retrieved February 5, 2020, from https://scratch.mit.edu/info/ext_download/

Hanning, K. (n.d.). Scratch Leap Motion Extension. Retrieved February 5, 2020, from http://khanning.github.io/scratch-leapmotion-extension/

Acknowledgements