Applying Scratch Programming to Facilitate Teaching in k-12 Classrooms

Dr. Afrin Naz, West Virginia University Institute of Technology

Dr. Afrin Naz is an assistant professor at the Computer Science and Information Systems department at West Virginia University Institute of Technology. She is working with high school teachers to inspire the K-12 students to the STEM fields. In last four years Dr. Naz and her team launched six workshops for high school teachers. Currently her team is training the high school teachers to offer online materials to supplement their face-to-face classroom.

Dr. Mingyu Lu, West Virginia University Institute of Technology

Mingyu Lu received the B.S. and M.S. degrees in electrical engineering from Tsinghua University, Beijing, China, in 1995 and 1997 respectively, and the Ph.D. degree in electrical engineering from the University of Illinois at Urbana-Champaign in 2002. From 1997 to 2002, he was a research assistant at the Department of Electrical and Computer Engineering in the University of Illinois at Urbana-Champaign. From 2002 to 2005, he was a postdoctoral research associate at the Electromagnetics Laboratory in the University of Illinois at Urbana-Champaign. He was an assistant professor with the Department of Electrical Engineering, the University of Texas at Arlington from 2005 to 2012. He joined the Department of Electrical and Computer Engineering, West Virginia University Institute of Technology in 2012, and he is currently an associate professor. His current research interests include wireless power transmission, radar systems, microwave remote sensing, antenna design, and computational electromagnetics. He was the recipient of the first prize award in the student paper competition of the IEEE International Antennas and Propagation Symposium, Boston, MA in 2001. He served as the chair of Antennas and Propagation Society of IEEE Fort Worth Chapter from 2006 to 2011.

Mr. Cody Ryan Zackoski, West Virginia University Institute of Technology

Currently, I am a second-year student enrolled at WVU Institute of Technology, studying for a B.S. in Computer Science and a B.S. in Information Systems. Before coming to WVU Institute of Technology, I graduated suma cum laude from both Midland Trail High School and from Fayette Institute of Technology in the Aries Computer Maintenance course. I taught a 12-week night class at Fayette Institute of technology on smartphones and tablets on three separate occasions. After coming to WVU Institute of Technology, I began working under Dr. Afrin Naz in a work-study arrangement. In this work-study, I am continuing research with parallel computing, and using the Scratch programming language as a tool for STEM education in the K-12 fields.

Mr. Caleb R Dingus, West Virginia University Institute of Technology

I am a student at West Virginia University Institute of Technology working towards a bachelor’s degree in Computer Science. I worked with Middle and High School teachers in developing curriculum and implementing Computer Science concepts.
Applying Scratch programming to Facilitate Teaching in K-12 classrooms
(Research-to-Practice, Strand: Other)

Introduction

This paper presents a project to apply Scratch programming in K-12 classroom. Scratch is a free educational programming language developed by Lifelong Kindergarten at Massachusetts Institute of Technology, oriented toward kids with age from 8 to 16 and/or with grade from third grade to high school\(^1\). Since Scratch employs a graphical programming interface rather than traditional code programming, it is simple and straightforward for kids as young as at the Kindergarten stage. Meanwhile, Scratch is highly versatile: if used effectively, it can facilitate teaching abstract concepts in any subject.

In this pilot project, we worked closely with a group of middle school and high school teachers with the aim of verifying the effectiveness of applying Scratch programming in K-12 classroom. This project consists of three phases. In Phase 1, we visited middle schools and high schools, helped the participating teachers to set up Scratch, and provided initial training of Scratch. Phase 2 of this project was entirely online, through which the participating teachers learned Scratch programming systematically. In Phase 3, middle school and high school teachers applied Scratch programming in their classes, and university students offered on-site and online support. Overall, such a “hybrid format” (that is, face-to-face plus online) is tailored for K-12 teachers without prior programming experience. After completing Phase 2, each teacher was expected to develop at least one Scratch project in his/her class. To date, nineteen middle school and high school teachers have applied Scratch programming in their classes, with subject areas spanning Math, Science, English, and Music. Each teacher assessed the performance of his/her students before and after applying Scratch programming. All the teachers univocally reported significant improvement of students’ scores after Scratch programming was applied. Based upon the data collected from the nineteen teachers, students’ average scores at least double as a result of Scratch programming. Most of the participating teachers commented that they are excited by applying Scratch in their teaching and they would recommend our training on Scratch programming to other teachers.

Related work

Scratch has been widely adopted across the nation to teach programming to young kids. Specifically, many universities have developed Scratch-based outreach programs, with a few examples articulated in the following. Through their SEEDS program for K-12 students, Wichita State University utilizes Scratch programming to promote STEM to young people\(^2\). The Young Women in Computing (YWiC) program at New Mexico State University integrates Scratch into
its curriculum of middle school summer camps in 2013. University of Texas in Dallas offers two Scratch camps (for beginner and advanced levels, respectively) in their K-12 outreach activities.

**Application of Scratch as a pedagogical tool**

Scratch was designed for youth from the ages of 8 to 16, but is used by people of all ages. With the visual nature of Scratch, it can parallel coding in a traditional programming language in a way that is generally more engaging to younger users and people without programming experience. Scratch possesses many of the features that are characteristic of more standard programming languages. It is possible to teach the most basic of computer science concepts—simple sequential instructions—up to Boolean logic, iteration, and even recursion by using Scratch.

However if Scratch were only a useful pedagogical tool for introducing older students to the concepts of computer programming, its utility would be limited. Scratch is not limited in its usefulness as a tool that can be used only for this purpose. Students are learning with Scratch at all levels (from elementary school to college) and across disciplines (such as math, computer

![Figure 1: Photos of Math and Science Scratch sample projects.](image-url)
science, language arts, social studies). Just as computers transcend nearly all facets of modern life, Scratch transcends many of the disciplines. It does this by giving its users the tools to create, recreate, or simulate virtually anything. Perhaps the best way to understand Scratch’s ability to be used as a pedagogical tool for multiple different K-12 subjects is through example.

The subject most closely related to computer science is math. Computer science is the application of mathematics in the same way that chemical engineering is the application of chemistry. This means that using Scratch as a pedagogical tool for exposing mathematical concepts comes more naturally than using Scratch for a language arts class, for example. Students can be taught geometry by having them make code in Scratch that will produce desired transformations and rotations on shapes. Scratch possesses code blocks that perform all the functions found on a scientific calculator. After learning the quadratic equation, students could write a program in scratch that will use the quadratic equation to calculate the roots of any quadratic polynomial. Using scratch’s pen function, code can be written to show the graphs of a function being drawn as its independent variable increases. Most mathematical principles can be expressed in some way with computers languages. With Scratch’s visible output, it becomes easy to adapt many mathematical principles to a project in Scratch.

With disciplines like language arts, social studies, or laboratory sciences, it may not be immediately apparent how Scratch can be utilized. One large part of Scratch is programming interactive stories, games, and animations. Coupled with the ability of users to create or use preexisting pictures in their projects, Scratch projects can contain anything that can be contained in a digital image file. By allowing users to create and upload their own content, the scope of Scratch becomes even wider to include the other disciplines. Scenes from classical plays can be recreated as simple animations in Scratch for a language arts project. Any diagram found in a text book can be made into a project by recreating the diagram in Scratch and writing a program to display more information as the user interacts with the points of interest. A map of historical battles, a political map of the world, a diagram of a human cell, and the syntax tree of an English sentence can all be made into interactive diagram projects with Scratch. Any concept that can be explained with an animation or diagram can be made into a project using Scratch. Music and the arts are just another example of a discipline that can be expressed in Scratch. It is possible to transpose any sheet music into Scratch. Visual arts, and graphic design are a large part of creating custom content to use in a project with Scratch. There are undoubtedly more subjects that can be supplemented by projects using Scratch.

Scratch is a tool that instructors in any setting can use to supplement the instruction of nearly any subject. It goes beyond its nature as a programming language by allowing its users to create their own custom projects based on any subject material. Students using Scratch will always be learning computer science principles to some degree, as all projects made in Scratch are a form of computer Program. Because of this, computer science education can be seamlessly integrated into classrooms everywhere.
Three-phase effort to apply Scratch programming in K-12 classroom

Total twenty six middle school and high school teachers from four counties were selected in this pilot project to apply Scratch programming in K-12 classroom. They teach diverse subject areas including English, Science, Math, and Music. None of the participating teachers has any prior programming experience.

This pilot project consists of three phases, as elaborated below separately.

During Phase 1, we visited middle schools and high schools. Our university students helped the participating teachers to set up Scratch over their computers and provided them initial training of Scratch. Particularly through interacting with the teachers, personal relationships were established between us and the teachers, which greatly facilitated the next two phases of this project. The photo in Figure 2 was taken when we visited one of the high schools.

Phase 2 was conducted entirely online. Specifically, the participating teachers learned Scratch programming through our online course named “Introduction to Scratch.” The online course consists of four units. Each unit further consists of four video tutorials. In each video tutorial, PowerPoint slides, programming demonstrations, and exercises together with solutions are offered to the teachers. The online course was delivered using Google Course Builder, which is a free software. Figure 3 shows a snapshot of one of the video tutorials.

In Phase 3, middle school and high school teachers applied Scratch programming in their classes. After completing the online course in Phase 2, each teacher was expected to develop at least one...
Scratch project in his/her class. Our university students offered them on-site and online support. The photo in Figure 4 was taken when one university student was helping a teacher with her project. According to teachers’ feedback, online support is especially helpful for schools in geographically-isolated regions.

Figure 4: A photo taken in Phase 3 of this project.

**Results**

The impact of the project on teachers and students is determined through a series of surveys and interviews. After attending our Scratch training, teachers were expected to develop at least one Scratch project in their own discipline area for his/her classroom. We also have collected pre and post raw scores from student assessments before and after administering a scratch project. Overall, the results indicate that, the project succeeded in instructing the basic concepts of Scratch programming to the participating teachers and increasing their confidence of incorporating Scratch based learning into their own subject areas. Results of the interviews, and student performance along with some projects created by the teachers are presented in this section.

1. **Sample Scratch projects developed by the participating teachers**

As mentioned before, after attending our Scratch training, teachers were expected to develop at least one Scratch projects in their own discipline area to his/her classroom. Our university students provided onsite and online help to the teachers if needed. To date, nineteen middle school and high school teachers have applied Scratch programming in their classes, with subject areas spanning Math, Science, English, and Music. In the followings we are describing some sample Scratch projects created by participating teachers. Some of their efforts are demonstrated in Figure 5.

**Sample Scratch project developed for Science class:**

The Balancing Equations Project allows the user to test their knowledge of balancing equations with a fun and interactive game to save the school by choosing the correct answers to these chemistry problems. There are two levels with increasing difficulty to master the subject.
Sample Scratch project developed for Math class:
The Math Graphing Project allows the user to learn about reflection, rotation, and transcribing points using a square with four points on the corners. Pressing the buttons shows how each point is moved around the graph.

Sample Scratch project developed for English class:
The Vocab Project is a program that allows the user to take a vocab quiz to test their knowledge of words taught in class. The program will tell the user if they got the answer right or not and explain what words they need to review the definitions.

Sample Scratch project developed for Music class:
The Music Intro Project is a fun and simple program to introduce the rules and expectations of a music/digital arts class. It uses animations and music to explain the syllabus and some of the guidelines for the class.

Figure 5: Samples of K-12 teachers’ efforts in applying Scratch programming in their teaching.
After developing their own projects in certain topic, the teachers first introduced the topic in his/her classroom and assessed the students by providing a test or quiz. After that the teacher introduced the student to his/her developed program and revisited the same topic. In some classrooms the students also developed their Scratch programs. Finally the teacher provided the same test or quiz to collect the post score. By comparing the pretest and posttest score, the teacher identified if there is any improvement in student achievement as a result of the Scratch programming based learning.

2. Pre and post assessment scores

As mentioned before, nineteen teachers delivered the engineering projects in their middle or high school classrooms. The teachers who incorporated the Scratch programming in their classrooms assessed their students before and after Scratch programming was applied. All of the teachers noted an improvement in the students’ assessment scores when comparing the pretest to the post test scores. On average about 45% improvement in student learning score has been reported. This increase indicates that the increase in student achievement is a result of the Scratch programming based learning.

3. Teachers comments from interviews

All the participating teachers took surveys before the online course, after the online course, and after delivering Scratch programming. Survey results indicate that most of them are excited by applying Scratch in their teaching. Several comments from the teachers are quoted below.

- *I had never used computer programming, and so this was the first time I’d ever used anything to do with computer programming in a science classroom, and I think what I saw was the students were extremely excited about it. It was a concept that I had not previously used or maybe they had not previously used but had heard of, and when I said computer—we’re going to use some computer programming, extreme excitement just, “What are we going to do with it? What—” I just have to say, “You have to just calm down and let me tell you what we’re going to do with it,” so definitely that’s what I saw affect the student learning.*

- *Students are more engaged and interested. They want to problem-solve, and are more pragmatic in their attack of the problems. They’re very academically competitive…and they wanted to know more just so they could demonstrate their knowledge and show other students what they had learned on their own before they came to class.*

- *I feel like my groups as a whole showed more interest in STEM, and they liked seeing the application of their knowledge to specific careers. As a whole, I noticed that more introverted students had a voice, and they were more involved in the project than they had been in the past.*
Also as indicated by the surveys, 89% of the participating teachers would recommend our training on Scratch programming to other teachers, the remaining 11% were neutral, and none of the participants selected “do not recommend.”

Conclusions

A pilot project is reported in this paper, in which we worked closely with a group of middle school and high school teachers with the aim of verifying the effectiveness of applying Scratch programming in K-12 classroom. This project employs a “hybrid format,” including both face-to-face and online, which is tailored for K-12 teachers without prior programming experience. To date, nineteen middle school and high school teachers have applied Scratch programming in their classes, with subject areas spanning Math, Science, English, and Music. All the teachers univocally observed significant improvement of students’ scores after Scratch programming was applied. Most of the participating teachers commented that they are excited by applying Scratch in their teaching and they would recommend our training on Scratch programming to other teachers. As this project is still ongoing and we are anticipating data from more teachers, we will present the outcomes comprehensively at the conference.

According to our survey, all participating teachers are very excited about this project and about 89% of the participating teachers would recommend our training on Scratch programming to other teachers. We already have contacted the superintendents of three school districts in West Virginia. Looking at our initial data they can see the prospect of our Scratch programming based learning and they all agreed to allow us to work with their teachers. In our 2017 effort we decided to recruit at least two teachers from the same discipline at same school. We are hoping that will allow the teachers to support each other during the school year. We also decided to set up an online forum so all teachers can post their experiences to learn from each other.

Acknowledgment

The authors would like to thank Google Inc. for providing the financial sponsorship.

References

1. www.scratch.mit.edu
2. www.webs.wichita.edu/?u=seeds&p=/main/aboutseeds/
3. www.ywic.nmsu.edu/curriculum-2/scratch/
4. www.utdallas.edu/k12/desc/