Out-of-school Time STEM: Teach Programming Using Python for High School Girls

Chaoyi Wang, Srikanth Vemula and Dr. Michael Frye
chwang2wang@gmail.com, vemula@uiwtx.edu, mfrye@uiwtx.edu

Abstract – **In the next decade, there is an enormous increase in job openings in the fields of science, technology, engineering, and mathematics (STEM). The early recognition of STEM talent is necessary to meet the demands of STEM labor force in the United States. Thus, it is essential for educators to apply diverse teaching methods to provide meaningful programming learning to students at High School level. In this study, the researchers designed an eight-session Python programming curriculum for high school girls and implemented in Girls in Engineering, Mathematics, Science (GEMS) STEAM program in San Antonio, Texas, USA. Through the analysis of pre- and post-surveys and interviews, the results showed that the Python programming course have created a fun and interesting learning environment. This eight-session course effectively expanded students' previous knowledge about programming and increase their interests in computer science (CS). In the process of learning, students developed the problem-solving skills. This study suggested that it is important for educators to create a fun and interactive learning environment when teaching programming for high school girls. There is a need of more efforts and opportunities which needs to be provided for girls to increase their participation in CS.**

Index Terms - Computer Science, High school girls, Python programming course, STEM out-of-school program

INTRODUCTION

Researchers have consistently shown that enhancing STEM workforce has been a priority for the United States to compete in the global economy [1, 2, 3]. According to the labor statistics, most of the STEM occupations are professional jobs in the fields of CS, mathematics, engineering, and technology [4]. From 2014 to 2024, computer occupations, engineers, and mathematical science occupations are growing the fastest with the highest job gains among all types of STEM occupations [5]. In the future, we will see many automated systems that are going to be used to perform various tasks. To fit in the future society, students need to be adaptable to modern and future technology [6]. The early recognition of STEM talent is necessary to meet the demands of STEM labor force in the United States [7, 15].

The interest in improving STEM learning and participation has led to the creation and growth of numerous out-of-school (OST) STEM programs that aim to increase the number of young students seeking a college degree and eventually enter STEM fields [3, 17]. During summer holidays (June to August), many STEM enrichment programs provided STEM learning opportunities and hands-on experiences for students at the elementary, secondary, and post-secondary levels. Although each program has different curricula representations, robotics and programming have become increasingly popular and have been identified as innovative pathways in integrating different subjects in OST STEM programs [8]. Affordable LEGO Mindstorms robots were widely introduced, and programming training was integrated into these programs as well. In south Texas, existing OST programs are using various programming platforms to teach from text-based and block-based coding to formal programming languages by integrating animation, game design, and robotic.

Although many public and private organizations offered informal and formal programming learning, a significant issue remains: how do educators teach programming in OST settings? From a technical point of view, three main aspects that students need to learn, including data, instructions, and syntax [14]. These three things cover the concepts of variables and data types, control structures, procedures, and the vocabulary of the language. However, in the process of teaching programming, the difficulties may come from teaching perspectives, concerning aspects such as motivation and technical issues. To address these challenges, teaching programming should follow an effective teaching sequence, which offers students a simple language, and choose various problems to solve [6]. Thus, there is a need for more educators to apply diverse teaching methods to provide meaningful programming learning to students at high school level.

STATEMENT OF PURPOSE

The purpose of the study was to create a programming course by using Python to examine high school girls' attitudes toward programming and whether the course had a positive influence on students' learning. To achieve these goals, the researchers designed an eight-session Python programming curriculum and implemented it in an OST STEM summer program based in San Antonio, Texas, USA, from June to August 2019. Through the analysis of pre- and post- surveys from the high school girls in the summer STEM camp, the research examined how the programming course influenced students' attitudes towards CS. The

978-1-7281-7520-1/20/$31.00 ©2020 IEEE

analysis of interviews explored high school girls' perspectives on programming and helped in explaining the survey results. In this paper we attempted to find answer these following research questions 1) what are high school girls' perspectives on programming? And 2) what are the influences of the OST STEM summer program on high school girls' attitudes towards CS?

METHODOLOGY

I. Overview of GEMS STEAM Program

This study implemented an eight-session Python programming course in GEMS STEAM program in San Antonio, Texas, USA. GEMS was founded at the University of the Incarnate Word (UIW) since 2015 with the mission of introducing more females into STEM fields [15]. In summer 2019, GEMS hosted two times two-week summer camps for high school girls in rising grades nine. The programming course was nested within the GEMS high school summer programs. Along with learning programming, GEMS high school summer program also covered other STEM curriculum and learning activities, including robotics, game design, nutrition, and engineering [16, 17].

II. Participants

A total of 48 girls participated in the GEMS two-week high school summer camps, but only 27 students were able to finish both pre- and post- surveys, and 39 girls were interviewed on the last day of the camp. Among those girls who participated in the GEMS, the average age was 13.95. Their ethnic backgrounds consisted of White (2%), Hispanic (42.9%), African American (28.6%), Asian (9.5%), and other (9.5%). Before the camp started, informed consent was collected from all the students and their parents. Most of the participants in this study had STEM learning experiences where they learned basic programming concepts such as block coding and graphic design. Only a few of girls learned Java, HTML, and Scratch in STEM academies.

III. Curriculum

This study utilized project-based learning (PBL) as teaching pedagogy, as many studies demonstrated that PBL had positive influences on students' learning. For example, research showed showed students expressed higher enthusiasm to explore STEM concepts with real-world solutions and presented higher self-efficacy in STEM fields [9,10]. PBL is also beneficial to offer students more hands-on opportunities and help students connect the contents with real-world situations.

Python was selected to introduce fundamental programming concepts for high school girls in this study after a thorough survey with the popularity and practicality from a variety of programming languages (e.g., C, C++, MATLAB, Java). Python consistently ranked among the most popular programming languages based on the rankings of most popular languages. Furthermore, Python supports several different programming paradigms. Students could achieve easy and complex tasks by using Python, whereas Java does not allow students to start with simple expressions and statements. Moreover, Python has been applied in various high schools, colleges, and top-ranking universities from around the world in their curriculum. For the real-life application, Python has diversified applications in various emerging industries like gaming, web frameworks and applications, language development, prototyping, graphic design applications, and Artificial Intelligence.

The researchers designed five gamed-based projects. Each project covers the fundamental concepts in programming and uses the Python IDE environment to practice those concepts. Students practiced in a computer lab at UIW, where everyone has a computer to work on in every class. The teachers presented concepts first and performed different syntaxes to achieve goals. Students not only received help from the teachers but also learned from other girls in their groups. The teachers checked if students were working on it correctly and addressed any misconceptions to the class. The following is the description of each project.

Project I: This project aimed to introduce the concepts of integers, floats, strings, Boolean, and turtle graphics in Python. Students started to learn basic graphics components in Python. Once they got familiar with the concepts, they were asked to draw the simplified version of the GEMS camp logo. By completing this game, students would be able to explore various graphical elements in Python, especially learning turtle graphics to draw various shapes (Figure 1).

FIGURE I
PROJECT I: "CAMP LOGO."

Project II: This project was an upgraded version of project I. Students applied the concepts and elements that they learned from the previous project to write a new program individually. The purpose of this project was to move the turtle and design a T-shirt by using text-based programming (Figure 2).

Project III. This project was designed to teach the concept of random number generation. Students learned the concepts of arrays and loops first. Then, they practiced how to create an algorithm or pseudo code. The purpose of this project was to design a pseudo code by using a random number generation function (Figure 3).

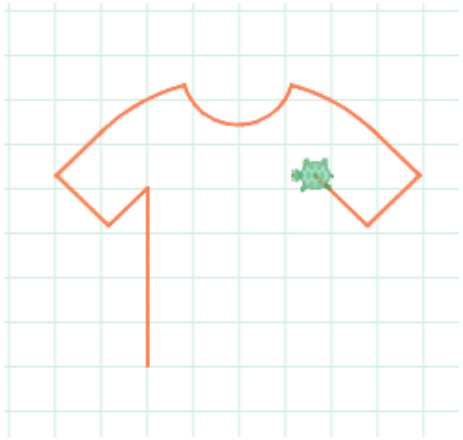
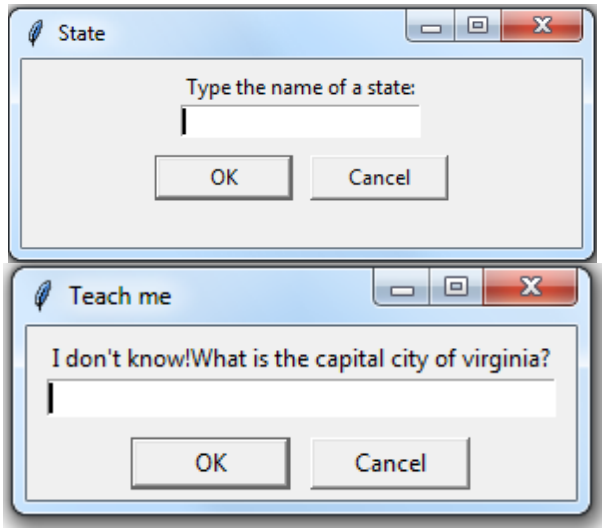


FIGURE 2
PROJECT II: "DESIGN YOUR TEE"

```
Please enter your guess: 56
Your guess is too low 155
>>>
======= RESTART: C:\Users\Srikanth\Desktop\pygames\guessthe number.py =======
Guess my number between 1 & 201: 55
Your guess is too low and thre number is 179
>>>
======= RESTART: C:\Users\Srikanth\Desktop\pygames\guessthe number.py =======
Guess my number between 1 & 201: 200
Your guess is too high and the number is 75
>>>
======= RESTART: C:\Users\Srikanth\Desktop\pygames\guessthe number.py =======
Guess my number between 1 & 201: 88
Your guess is too high and the number is 47
```

FIGURE 3
PROJECT III: "GUESS IT!"

Project IV: This project aimed to teach students how to create, read, and write content on to a text file. It also had the purpose of demonstrating how to display windows concepts in Python. By completing the project, students could able to create a quiz game with their subject of interest and learn the contents of those subjects by playing the program. For instance, we made a game of learning the Capitals of each State in the U.S. Then, students made a quiz game to learn how to create functions in Python (Figure 4).



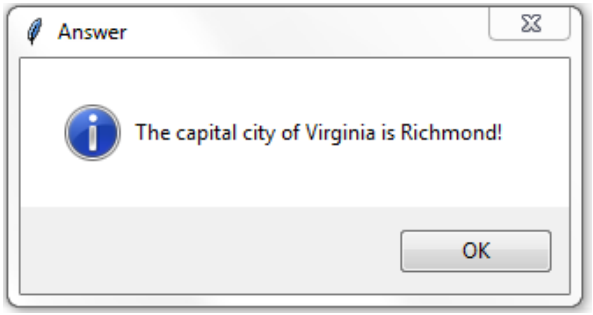


FIGURE 4
PROJECT IV: "BUILD A QUIZ."

Project V: This project was designed to combine all the elements that were taught in the above projects. Students could able to have a start button, restart button, score, timer, and adding player character functionalities to the snake game (Figure 5).

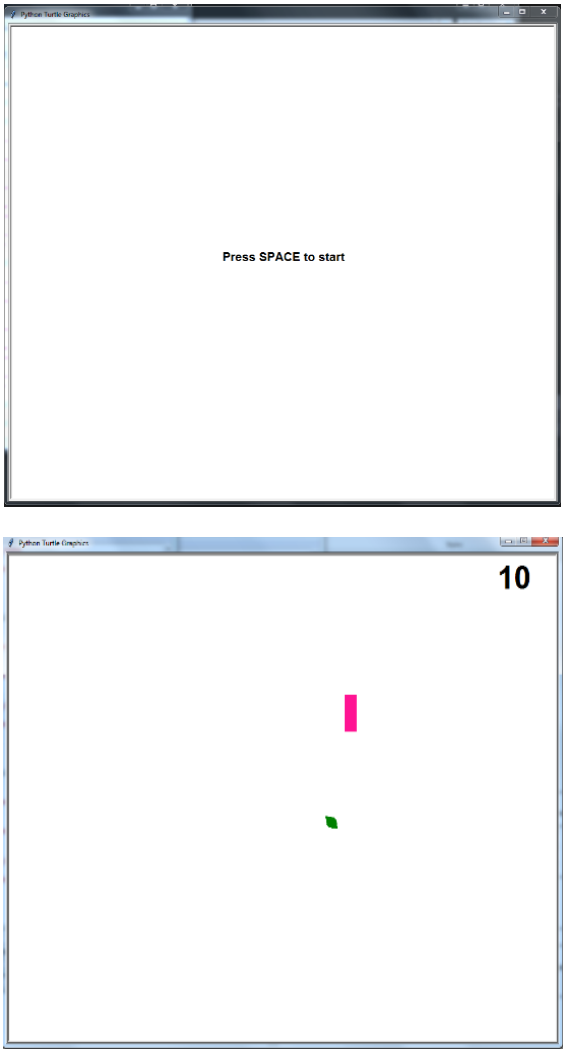


FIGURE 5
PROJECT V: "PLAY WITH SNAKE."

IV. Measures: Interview and Survey

In this study, interview and survey were used as measures to explore participants' perspectives on programming and examine the impact of Python programming course on participants' attitudes towards programming. The researchers modified the STELAR's computational thinking survey into a seven-item survey [11]. The pre-test and post-test were conducted on the first day and the last day of the program. The survey questions consisted of Likert-scale (1= "strong disagree", 2= "disagree", 3= "neutral", 4= "agree", 5= "strongly agree"). A software package, SPSS 24, was used for statistical analysis.

The factor analysis results showed an adequate fit of the mode for pre- test (x^2 (21) = 89.91, p <.001) and post- test (x^2 (21) = 70.80, p <.001). As shown in Table 1, most of the factor loading were large (>.60). The calculated Cronbach's alpha coefficients for each question were all exceeding the 0.60. Taken together, these results suggested the adequate reliability and validity of the survey.

TABLE I
RESULTS OF FACTOR ANALYSIS AND RELIABILITY TEST

	Factor Loading		Alpha	
	pre	post	pre	post
Computer Science Attitudes			.86	.84
Q1. Knowledge of computing will allow me to do a better job	.76	.77		
Q2. I can solve problems by using computer applications	.48	.49		
Q3. I expect to use computer applications in my future work	.81	.80		
Q4. I think that programming helps computer problem solving.	.72	.73		
Q5. I expect to use computer applications for future classes	.81	.68		
Q6. I hope that my future career will require the use of computing concepts	.92	.82		
Q7. I think that computer science and problem solving are interesting	.75	.73		

Additionally, on the last day of the second week, students participated in semi-structured interviews lasting approximately 10 minutes. The purpose was to explore students' programming learning experiences in the camp and how the learning experience influenced their attitudes toward programming. Table 2 presents the interview questions.

TABLE 2
INTERVIEW PROTOCOL

End of 1st Week	Questions
1.	What is programming in your opinion?
2.	Did the camp give you more ideas about programming and computing? Please give examples.
3.	Give an example of future careers you are interested in.
End of 2nd Week	Questions
	What do you think of programming? Is there anything that you learned at camp that may apply to your school learning and future careers? Give an example of future careers you are more interested in after experiencing GEMS?

RESULTS

I. Survey Results

To determine if the programming learning was helpful, descriptive analysis was used. As shown in table 3, the mean score for most of the participants on the posttest was higher than that for the same participant on the pretest, except for questions 3 and 5. The mean differences between the pre- and post-survey results are shown in Table 3 ranging from -.44 to .07 and the standard deviation differing from .75 to 1.07. The great mean different was question 7 and the smallest mean different was question 3.

To answer the question of the influences of the STEM summer program on high school girls' attitudes towards CS, paired samples *t*-tests were used. Table 3 presented a significant increase in the question of "I think that computer science and problem solving are interesting" (t(26)=-2.37, p<.05). The significant increase showed students had higher interests in CS and problem solving. However, there were no statistically significant differences in other questions.

TABLE 3
RESULTS OF PAIRED SAMPLES T-TESTS

	Mean		Mean Difference	SD Difference	t	df
	pre	post				
PairQ1	3.74	3.85	-.11	.75	-.77	26
PairQ2	3.40	3.52	-.12	.88	-.68	24
PairQ3	3.44	3.41	.03	.90	.21	26
PairQ4	3.78	4.07	-.29	.87	-1.77	26
PairQ5	3.70	3.63	.07	1.07	.36	26
PairQ6	2.64	2.80	-.16	.85	-.94	24
*PairQ7	3.00	3.44	-.44	.97	-2.37	26

Note. * p<.05 (two-tailed test).

II. Interview Results

To explore high school girls' perspectives of programming, the researchers analyzed the interview data by using comparative analysis [12]. The researcher coded line by line and sorted the codes into three emerging themes, including 1) students' perspectives towards programming, 2) evaluation of the programming course, and 3) influences of the programming course on students' learning. When reporting the findings, the terms *most, majority, the participants, the girls, generally or usually* were used when referring to emerging findings that were common for at least 10 majority of the sample. The terms *some and several* referred to response's characteristic for 4-6 participants. The term *a few* was used when the responses were common for 3 or fewer participants.

Students' perspectives towards programming and careers in CS. Most participants mentioned that programming is useful and has different applications in the future. Girls mentioned that the world was developing

through technology. Thus, programming was more than making a simple game and creating a basic program for fun. Instead, programming languages guided computers to complete tasks, which could have great impacts on people's lives. The advanced technology could make the world better and more efficient. For example, a girl expressed, "*the purpose of programming is to solve problems, there are different ways, like using computers and using your minds to work on."*

Some participants expressed their perspectives on careers in CS. For those who shared thoughts about future careers, they did not believe that being a programmer or getting a job in CS was suitable for them. They expressed that essential programming skills could be a gateway for them to get into the CS field. However, they indicated they were not confident in their abilities and needed to improve their programming skills. Although learning programming was interesting, it was not related to their future pathways. Computer programming could be a hobby or an alternative career choice for them. For instance, one girl mentioned, "*I am more interested in if it is more to the medical field. My family have more people in the medical field. medical shows is so popular."*

Evaluation of the programming course. The interview results showed the programming course created a fun and interesting learning environment and effectively expanded students' previous knowledge about programming. One participant mentioned, "*I never liked the programming. But I am really interested in Python. It is fun to me."* It is important to note that most participants in this study had previous programming experiences. They came to the camp with a high interest in STEM fields, especially in technology and engineering. Under this condition, many participants still expressed that they were gaining more interest in programming. After the two-week camp, most of the participants had a better understanding of Python. For example, a girl stated, *"I had some previous programming experience and now it [GEMS program] has expanded my knowledge of programming."* Similarly, a girl expressed, *"I learned the game before, but I do not know there are so many types of games. I learned programming, but I do not know, this could be so fun."*

Although many participants had programming experience, they still expressed that learning Python programming was hard and more complicated than they thought before. Two girls revealed programming was boring and repetitive. For example, one girl stated, "*before I did not have any patience, but this camp has helped me learn to be patient with programming."* However, most of girls believed the learning process was struggling but rewarding. They became more familiar with syntaxes after practicing several times. One participant mentioned, "*I can see how much time and work and focus I put on the coding. I can see the results."*

Influences of the programming course on students' learning. Many participants revealed that programming required a lot of creativity. In the process of learning it, students developed their problem-solving skills. One girl stated, "*Programming helps use problem solving and be more efficient in creating new things."*

The first aspect of creativity for participants was figuring out how to use code to solve tasks. Teachers in the programming class usually presented different ways to code. Students compared each way and picked the optimal codes. The second aspect of creativity was students had freedom to choose different items to create their own game. For example, one girl mentioned, she felt cool about customizing the patterns, contexts, colors in the games. The third aspect of creativity was being able to write code and apply in the future. As students expressed, the basic concepts and functions (e.g., loop, while) behind each game were useful to apply them in other scenarios.

DISCUSSION

The findings of this study confirmed that the previous research on teaching concepts through technology can motivate students to become excited because it is a fun way for them to engage in STEM lessons [18,19]. Providing hands-on programming activities for students at the K-12 levels has the benefits of developing problem-solving skills, which facilitated them to transfer the knowledge to different disciplines [6]. Thus, this study suggested when teaching programming in OST settings, it is important to motivate students by integrating modern technology and making the learning environment more fun.

This study revealed high school girls' perspectives on programming and CS-related careers. High school girls believed programming would be useful in the future, but the learning process was struggling. For a few of girls, learning programming was boring. Being a programmer or getting a career in CS was not suitable for several participants. To successfully achieve a career in CS field, participants revealed they needed to get more practical opportunities and spend more time on programming. This finding is similar to some previous studies about children programmers. When girls and boys have similar experience and interests in computer programming, few girls choose to program [13]. One the other hand, the ability to develop programming skills are related to the prior programming experience and the amount of time students spent on programming. From these, one of the key elements in increasing the participation of high school girls in CS may be by providing more learning opportunities and connecting the practice with real-world applications.

LIMITATIONS

There are limitations to the study we present: one is the duration of the Python programming course and the other one is length of interview. Although our results are encouraging, high school girls in both camps only spent about eight hours learning and practicing Python. A longer course and more projects might generate different results.

Furthermore, although we conducted 39 interviews with high school girls who participated in the GEMS summer

program, each interview only lasted 10 minutes. Because of the tight schedule in the summer program, we could not have chance to conduct in-depth interview with participants. A longer conversation might generate more ideas about the reasons why girls did not like programming and careers in CS.

FUTURE WORK

We will continue to develop Python programming courses and implement more hands-on projects for high school girls. We would like to focus more on 1) developing and evaluating Python programming curriculum, 2) findings ways to motivate girls to participate in programming, and 3) exploring high school girls' career pathways into CS.

CONCLUSION

The Python programming hands-on learning experience in a summer program had a positive influence on developing high school girls' problem-solving skills and increasing students' interests in CS. This overall positive result suggests it is important for educators to teach programming in a playful environment. Considering the gender differences in learning programming, the researchers suggest that it is necessary to provide more learning opportunities for girls align with the meaningful learning process.

ACKNOWLEDGEMENT

This research was partially supported by the San Antonio Area Foundation Grant 201752177, Rackspace Award 23933, and the UIW Autonomous Vehicle Systems Laboratory. The author would like to give special thanks to all GEMS staffs for their invaluable support during the research.

REFERENCES

[1] Marginson S., Tytler R., Freeman B., & Roberts K., "STEM: Country comparisons: International comparisons of science, technology, engineering and mathematics (STEM) education, ". Final report, 2013.

[2] White House. "A strategy for American innovation: Securing our economic growth and prosperity, ". Washington, DC: Executive Office of the President, 2011.

[3] Saw, G. K., Swagerty, B., Brewington, S., Chang, C. N., & Culbertson, R. (2019). Out-of-School Time STEM Program: Students' Attitudes toward and Career Interests in Mathematics and Science. International Journal of Evaluation and Research in Education, 8(2), 356-362.

[4] Christensen R., Knezek G., Tyler-Wood T, & Gibson D. (2014). Longitudinal analysis of cognitive constructs fostered by STEM activities for middle school students. Knowledge Management & E-Learning: An International Journal, 6(2), 103-122.

[5] Fayer S., Lacey A., & Watson A., (2017). STEM Occupations: Past, Present, And Future. U.S. Bureau of Labor Statistic.

[6] Saeli, M., Perrenet, J., Jochems, W. M., & Zwaneveld, B. (2011). Teaching programming in Secondary school: A pedagogical content knowledge perspective. Informatics in education, 10(1), 73-88

[7] Young, J., & Young, J. (2018). The structural relationship between out-of-school time enrichment and black student participation in advanced science. Journal for the Education of the Gifted, 41(1), 43-59.

[8] Hinton, T. H. (2017). An exploratory study of a robotics educational platform on stem career interests in middle school students (Doctoral dissertation, University of Alabama Libraries).

[9] Kwon, H. (2017). Effects of 3D Printing and Design Software on Students' Overall Performance. Journal of STEM Education: Innovations and Research, 18(4), 37-42.

[10] Ogle, J. P., Hyllegard, K. H., Rambo-Hernandez, K., & Park, J. (2017). Building Middle School Girls' Self-Efficacy, Knowledge, and Interest in Math and Science Through the Integration of Fashion and STEM. Journal of Family & Consumer Sciences, 109(4), 3340.

[11] Yadav, A., Zhou, N., Mayfield, C., Hambrusch, S., & Korb, J. T. (2011, March). Introducing computational thinking in education courses. In Proceedings of the 42nd ACM technical symposium on Computer science education (pp. 465-470).

[12] Charmaz, K., & Belgrave, L. (2012). Qualitative interviewing and grounded theory analysis. The SAGE handbook of interview research: The complexity of the craft, 2, 347-365.

[13] Harel, I. Children Designers. Ablex Publishing Norwood, N.J., 1991.

[14] Govender, I. (2006). Learning to Program, Learning to Teach Programming: Pre- and In-service Teachers' Experiences of an Object-oriented Language. University of South Africa.

[15] Wang, C., & Frye, M. (2019, June). miniGEMS 2018: A Mixed Methods Study Exploring the Impact of a STEAM and Programming Camp on Middle School Girls' STEM Attitudes. In 2019 ASEE Annual Conference & Exposition.

[16] Wang, C., & Frye, M. (2019, March). miniGEMS 2018 Summer Camp Evaluation: Empowering Middle School Girls in STEAM. In 2019 IEEE Integrated STEM Education Conference (ISEC) (pp. 149-155). IEEE.

[17] Wang, C., & Frye, M. (2019, March). Measuring the Influences of a STEM Enrichment Program on Middle School Girls' Self-efficacy and Career Development. In 2019 IEEE Integrated STEM Education Conference (ISEC) (pp. 165-168). IEEE.

[18] Hollenbeck, R., & Fey, J. (2009). Technology and mathe- matics in the middle grades, Mathematics Teaching in the Middle School, 14, 430-435.

[19] Jones, K. (2000). Proving a foundation for deductive reasoning: Students' interpretations when using dynamic geometry software and their evolving mathematical explanations. Educational Studies in Mathematics, 4, 55-85.

AUTHOR INFORMATION

Chaoyi Wang, Ph.D. candidate, Dreeben School of Education, University of the Incarnate Word.

Srikanth Vemula, Adjunct Professor and PhD Student in Management Information systems, HEB School of Business, University of the Incarnate Word.

Dr. Michael Frye, Associate Professor, Department of Engineering, University of the Incarnate Word