International Journal of Scientific Advances
ISSN: 2708-7972
Volume: 1 | Issue: 3 | Nov - Dec 2020 Available Online: www.ijscia.com

Children Perceptions of the Effectiveness of Online Coding as a Supplement to in-person Boot Camps

Akinode John Lekan[1]* and Oloruntoba Samson Abiodun[1]

[1]Computer Science Department, School of Pure and Applied Science, The Federal Polytechnic Ilaro, Nigeria
E-mail: john.akinode@federalpolyilaro.edu.ng; samson.oloruntoba@federalpolyilaro.edu.ng

*corresponding author details: Akinode John Lekan, john.akinode@federalpolyilaro.edu.ng

ABSTRACT

The wide spread of Corona virus or Covid-19 pandemic across the world has affected educational system worldwide, resulting to partial closures of schools. In line with the efforts to contain the spread, governments in different countries have issued directives to close schools to curtail the virus. As a result, there is a paradigm shift in the educational sector with the rise of Online/E-Learning and Teaching. The purpose of this study was to examine children perceptions of online coding in a boot camp program in Nigeria during COVID-19 period. The data were collected through interviews and questionnaire from the participants. A sample of fifty students in primary schools (K-8) were randomly selected. The students were examined based on their previous knowledge using the normal classroom teaching and the knowledge acquired using the e-learning platform on Computational thinking via unplugged activities and scratch programming. Sample t-test was used to compare the level of skill/knowledge acquired during online coding class and coding in formal contexts. The result shows that there is no significant difference between the teaching methods because p-value >5% significance level. However, the class activities were hampered by the difficulties in breaking the participants into groups and technical challenges as a result of internet and power failure. It was concluded that both interventions significantly improved students' computational thinking skills and competency.

Keywords: perceptions-learning; computational thinking; unplugged activities; scratch programming

INTRODUCTION

In recent times, the introduction of coding and robotics activities for children in K-8 education has increased drastically. Similarly, modern technological tools and coding environments have presented new opportunities and promoted the need to design effective learning experiences [13]. Globally, children from early age are encouraged to acquire digital competences and computational thinking (CT) skills. The rise in importance of computational thinking skills with respect to STEM (science, technology, engineering, and mathematics) fields has been recognized both by those within the STEM education communities and CS education organizations. According to Papavlasopoulou *et al.* [13], there is growing evidence supporting the introduction of computer science (CS) and computational thinking (CT) into K-12 education.

Computer Science as course is now being taught among children in England as a mandatory requirement of the primary National Curriculum Department for Education [3]. In 2016, President Obama launched the initiative 'Computer Science for all' with the aim of empowering US students, from Kindergarten through high school, to learn computer science and to be equipped with the computational thinking (CT) skills needed to be creators and not just consumers in the digital economy [14]. Without any iota of doubt, there has been a considerable increase in the Computer Science opportunities in K-12 category. Tewes [16] reiterates that administrators and tutors emphasize that teaching computer science can stimulate important 21st Century skills for learners. The 21st century skills include critical thinking, computational thinking, problem solving, creativity, and collaboration. Computer programming for young children has grown in popularity among both educators and product developers. Several countries have introduced coding education into school curriculum in order to enhance children's computational thinking and coding skills [18]. Turan and Aydoğdu [18] submit that Coding and Robotic education provides a platform for individuals to be immersed in the problem-solving process, test their hypotheses and make personally worthy discoveries. Discovering of the coding at early ages helps children understand, learn and apply the coding easier in their future lives. Furthermore, early exposure to coding will stimulate children interest to develop various skills such as, direction, movement and mathematics and also learn how to work together in a team. Learning computer science (CS) skills can benefit students economically and academically. In the United States, job opportunities in computer and information technology are projected to increase 13% in 10 years, compared to 7% overall projected job growth [11]. Numerous studies have indicated a host of benefits from learning CS, including improvement in student engagement, motivation, confidence, problem-solving, communication, and science, technology, engineering, and math (STEM) learning and performance.

Computational thinking and coding activities for young students are becoming an integral part of contemporary informal learning in different contexts (e.g., in makerspaces, after school activities, museums, libraries etc.). It is obvious that kids should start developing computational thinking skills early, Jennifer et al. [2], and thus, several organizations design and deliver coding activities, as part of their curriculum or their outreach program. Computational Thinking describes problem solving, design of systems and understanding human behavior by employing central concepts of CS [22].

This research work seeks to investigate the perception of children in a face-face coding workshop and online coding workshop. To address the aforementioned proposition, we conducted a study with 50 children participating in coding activities. We used both unplugged activities and computer codding techniques to measure their engagement and collaboration during the activity and to get their feedbacks on their learning experience in both physical class and online coding class. By investigating the impact of the two settings in learning coding among children, we provide a quantified evidence of how those two important elements moderate other attitudes and enable various insights for the design of future coding activities.

This study investigated primary school children's performance on a programming assessment after engaging in a 5-week online coding class through Zoom and WhatsApp applications. Children used the unplugged activities to solve computational thinking problems and Scratch programming tool to create animated stories, animations and games.

RELATED WORKS

Turan *et al.* [18] carried out a study to determine the effect of robotic coding education on pre-school children's skills of scientific process. The Study group consisted of 30 children aged five who studied in an independent kindergarten connected to Ministry of National Education in Refahiye district. In the study, experimental design with pre-test, post-test and control groups was used. "Scale for Preschool Students' Basic Skills.

Sharma *et al.* [15], in their research work investigated how collaboration and engagement moderate children's attitudes about coding activities. The authors designed a study with 44 children (between 8 and 17 years old) who participated in a full-day coding activity. They measured their engagement and collaboration during the activity by recording their gaze, and their attitudes in relation to their learning, enjoyment, team-work and intention by post-activity survey instruments. The analysis shows that there is a significant moderating effect of collaboration and engagement on children's attitudes.

CODING AT EARLY GRADE IN DIFFERENT COUNTRIES

Computer coding and Computational thinking have become an important part of the school curricula in different countries. The United Kingdom, being pace-setter has embedded computer programming as a mandatory course starting from primary school [6]. Similarly, Denmark, also encourage digital literacy, focusing on the knowledge gained from building technologies [19]. Other countries that have implemented coding in their curriculum are listed below

- Singapore.:CS education has been adopted in Singapore before 2014, it is going to be mandatory starting from 2020. Some of the popular coding academy in Singapore include FirstCode Academy, Saturday Kids, Computhink Kids, SG Code Campus and Early Coders Academy etc.

- Others countries include Austria, Hungary, Denmark, France, Spain, Portugal and Bulgaria.

COMPUTATIONAL THINKING

Wing [21], in his research work, described Computational Thinking (CT) as "the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can effectively be carried out by an information processing agent.

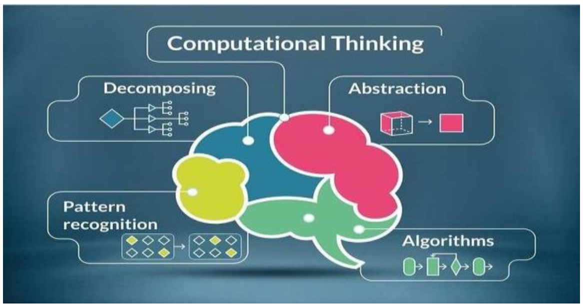

FIGURE 1: The elements of Computational thinking [4]

INITIATIVES FOR ROBOTICS AND COMPUTER SCIENCE EDUCATION

In Europe, some organizations have been become pace-setters in driving interest in coding in pedagogical context. The organizations include Code.org, Codeacademy, Africacodeweek.They emphasized the need to create skills that support future career opportunities while highlighting the educational advantages that coding presents.

TOOLS FOR TEACHING CODING

Scratch

Scratch is arguably the most popular programming environments for Children and Beginners. It is used by Millions of school Children across the World, mostly in out-of-school contexts [1]. The concept of Scratch programming was conceived at MIT media labs and as a descendent of Logo, offers much of the same functionality [7]. Scratch inherits different unique component from Logo, it is also different in many ways. Scratch provides an opportunities and platform to learn important computational thinking concepts as well as offering a platform for learning 21st century skill. Brennan *et al* [2] submit that Scratch-based projects provide a useful framework for thinking about programming skills. The three key dimensions identified in their proposed framework includes: Computational concepts, Computational practices and Computational perspectives.

Scratch JR

ScratchJr, a free programming language was designed to teach children ages 5-7 coding and computational thinking ScratchJr leverage on block programming concept to allow children to create their own imaginative stories and games. The ScratchJr programming app was created as a collaboration among the DevTech Research Group at Tufts University, MIT's Lifelong Kindergarten Group, and the Playful Invention Company through generous funding from the National Science Foundation (DRL-1118664 Award) and the Scratch Foundation. In Scratch environment, dragging blocks into a coding area -and then snapping them together creates code [26]

Tynker

Tynker, a multimedia-authoring tool and visual programming language was invented in 2013.Tynker utilizes visual code blocks to introduce logic and programming concepts to children. This platform provides free activities, mainly games and stories, for children to learn code during the popular hour of code [20].

Blockly

Google's Blockly is a graphical language implemented in JavaScript. It allows user to write program by dragging and dropping code blocks onto a design surface. At its heart, Blockly is a client-side JavaScript library for creating visual block programming editors that can also be compiled into Dart and Phyton code too.

MIT APP Inventor

App Inventor (for Android), is an open-source provided by Google and maintained by MIT (Massachusetts Institute of Technology). It has a graphical interface which assist beginner to create applications [10].

METHODOLOGY

In this section, we present the methodological details of our study, like, the measurements used and the data analysis implemented.

For the purpose of this study, an online coding programme was organized using the Zoom client and WhatsApp application by Littlecoders Nigeria (A robotic and Coding initiative for kids) in South West Nigeria. The workshops have been designed following the constructionist approach and its main principle, learning-by-doing as done by previous efforts.

The online summer Coding workshops are out-school activities, in which students from K-8 education participated. Our participants are children from 6 to 12 years, and during the workshop they learnt about Computational thinking using unplugged activities and then, they code their own game using Scratch programming language. This activity lasts about three hours daily for 5 weeks. The parents were also involved in the workshop as they help their kids to set up the zoom platform and also help in assignment submission via WhatsApp and E-mail. The data were collected through interviews and questionnaire from the participants. A sample of fifty students in primary schools (K-8) were randomly selected. The students were examined based on their previous knowledge using the normal classroom teaching and the knowledge acquired using the e-learning platform on Computational thinking via unplugged activities and scratch programming. Sample t-test was used to compare the level of skill/knowledge acquired during online coding class and coding in formal contexts.

DESCRIPTION OF ACTIVITIES

Activity-based unplugged coding

The participants implemented the basic principles of Computational thinking using unplugged activities. The kids were asked to prepare the recipe for indomie (unique brand of Instant Noodles) and a direction required to travel from point A and B.

Coding Activities using Scratch Programming

The participants used scratch to design games, animation, stories and solve mathematical problems.

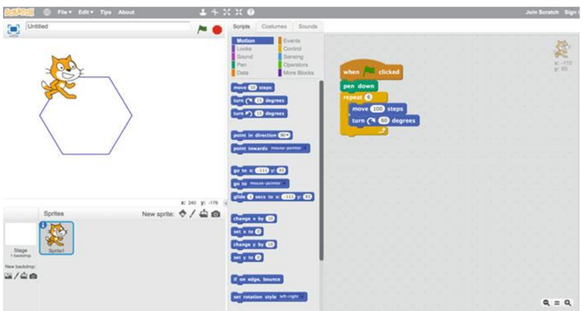

FIGURE 2: Sample scratch program [27].

FIGURE 3: Cross Section of participants during the face-to-face Coding Class.

EVALUATION TOOLS: ASSESSMENT AND FEEDBACKS

The feedback mechanism adopted during the online coding workshop depends on the type of the assessments. However, the assessments are divided into Quiz and Project categories.

Quiz

The quiz competitions were conducted through Kahoot and Gradescope

Kahoot

Kahoot! is a game-based learning platform, used as educational technology in schools and other educational institutions. Its learning games, "Kahoots", are user-generated multiple-choice quizzes that can be accessed via a web browser or the Kahoot app.

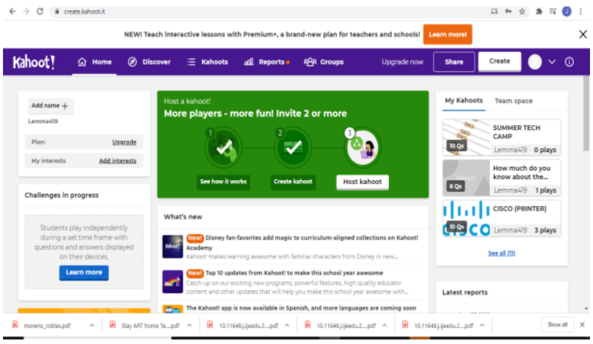

FIGURE 4: Kahoot home page [25]

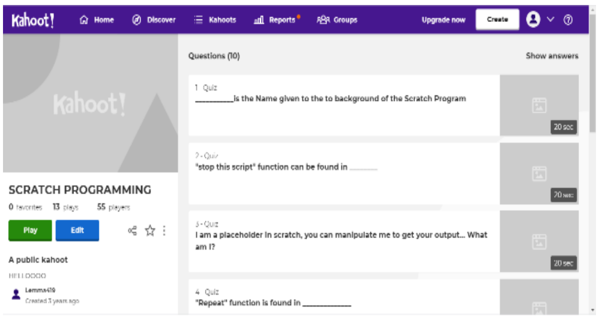

FIGURE 5: Kahoot sample quiz [25]

Gradescope

Gradescope grading software allows students to receive faster and more detailed feedback on their work, and allows instructors to see detailed assignment and question analytics. It is an easy way to take submissions digitally in order to preserve the original work and allow for quick and easy viewing from anywhere.

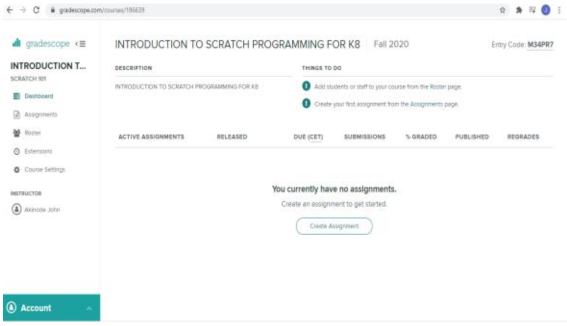

FIGURE 6: Gradescope homepage [24]

PROJECT BASED GRADING

Drscratch

Dr scratch is an analytical tool that evaluates pupil Scratch projects in a variety of computational areas. It is an online tool for evaluating the effectiveness of individual Scratch projects in terms of the computational thinking evident in the project. The tool provides feedback on a variety of computational areas, including: flow control, data representation, abstraction, user interactivity, synchronisation, parallelism and logic, as well as use of best visual programming practice (eg use of sprites attributes and naming, and script performance).

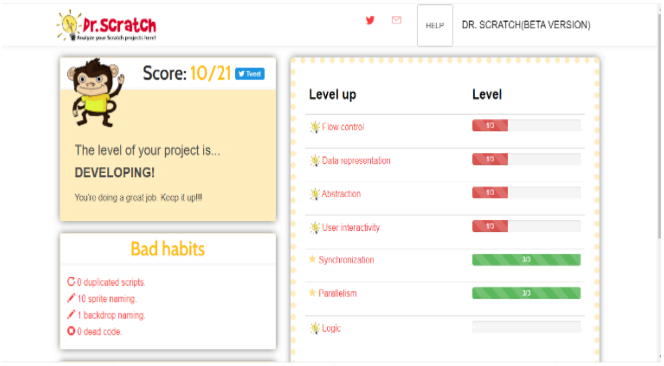

FIGURE 7: Example of DrScratch Project Evaluation [23]

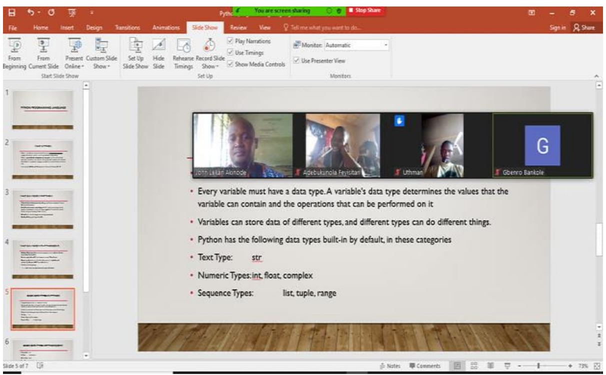

FIGURE 8: Cross Section of participants during the Online Coding Class via Zoom.

RESULT ANALYSIS

TABLE 1: Average performance of pupils in both Face Workshop and Online-Coding Workshop

Tested Knowledge	Teaching Mode	N	Mean	Standard Deviation	Standard Error Mean
A	Contact Teaching	50	3.64	0.921	0.130
	Online Teaching	50	3.50	0.995	0.141
B	Contact Teaching	50	3.42	0.992	0.140
	Online Teaching	50	3.44	1.033	0.146

The knowledge of the pupils was examined by given then practical examination and their performance was graded from 1 to 5. The average performance of their comprehension in A and B are as presented in table 1. The average performance for skill A using normal contact teaching method was 3.64 with standard deviation of 0.921 and the standard error of mean was 0.130. However, the result for the online teaching method indicates an average of 3.50 with standard deviation of 0.995 and standard error for mean of 0.141.

On the tested knowledge B, the average performance for normal contact teaching method was 3.42 with standard deviation of 0.992 and standard error of mean of 0.140. In addition, the online teaching method had an average of 3.44 with standard error of mean of 0.146.

TABLE 2: t-test for the Average performance of pupils

Tested Knowledge	t-Statistics	df	Sig-value	Mean Difference	Standard Error Difference
A	0.730	98	0.467	0.140	0.192
B	-0.099	98	0.922	-0.020	0.203

CONCLUSIONS

The students were examined based on their previous knowledge using the normal classroom teaching and the knowledge acquired using the e-learning platform on Computational thinking via unplugged activities and scratch programming. Sample t-test was used to compare the level of skill/knowledge acquired during online coding class and coding in formal contexts. The result shows that there is no significant difference between the teaching methods because *p*-value >5% significance level. However, the class activities were hampered by the difficulties in breaking the participants into groups and technical challenges as a result of internet and power failure. It was concluded that both interventions significantly improved students' computational thinking skills and competency.

ACKNOWLEDGMENT

The authors wish to acknowledge Architect Olusegun Aluko (PhD), Rector, Federal Polytechnic Ilaro, Nigeria for providing a well-equipped Computer Laboratory for the "KIDSCANCODE" Children Summer Camp.

REFERENCES

[1] Benton, L., Hoyles, C., Kalas, I., & Noss, R. (2017). Bridging Primary Programming and Mathematics: some findings of design research in England. Digital Experiences in Mathematics Education.

[2] Brennan, K., & Resnick, M. (2012). New frameworks for studying and assessing the development of computational thinking. Annual American Educational Research Association Meeting, Vancouver, BC, Canada.

[3] Department for Education [DfE] (2013). Computing programmes of study: key stages 1 and 2. London

[4] Harimurti R. , Ekohariadi , Munoto and IGP Asto IGP (2019). The concept of computational thinking toward information and communication technology learning IOP Conf. Series: Materials Science and Engineering 535

[5] Jennifer Tsan, Collin F Lynch, and Kristy Elizabeth Boyer(2018). alright, what do we need?: A study of young coders collaborative dialogue. International Journal of Child-Computer Interaction.

[6] Jones, S. P. (2013). Computing at school in the UK. http://research.microsoft.com/en-us/um/people/simonpj/papers/cas/computingatschoolcacm.pdf.

[7] Lewis, C. M. (2010). How programming environment shapes perception, learning and goals: Logo vs. Scratch. SIGCSE'10 - Proceedings of the 41st ACM Technical Symposium on Computer Science Education. https://doi.org/10.1145/1734263.1734383

[8] Marina Umaschi Bers, Louise Flannery, Elizabeth R Kazakoff, and Amanda Sullivan. Computational thinking and tinkering: Exploration of an early childhood robotics curriculum. Computers & Education, 72:145–157, 2014

[9] Maloney, J., Resnick, M., Rusk, N., Silverman, B., & Eastmond, E. (2010). The scratch programming language and environment. ACM Transactions on Computing Education. https://doi.org/10.1145/1868358.1868363

[10] Monika Kohli & Harmeet Kaur (2015). Proposing mobile app for Buses information service using mobile technology. International Journal of Scientific and Research Publications, Volume 5, Issue 9, September 2015 1 ISSN 2250-3153

[11] Mason, S. L., & Rich, P. J. (2019). Preparing elementary school teachers to teach computing, coding, and computational thinking. Contemporary Issues in Technology and Teacher Education, 19(4), 790-824

[12] Moreno-León J and Robles G. 2015. Dr. Scratch: a Web Tool to Automatically Evaluate Scratch Projects.https://doi.org/10.1145/2818314.2818338

[13] Papavlasopoulou, S., Giannakos, M. N., & Jaccheri, L. (2019). Exploring children's learning experience in constructionism-based coding activities through design-based research. Computers in Human Behavior. https://doi.org/10.1016/j.chb.2019.01.008

[14] Price, C. B., & Price-Mohr, R. M. (2018). An Evaluation of Primary School Children Coding Using a Text-Based Language (Java). Computers in the Schools. https://doi.org/10.1080/07380569.2018.1531613

[15] Sharma k., Papavlasopoulou S and Giannakos M., Coding games and robots to enhance computational thinking: How collaboration and engagement moderate children's attitudes? International Journal of Child-Computer Interaction (2019), https://doi.org/10.1016/j.ijcci.2019.04.004

[16] Tewes, Ashley. (2019). The Effects of Incorporating Coding on Student Experience and Understanding of Middle School Mathematical Concepts. Retrieved from Sophia, the St. Catherine University repository website: https://sophia.stkate.edu/maed/312

[17] Tuhkala A, Wagner M. L.,Nielsen N., Iversen O.S. and Kärkkäinen .(2018).Technology comprehension: Scaling making into a national discipline.Paper presented at the proceedings of the conference on creativity and making in education.

[18] Turan, S., & Aydoğdu, F. (2020). Effect of coding and robotic education on pre-school children's skills of scientific process. Education and Information Technologies. https://doi.org/10.1007/s10639-020-10178-4

[19] Tuhkala, A., Wagner, M.-L., Nielsen, N., Iversen, O. S., & Kärkkäinen, T. (2018). Technology comprehension: Scaling making into a national discipline. Paper presentedat the proceedings of the conference on creativity and making in education.

[20] Ugur Tevfik Kaplancali, Zafer Demirkol. Teaching Coding to Children: A Methodology for Kids 5+. International Journal of Elementary Education. Vol. 6, No. 4, 2017, pp. 32-37. doi: 10.11648/j.ijeedu.20170604.11

[21] Wing, J. M. (2011). Computational thinking: What and why. The Link. Retrieved from http://www.cs.cmu.edu/link/research-notebook-computational-thinking-what-and-why

[22] Wing, J. M. (2006). Computational thinking. Communications of the ACM, 49(3), 33-35. doi:10.1145/1118178.1118215

[23] http://www.drscratch.org/

[24] https://www.gradescope.com/

[25] https://kahoot.it/

[26] https://www.scratchjr.org/

[27] https://scratch.mit.edu/