Coding and computational thinking: Foundation Phase mathematics students' experiences and reflections

Nothile Kunene & Marietjie Havenga

North-West University
Potchefstroom, South Africa

ABSTRACT: In response to the Fourth Industrial Revolution, the South African government has formulated a preliminary coding and robotics curriculum to equip learners for engagement in a digital society. As a result, it is imperative to provide pre-service teachers with the necessary support for the development of essential skills required for the future. This current study aimed to go beyond traditional instructional practices, and to investigate mathematics students' experiences and reflections when introduced to coding and computational thinking. A qualitative methodology was employed, and one cohort of 167 fourth-year Foundation Phase BEd mathematics students participated. Students collaborated in groups over a period of four weeks using the Scratch visual environment. Activities comprised a coding assignment, participants' reflections on their experiences, as well as semi-structured focus group interviews. Findings revealed that using block-based coding enhanced participants' computational thinking, improved their mathematics understanding, and inspired them to develop several self-directed learning abilities.

Keywords: Coding experiences, computational thinking, mathematics education students, reflections

INTRODUCTION

The World Economic Forum (WEF) has stated the importance of the Fourth Industrial Revolution (4IR) and artificial intelligence (AI) being integrated into future jobs and highlighted education as one of the disciplines with the *highest exposure to AI*, which may require additional skill development for the world of work [1]. Consequently, the 4IR and AI are considered important drivers for skill development and innovation in education [2]. Education 4.0 requires the development of relevant knowledge and skills to meet future needs in order to be fluent in the context of AI, for example using ChatGPT responsibly and working on coding and robotics. Accordingly, curricula for the 4IR need to be designed with the aim of addressing real-world problems and promoting self-directed learning (SDL) abilities [3].

The South-African concept curriculum for coding and robotics has been piloted in several schools with the aim of preparing learners to solve open-ended problems and function in a digital and information-driven world associated with the 4IR and Fifth Industrial Revolution (5IR) [4]. Coding involves the logical structuring of statements by using, for example, visual blocks or textual programming languages, such as Python to solve problems.

Furthermore, coding and robotics is associated with the introduction of computational thinking (CT) in schools [5]. CT is considered an essential competency [6] and problem-solving process based on the principles of computer science, and is an important way of thinking that can be implemented when working on digital platforms such as Scratch [7][8].

Although pre-service Foundation Phase (FP) (grade 1-3) mathematics students are familiar with applying formulas and solving problems, their knowledge of coding and CT is limited and, to most, probably unknown [8]. Engaging pre-service students in CT activities cannot be overstated [9-12]. There is a need to facilitate students in implementing new technologies regarding coding and CT, and planning lessons on how they would go about providing for such skill development in their future classes. Consequently, the study on which this article is based endeavoured to indicate how FP mathematics students experienced coding activities, and how it influenced their CT. The study was guided by the following research questions:

1. What are mathematics students' experiences and reflections regarding their coding activities?
2. How did the coding tasks contribute to the enhancement of participating students' higher-order thinking and CT?
3. How did coding and CT contribute to the cultivation of participating students' SDL abilities?

THEORETICAL FRAMEWORK

Certain aspects pertaining to this research are outlined in more detail in the following sections below.

Foundation Phase Students' Mathematics Knowledge and Skills

It is essential that FP students understand how learners construct knowledge in order to scaffold them with insight in the classrooms from an early age. Constructivism, a learning theory associated with prominent scholars, such as Piaget and Vygotsky, focuses on the active construction of knowledge and involves individuals' own understanding of the world [13]. Social constructivism suggests that individuals build new knowledge through their experiences, reflections and interaction with peers [14]. In other words, learning is considered a collective activity where students work together on challenging tasks. They are required to develop essential knowledge and skills and apply higher-order thinking, such as problem solving, creativity and critical thinking [15].

There is substantiated evidence that the constructivist approach transforms the dynamics of teaching and learning mathematics, connecting it with real-life experiences, rather than only teaching how to solve mathematics problems [16]. Consequently, the teacher has a dual responsibility of introducing learners to fundamental concepts and skills, and of facilitating them in solving a variety of meaningful problems with the aim of becoming autonomous in their mathematical thinking [16].

Coding and Computational Thinking

Coding and CT are underpinned by the constructionist learning theory coined by Papert who elaborated on the constructivist view of learning [17]. Papert considered the construction of new knowledge through the creation of physical objects, often referred to as *thinking objects*, while students create their own reality [18].

Scratch is a visual programming environment and on-line platform that enables learners to create interactive stories, simulations and object (sprite) movements in a playful environment using a drag-and-drop interface [7][19]. Through Scratch, learners are introduced to aspects associated with mathematics, such as variables, logical thinking, mathematical functions and angles within geometric shapes [7]. Coding has therefore the potential to lead to mathematical understanding and concepts, such as early algebraic thinking [10]. Coding is also essential to CT, which is considered part of logical reasoning and is emphasised as a 21st Century skill [19].

Inspired by the principles of computer science, CT is characterised by the following core elements: problem decomposition (breaking down intricate problems into simpler ones), development of algorithms (providing step-by-step solutions to problems), abstraction (emphasising essentials while ignoring certain details), and the application of thinking patterns (known as pattern recognition), as highlighted in the seminal work of Wing [6]. CT focuses on certain skills that learners develop, such as practicing programming and algorithms that promote the development of higher-order thinking [9]. Moreover, CT involves elements of development that are useful for people in their professional and personal lives as ways of organising the resolution of problems and representing reality [19]. To succeed in coding and robotics, the ability to persist in solving problems and a sense of responsibility in own learning are important.

Self-directed Learning

SDL involves a learning process where individuals take responsibility for promoting their own learning, and emphasising initiative in learning, autonomy, independence, reflection and flexibility [20]. SDL is a valuable skill and essential for the 4IR where people are confronted with robotics, machine learning, and 3D printing, among others. In addition, Penprase highlights the simultaneous development of adaptability, critical thinking, innovation, ethical behaviour, SDL and lifelong learning to prepare for challenges of the 4IR [3]. Such skill development is essential in the dynamic landscape of education in order to stay relevant. Krishnannair and Krishnannair further emphasise the importance of relevant learning environments in higher education to be positioned for the challenging world of the 4IR [21].

In an ever-changing future, the ability to adapt continuously is crucial. Enhancing the learning experience can be accomplished through cultivating metacognitive awareness, fostering self-regulation, and enhancing the skill of reflecting on one's own thinking [22]. Reflection is, therefore, an integral part of the learning process that enhances critical thinking and problem solving, and empowers learners to take ownership of their educational journey. Participants' reflections and experiences are outlined in the next section.

RESEARCH CONTEXT AND METHODOLOGY

Participants and Ethical Aspects

The study involved one cohort of 167 fourth-year FP BEd students enrolled in a mathematics course aimed at teaching future grade R-3 learners. A collaborative learning approach was followed, where participants engaged in group activities within 35 groups comprising four or five members each. Notably, participants had the freedom to choose their peers with whom they were willing to collaborate. Each group chose their own group leader and scribe, and they assigned various responsibilities to each member. The research project was approved by the relevant Scientific Committee, Ethics Committee and the University Research Data Gatekeeper Committee. As part of ethics practice, informed consent was obtained from all participants indicating their willingness to participate.

Research Context and Class Activities

The course outcomes required students to demonstrate integrated knowledge and practical application of mathematics vocabulary and symbolism, problem-solving processes and design of lessons. Class activities were done over a period of four weeks. The first author (i.e. the class lecturer) introduced the Scratch programming environment and corresponding coding activities to students. They were also scaffolded with relevant links to supportive materials and videos on Scratch, and worked together on the subsequent activities:

- Activity 1A involved a guided Scratch task (Jumping Game).
- Activity 1B (assignment) required students to design a simple game in Scratch related to mathematics patterns.
- Students had small-group meetings to reflect on their game activities.
- They completed individual reflections by answering 19 questions on the learning management system (LMS).

Instructions regarding activity 1A were the following: use Scratch to design a Jumping Game. Watch a video about Scratch provided on the LMS, change the sprite (cat is the default object) and choose your own sprite from the library. Make screen prints of every step as follows: 1) make the sprite jump; 2) design a moving obstacle; 3) stop the game; 4) add more obstacles; and 5) keep the score. Explain the mathematical concepts of patterns within this game.

Activity 1B involved the following: design your own Scratch activity related to the topics, such as patterns (FP mathematics curriculum). Include five images of block codes with related output sequentially to indicate your thinking behind the drawings. List the aspects of CT used in your game, and explain how this activity links to patterns as a mathematics topic. In addition, complete individual reflection questions about your experiences on the LMS.

Data Collection and Analysis

Participants submitted activities 1A and 1B as Word or PDF documents, their narrative reflections regarding group meetings, as well as individual responses to questions on the LMS. Data were manually coded using open coding [23] and certain themes emerged. Evidence of participants' experiences on planning, support and group collaboration, challenges and problems, high-order thinking, CT and examples of SDL emerged from the findings.

RESULTS AND DISCUSSION

This section delineates the results by means of participants' examples (Figure 1) and discussions organised according to emergent themes in this study.

Figure 1 below illustrates an example by group 5 in response to activity 1A on the Jumping Game in Scratch.

According to group 5, they created their game by changing the sprite and choose their own which they named Mika. They made Mika to jump over an obstacle and landed on the other side of the obstacle as per instruction 2 of the activity. Seeing Mika on the other side of the obstacle was enhanced by their correct coding.

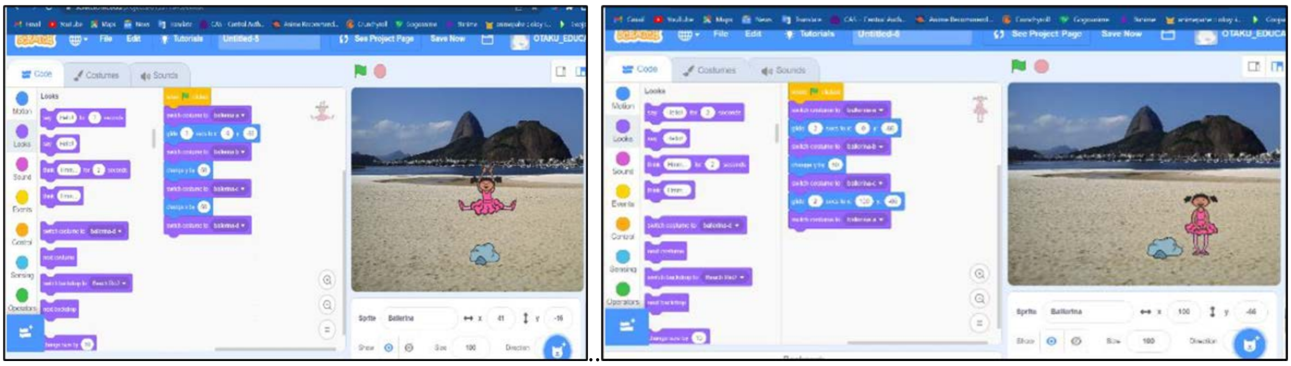

Figure 1: Example of group 5's Jumping Game in Scratch.

The following themes emerged: 1) participants' experiences and reflections regarding Scratch; 2) higher-order thinking and CT; and 3) participants' perceptions of their own learning and self-directedness.

Table 1 presents some responses regarding participants' experiences and reflections on their activities. The theme reflects issues of planning, group collaboration, and support. P1, for example, indicates the number of the participant. All responses are reproduced verbatim and unedited.

Table 1: Exemplars of participants' experiences and reflections regarding Scratch.

Experiences and reflections	Participants' responses
Planning and time management	*We used WhatsApp, YouTube videos to assist in planning.* (P15) *We assigned each other to go search and bring ideas of how to execute the game. (P2)* *We met face-to-face with my group members …at our own time to plan.* (P98) *We struggled a lot, the activity was overwhelming, and it needed more time.* (P44) *[S]ometimes our* Sprite *would disappear as we were adding more obstacles …we got frustrated as we had to start over on so many occasions. (P1)*
Group collaboration and support	*We took turns in creating the games, according to members' skills and capabilities.* (P5) *Our communication was clear …motivating us to complete the game.* (P22) *We respect each other and all the members' opinions, we never experienced any fights or complaints from all the members.* (P3) *Some were dragging their feet at first, the second [activity], was a bit better.* (P67) *Personal interaction was not always good. Disagreements created conflicts.* (P51) *[A]sked for assistance from other group members.* (P119) *We could not get a game that we all liked, as a group we were not able to even decide which characters best suited the games options.* (P140)

Table 2 provides exemplars of participants' feedback regarding their higher-order thinking and CT (algorithmic thinking, abstraction, pattern recognition and decomposition).

Table 2: Exemplars of participants' higher-order thinking and CT.

Higher-order thinking and CT	Participants' responses
Algorithmic thinking	*[The] second activity was slightly different to the first one as we now had to start our coding from scratch.* (P97) *Coded to get a desired output, i.e. 40 steps, motion code, action, etc.* (P12) *[We] took screenshots for every step done, and identified patterns created.* (P36) *[D]esigning the* flying octopus *game which was a bit of work and challenging, but it was also beneficial as at least we were able to learn some skills on Scratch.* (P63) *[The] jump we changed (x by -10) and we repeated it (10 times) in order to make our rabbit to jump high.* (P135)
Abstraction	No explicit mention was made by participants regarding abstraction.
Pattern recognition	*[D]escribed and compared shapes in terms of size, colour, and shape to form patterns. (P121)* *We used fruits to make pattern sequences.* (P53) *[M]ake diamond pattern repeatedly to form a Diamond flower.* (P144) *Our Sprite was used to draw circles using different colours, to form circle pattern.* (P11)
Decomposition	*We made our basketball [Jumping Game] 70% less than the size of the Rabbit in order for the Rabbit to be able to jump over. For our basketball to glide or move from right to left we positioned it at (x:209 and y:-157) and for the basketball to glide we made it to delay for 3 seconds.* (P125) *[W]e added our* stop all *control and made our basketball to stop until it touches the rabbit when we clicked. To avoid the rabbit to freeze we coded our motion to be (x:180 and y:-120).* (P149) *Add more obstacles …Made our duplicates to wait 3 seconds before it glides. Hid and showed our basketball so that they all do not appear at the same time.* (P71)

In Table 3, instances of participants' responses regarding their perceptions of their own learning and self-directedness are given.

Table 3: Exemplars of participants' perceptions of their own learning and self-directedness.

Perceptions of learning and SDL	Participants' responses
Planning the game	*While we were planning … some were very stressed, clueless, confused, nervous not knowing where to start and did not feel technologically advanced.* (P133) *We must start to explore topics of interest and agree on what we want to do.* (P129)
Identification of relevant resources	*Our cellphones became handy when searching for the games and coding.* (P88) *We used textbooks and worksheets to remind ourselves of pattern designs that we can create for our assignment.* (P160) *We collaboratively revisited PowerPoints from previous lectures. We sought advice from the lecturer on how to go about coding [regarding] some of the procedures.* (P73)

Curiosity and persistence	*[I]nvestigate, foster our curiosity and imagination as we plan for this game.* (P150) *The scratch app allowed us to think outside the box and express our thoughts* (P79) *Programming this game has been exciting making learning something enjoyable, improves skills and the urge to implement ideas.* (P20)
Evaluation and reflection	*The implementation of coding and robotics at schools must kick in and who ever engages in it must have time and patience.* (P87) *We find this app interesting because it promotes computational thinking and problem-solving skills, creative teaching and learning, self-expression and collaboration.* (P107) *We found the game good, had fun, and enjoyed in such a way that we even got addicted to playing and refreshing with Scratch games even later in the day.* (P106) *Such an engagement demanded working in a team, having leadership skills, accountability and discipline.* (P28) *This game motivates the ability of a learner to learn mathematics on his/her own or individually without teacher or learner assistance.* (P60)

The research questions were addressed, and the first question was: What are mathematics students' experiences and reflections regarding their coding activities?

The participating students had very little to no background of coding and CT (Table 1). During their planning, they converged in face-to-face mode in class, small-group meetings at times convenient to them, as well as on-line platforms, such as Microsoft Teams and WhatsApp, to share roles and they brought along some game ideas after consulting YouTube videos. Positive and negative experiences regarding group collaboration were shared.

There was noticeable progress as support was rendered within each group through collaboration. Group members agreed that their communication was clear, which in return motivated them to complete the game they had programmed. Members took turns among themselves to create the games in accordance with their skills and abilities. Some groups reported that they had respected each other, and several opinions had been considered. This led to environments without fighting and complaints from the group members.

Unfortunately, in some groups, participants were not cooperative. They were dragging their feet and their personal interaction was not always good, which resulted in disagreement and conflict during the development of their games. Certain groups experienced some challenges of the disappearing sprite character, as they were adding more obstacles. This led to frustrations within the group. Others found it challenging to design a suitable and relevant game that they all liked. Similar experiences have been reported by scholars regarding student collaboration in mathematical game-based tasks [24].

Regarding the contribution of coding tasks to enhance higher-order thinking and CT (second question), participants demonstrated several skills (Table 2). They embraced problem solving, and responded to algorithmic thinking, pattern recognition and decomposition as aligned with the concepts of CT, highlighted by Wing [6]. One group mentioned that they realised that the second activity was slightly different from the first, and this required of them to develop their own game related to a mathematics topic.

In terms of algorithmic thinking, participants planned their code to get the desired output, such as taking 40 steps and using several coding blocks. P63 mentioned that designing the *flying octopus* game was challenging, but also beneficial as they developed certain skills in Scratch. Unfortunately, examples of abstraction were not explicitly mentioned. For pattern recognition, participants compared shapes in terms of size, colour and shape to form certain mathematical patterns. Others mentioned that they programmed the Scratch sprite to form a diamond flower pattern. Participants also managed to identify mathematics in their Scratch games, such as shapes and lines. They revealed that their sprite was used to draw circle patterns with different colours. While breaking the problem into smaller sections, as an indication of decomposition and algorithmic thinking (Table 2), participants repeated some coding blocks in order to make their rabbit jump high. They also designed a ball for the rabbit to jump over, which required precise positioning and coding.

The third research question investigated how coding and CT may contribute to the cultivation of SDL skills. Considering the importance of SDL, as advocated by Knowles [20], participants mentioned detailed planning of activities. Initially, the mathematics students were confused, did not know where to start, and did not feel technologically advanced but persisted in their learning. They managed to identify relevant resources, such as textbooks, YouTube videos and PowerPoints to assist them with the gaming tasks.

The participants also demonstrated attributes of curiosity, and were inspired to associate mathematics learning with real-world problems. Students acknowledged that they found the Scratch application interesting as it promotes CT and problem-solving skills, creative thinking, self-expression and collaboration, and it allowed them to think outside the box. Furthermore, group members revealed that programming this game was exciting, as it contributed to improving their skills and implementing new ideas. Participants realised that such an engagement demanded working in a group, and having leadership skills, accountability and discipline. In their adventure of becoming self-directed, participants reflected on their activities, and they found the game enjoyable. As a result, they believed that such games inspire them to learn mathematics in order to address real-world problems that require creative and innovative thinking.

CONCLUSIONS

The study emphasised the importance of Foundation Phase mathematics students' implementation of coding to enhance their CT. Students' experiences regarding the coding activities were also identified and documented. Participating students planned their activities, monitored their progress, reflected on CT strategies, such as algorithmic thinking, pattern recognition and decomposition. Creativity, knowledge acquisition and positive experiences during the intervention were visible.

The use of coding enhanced students' CT, improved their mathematics understanding, and motivated them to engage in active learning. Students also enhanced their SDL skills by viewing YouTube videos to learn more about Scratch. Future research on exploring effective ways to assist Foundation Phase mathematics students in the teaching and learning of coding and robotics is inevitable in the 21st Century and 4IR era.

ACKNOWLEDGEMENTS

The authors would like to acknowledge the students who participated in this project.

REFERENCES

1. Shine, I., World Economic Forum (WEF). Automation or Augmentation? This is how AI will be Integrated into the Jobs of Tomorrow (2023).
2. Josi, N., *Understanding Education 4.0: The Machine Learning-Driven Future of Learning.* Forbes (2023).
3. Penprase, B.E., *The Fourth Industrial Revolution and Higher Education*. In: Gleason, N.W. (Ed), Higher Education in the Era of the Fourth Industrial Revolution. Palgrave Macmillan, Singapore, 207-229 (2018).
4. BusinessTech. Government Finalising New Subjects for Schools in South Africa (2023).
5. Chalmers, C., Robotics and computational thinking in primary school. *Inter. J. of Child-Computer Interaction,* 17, 93-100 (2018).
6. Wing, J.M., Computational thinking. *Communications of the ACM*, 49, **3**, 33-35 (2006).
7. Estapa, A., Hutchison, A. and Nadolny, L., Recommendations to support computational thinking in the elementary classroom. *Technol. and Engng. Teacher*, 77, **4**, 25-29 (2017).
8. Havenga, M. and Jordaan, T., Introducing coding and robotics: prospective mathematics teachers' metacognitive thinking in Scratch. *Proc. Inter. Conf. on Robotics in Educ. (RiE)*, Switzerland: Cham, Springer Nature, 17-26 (2023).
9. Angeli, C. and Giannakos, M., Computational thinking education: issues and challenges. *Computers in Human Behavior*, 105, 106185 (2020).
10. Miller, J., STEM education in the primary years to support mathematical thinking: using coding to identify mathematical structures and patterns. *Zentralblatt für Didaktik der Mathematik (ZDM)*, 1, **6**, 915-927 (2019).
11. Rodríguez-Martínez, J.A., González-Calero, J.A. and Sáez-López, J.M., Computational thinking and mathematics using Scratch: an experiment with sixth-grade students. *Interactive Learning Environments*, 28, **3**, 316-327 (2020).
12. Abd Algani, Y.M., Role, need and benefits of mathematics in the development of society. *J. for the Mathematics Educ. and Teaching Practices*, 3, **1**, 23-29 (2022).
13. Triantafyllou, S.A., Constructivist learning environments. *Proc. 5th Inter. Conf. on Advanced Research in Teaching and Educ.*, 3-6 (2022).
14. Vygotsky, L.S., *Mind in Society*. London: Harvard University Press (1978).
15. Yüksel, A.O., Investigation of pre-service science teachers' learning experiences on educational robotics applications. *J. of Computer and Educ. Research*, 10, **19,** 50-72 (2022).
16. Vintere, A., A constructivist approach to the teaching of mathematics to boost competences needed for sustainable development. *Rural Sustainability Research*, 39, **334**, 1-7 (2018).
17. Ackermann, E., Piaget's constructivism, Papert's constructionism: what's the difference. *Future of Learning Group Publication*, 5, **3**, 1-11 (2001).
18. Papert, S., Children, computers and powerful ideas. *New York: Basic Books*, 10, **1990**, 1-230 (1990).
19. Avello, R., Lavonen, J. and Zapata-Ros, M., Coding and educational robotics and their relationship with computational and creative thinking. a compressive review. *Revista de Educación a Distancia (RED)*, 20, **63**, 1-21 (2020).
20. Knowles, M., *Self-Directed Learning: a Guide for Learners and Teachers*. Chicago: Follett-Publishing Company, (1975).
21. Krishnannair, A. and Krishnannair, S., Learning environments in higher education: their adaptability to the 4th industrial revolution and the *social transformation* discourse. *South African J. of Higher Educ.*, 35, **3**, 65-82 (2021).
22. Potgieter, E. and van der Walt, M., Metacognitive awareness and the zone of proximal intermediate phase mathematics teachers' professional development. *EURASIA J. of Mathematics, Science and Technol. Educ.*, 18, **8,** 1-15 (2022).
23. Saldaña, J., *The Coding Manual for Qualitative Researchers*. London: Sage (2021).
24. Jordaan, T., Havenga, M. and Bunt, B., Mathematical game-based learning: education students' collaboration and on-line experiences during disrupted Covid-19 circumstances. *World Trans. on Engng. and Technol. Educ.*, 19, **3**, 263-270 (2021).

Nothile Kunene is a senior lecturer in the School of Mathematics, Science and Technology Education and the Research Unit Self-Directed Learning in the Faculty of Education at North-West University, South Africa. Her research focuses on problem-solving, problem-based learning, assessment and collaborative learning. She is currently involved in coding and robotics for the Fourth Industrial Revolution towards self-directed learning. She has published at the national level, and also acts as a supervisor for postgraduate students. Kunene received an Institutional Research Excellence Award.

Marietjie Havenga is a Professor in the School of Mathematics, Science and Technology Education and the Research Unit Self-Directed Learning in the Faculty of Education at North-West University, South Africa. She is involved in teaching courses in information technology and computer programming to pre-service teachers on undergraduate level, as well as the supervision of post-graduate students. She has published in various national and international journals, conference proceedings and chapters in scholarly books. She is a reviewer for several international journals and was editor for a scholarly academic book. Her research interests include educational robotics, computer science education and engineering education to enhance students' self-directed learning.