A Project entitled

Comparing programming interest, perceived programming difficulty and computational thinking skills in Block-based and Text-based programming learning in Minecraft in secondary school students

Submitted by

Wu Chui Tim

Submitted to The Education University of Hong Kong for the degree of Bachelor of Education (Honours) (Secondary) – Information and Communication Technology

in *April 2022*

Declaration

I, *Wu Chui Tim* , declare that this research report represents my own work under the supervision of *Title and Name of Project Supervisor*, and that it has not been submitted previously for examination to any tertiary institution.

Signed

Wu Chui Tim

08-04-2022

Abstract

Programming is one of the teaching topics in STEM education. Meanwhile, Minecraft is one of the most popular games among youngsters worldwide and Minecraft education edition allows players to create objects through programming. This study aims at comparing the changes of students' learning interest, perceived programming difficulty and computational thinking skills in text and block interface programming teaching in a game-based programming learning tool-Minecraft. Both qualitative and quantitative research method were applied. Data were collected from 2 class of 10 secondary students by using pre and post domain tests, self-reflection questions and interviews. Results showed that using Minecraft education edition to teach programming is suitable for students learning programming that provide positive effects on students' learning interest, perceived programming difficulty and computational thinking skills. In addition, the study found that Block-based programming learning has greater benefit than Text-based learner in learning interest and perceived programming difficulty.

Content

1. Introduction……7
2. Literature reviews……7
3. Research objectives……11
4. Research questions……11
5. Research methods……12
   5.1 Participants and school context……12
6. Data collection……12
7. Data analysis……13
8. Research implementation……17
   8.1 Teaching schedule……17
9. Result and discussion……20
10. Limitation……31
11. Conclusion and suggestion……32
12. Reference list……33
13. Appendix……36

List of Tables

Table 1. Summary of the data source and research questions addressed……15

Table 2. Teaching schedule of Group A (Block-based group) and Group B (Text-based. group)……...…….....19

Table 3. Part A Pre-test and post results……..20

Table 4. Interview result in Question 1……22

Table 5. Interview result in Question 2……23

Table 6. Self-reflection question Q1 result……..24

Table 7. Part B Pre-test and post results……..25

Table 8. Self-reflection question Q2 result……..27

Table 9. Interview question Q4 result……..30

Table 10. Interview question Q5 result……30

List of Figures

Figure 1: An image with text explanation on Block-based group PowerPoint…….18

Figure 2: An image with text explanation on Text-based group PowerPoint……...…….19

Figure 3. The change in learning interest in Block-based group and Text-based group……...21

Figure. 4 The change in learning interest in Block-based group and Text-based group……...24

Figure 5. Perceived learning difficulties in Block-based and Text-based groups…….26

Figure 6. The changes in computational thinking skills in Block-based and Text-based groups. Q1-Q3……28

Figure 7. The changes in computational thinking skills in Block-based and Text-based groups. Q4-Q8……....29

1. Introduction

The Education Bureau has promoted STEM (Science, Technology, Engineering, and Mathematics) education to local schools since 2015 (The Education Bureau, 2015). Programming is one of the teaching topics in STEM education- schools often use different Block-based and Text-based educational packages, for example, mBot, app inventor and scratch to deliver programming concepts to students. Meanwhile, Minecraft is one of the most popular games among youngsters worldwide and the game allows players to create objects through programming. Therefore, researchers started to investigate the impact of teaching programming by the game Minecraft using the two interfaces (Block-based and Text-based). Therefore, this project aims at exploring the impact in programming interest, perceived programming difficulty and computational thinking skills in Block-based and Text-based programming learning in Minecraft in secondary school students.

2. Literature reviews

Digital game-based learning refers to students learning new knowledge by playing games that have been designed to offer specific learning outcomes (Hiltunen, 2016). Game based learning is a high potential teaching tool because games are motivating, engaging, and enjoyable for learners (Papastergiou, 2009; Garris et al., 2017). Using games for learning has been adopted in teaching many topics, including programming concepts, in order to promote interest of learners (Prensky, 2003).

Papastergiou (2009) also agreed that digital game-based learning increased motivation of learners. Papastergiou (2009) investigated the learning effectiveness and students' learning motivation of computer games for secondary school students' computer science learning. Papastergiou (2009) revealed that digital game-based learning could provide effective and

motivating learning environments for both genders, which also contributed to the positive impact on learning outcomes. Huang (2011) also mentioned that digital game-based learning could increase students' learning motivation. Furthermore, Huang (2011) revealed that educational video games could initiate and support learners' goal-setting activities. Since games had small tasks that accumulated toward achieving ultimate game goals, learners would complete a series of tasks to achieve the goal of the game.

Game-based learning can also bring enjoyment to students, Ritterfeld and Weber(2006) suggested that enjoyment in playing games stems from sensory delight, suspense, thrill, and relief, or achievement, control, and self-efficacy when playing games. Furthermore, achievement and challenges in educational games are a major source of enjoyment in digital games and act as a key motivation for players to engage (McGonigal, 2011). McGonigal (2011) also mentioned that the challenges and "hard" in games came along with enjoyment, learners tend to feel the "hard" in games enjoyable and rewarding.

Barzilai and Blau (2014) also mentioned that game-based learning offers pleasure, interest, excitement to students. The perceived learning in students also increased after learning through games. Furthermore, the research found that students who studied before the educational game performed significantly better than those who did not in the post-game learning outcome assessment, while studying before games did not reduce the enjoyment in playing games for learners (Barzilai & Blau, 2014).

Pratama and Setyaningrum (2018) also agreed that digital games have a strong effect in raising students' interests in learning and suggested the reason behind- the combination of colors, animations, and presentation of material using images in the educational games could strongly boost students' interests in learning. Furthermore, games which are related

to real-life context can increase students' interest in learning, which also educates students in the problem-solving method in real life.

With all the benefits of game-based learning, scholars attempted to investigate the benefits of a popular game, Minecraft, and its implement to game-based learning. Minecraft is an open-ended game which allows children to perform their creativity. Players could construct their own buildings, instruct robots in the game to make actions and cooperate with other players to explore the virtual world. In constructivist learning approaches, Minecraft allows students freedom to explore ideas, solve them, and learn from the process, which enables active knowledge construction (Loyens & Gijbels, 2008; Nebel et al., 2016).

According to Pusey et al. (2016), Minecraft in STEM lessons increased in student engagement and motivation in learning. Pusey et. al (2016) suggested that students experienced an increase in enthusiasm about STEM class, since students considered STEM lessons offered opportunities for them to participate in a "fun" activity. Furthermore, Minecraft learning increased students' enjoyment in lessons because students considered Minecraft interactive, collaborative, interesting, and different to "regular" schoolwork (Pusey et. al, 2016).

Callaghan (2016) also shared the same opinion that Minecraft promoted high engagement and motivation levels for learning, thus it could be a useful teaching tool. In Minecraft, the game encouraged students to exercise their problem-solving skills(computational thinking skills) and to complete their initial planning of their designs so they could continue to "play" on Minecraft (Callaghan, 2016). Thus, students were motivated to continue and engage in the game. Kutay and Oner (2022) also stated that Minecraft programming could raise computational thinking skills of students. According to Kutay and Oner (2022),

students are encourage to break down problems to simple parts, and creating possible solutions of the problem, which are building computational thinking skills in problem-solving and creativity.

In game-based programming learning packages, there are two common interfaces, Block-based and Text-based interface. According to Bau et al. (2017), Block-based programming is effective in teaching computational thinking skills to programming beginners by reducing cognitive load, encouraging recognition of the block shapes rather than Text-based syntax, and avoiding syntax errors.

Weintrop and Wilensk (2017) mentioned Block-based Programming learning gave secondary school students a higher enjoyment and more interest in learning programming. However, perceived difficulty of programming raised in both block based and Text-based programming learning, despite the blocks-based learners performed significantly better than students in the Text-based learners on the post content assessment (Weintrop & Wilensk, 2017). Also, the increase in perceived difficulty in Block-based programming is more significant than Text-based. The researchers suggested that the reason might be the students see a difference between what they were doing in the Block-based interface and the "programming" (Weintrop & Wilensk, 2017).

Zorn et.al (2015) adopted Minecraft as a tool to investigate the impact of Text-based and Block-Based programming in programming interest, perceived programming difficulty and programming enjoyment. They had a same result as Weintrop and Wilensk (2017)- students have a higher enjoyment and interest in both Block-based and Text-based programming learning. However, Zorn et.al (2015) suggested that there is no significant difference in participants' enjoyment and interest between the two interfaces, which is

different from result proposed by Weintrop and Wilensk (2017). They also mentioned that the perceived programming difficulty in both Text-based and Block-based programming decreased after the learning, with is in contrast with the result suggested by Weintrop and Wilensk (2017).

3. Research objectives

Zorn et.al (2015) found that Minecraft increase participants raise their learning interest in programming, enjoyment, reduce in perceived programming difficulty in learning programming through Minecraft. However, the participants are not secondary students from Hong Kong. Does the educational game Minecraft also promote interest, enjoyment, and reduce perceived programming difficulty to secondary students in Hong Kong? These are questions that have not been investigated in the previous literatures. The comparison in Block-based and Text-based programming teaching in learning interest and perceived learning difficulties could also help teachers to choose a more interesting and less resistant way for students to get started in programming learning. This study compares the changes of students' learning interest, perceived programming difficulty and computational thinking skills in text and block interface programming teaching in a game-based programming learning tool- Minecraft.

4. Research questions

Q1. Does teaching programming using Minecraft increase the learning interest of students?

Q2.What are the differences in learning interest in the Block-based programming teaching and Text-based programming teaching?

Q3.What are the impacts of Block-based programming teaching and Text-based programming teaching in learners' perceived learning difficulties?

Q4. What is/are the changes in computational thinking skills in Block-based programming learners and Text-based programming learners?

5. Research methods

5.1 Participants and school context

The research was conducted in Pentecostal Lam Hon Kwong School in Hong Kong in February to March 2022. The 20 Form 1-3 students were taking the STEM club as an extracurricular activity afterschool. The students were divided into two groups (Group A and Group B) randomly and each group had three 60-minutes Minecraft programming lessons, covering basic programming concepts, including command in agent movement, for loop and nested loop. Mixed research methods were adopted in this research. Both qualitative and quantitative research methods were applied. Group A (Block-based group) had a total of 10 students with 4 males and 6 females. While the Group B (Text-based group) had a total of 10 students with 7 males and 3 females.

6. Data collection

All the data was collected by using pre and post-domain tests, self-reflection questions and interviews with students. The collected data was used to analyze students' learning interest, learning interest and perceived programming difficulty and computational thinking skills.

Pre-test and post-test questionnaire

The tests were an online questionnaire on Google form. Students needed to finish the tests before and after the three lessons separately to investigate their change in learning interest and perceived programming difficulty and computational thinking skills before and after the lessons and the changes. The questions in learning interest are based on Intrinsic Motivation Inventory (IMI)(Leng et.al, 2010). The questions in computational thinking skills are based on Computational Thinking scales (CTS) (Korkmaz et.al, 2017).

According to Marsden and Torgerson (2012), the pre-post test design can provide some data which is observable and comparable.

Self-reflection questions

The self-reflection questions are 3 open-ended questions for investigating the reason behind the students' perspective in their learning interest, learning interest and perceived programming difficulty and computational thinking skills. The self-reflection questions was attached in the same document as the questionnaire to the students.

Interviews

The interviews include five open-ended questions. According to Qu and Dumay (2011), interview has been widely recognised as one of the most effective ways in collecting qualitative data in field studies and ethnographic research. Questions of the interview were generated by summarizing the framework of Intrinsic Motivation Inventory (IMI)(Leng et.al, 2010) and Computational Thinking scales (CTS) (Korkmaz et.al, 2017).

7. Data analysis

The study applied the mixed research method. The collected data was used to analyze the result

of the research questions (Refer to Table 1).

Research questions	Research method	Data Source	Analysis method
1. Does teaching programming using Minecraft increase the learning interest of students?	Quantitative	Pre, post-domain test	Finding the mean and standard deviation of the question score.
	Quantitative	Interview	Content analysis
2. What are the differences in learning interest in the Block-based programming teaching and Text-based programming teaching?	Quantitative	Pre, post-domain test	Finding the mean and standard deviation of the question score.
	Qualitative	Self-reflection questions	Content analysis
	Quantitative	Interview	Content analysis
3. What are the impacts of Block-based programming teaching and Text-based programming teaching in learners' perceived learning difficulties?	Quantitative	Survey	Finding the mean and standard deviation of the question score.
	Qualitative	Self-reflection questions	Content analysis
	Quantitative	interview	Content analysis
4. What is/are the changes in computational thinking skills in	Quantitative	Pre, post-domain test	Finding the mean and standard deviation of the question score.
	Quantitative	interview	Content analysis

block-based programming learners and Text-based programming learners?			

Table 1. Summary of the data source and research questions addressed

The data will be collected in the STEM club lessons by pre and post-domain tests, self-reflection questions and student interviews. Three lessons will be conducted to 2 groups of Form 1-3 students. Group A is a Block-based group, using coding blocks to code while Group B is a Text-based group, using javascript to code. Both groups leant programming in the game Minecraft education edition. Then, the data was used to analyze student's interest, perceived programming difficulty and computational thinking skills.

For research question 1, data was collected in pre and post domain test and interview to investigate the students' change in learning interest. Quantitative data analysis method with the usage of excel to analyze the data in pre-test and post-test. The pre and post domain test aim to measure learning interest in a 5-point Likert scale questionnaire. Also, students were interviewed randomly after the three lesson to investigate the students' change in learning interest. Then, comparing both groups of data to gain information for answering the research question.

For research question 2, data was collected in pre and post domain test and a self reflection question to investigate the difference in students' learning interest in Group A (Block-based group) and Group B (Text-based group). Quantitative data analysis method with the usage of excel to analyze the data in pre-test and post-test. The self-reflection includes an open-end question to measure the students' learning interest. The pre and post domain test is used to measure learning interest which is a 5-point Likert scale questionnaire. Also,

students were interviewed randomly after the three lesson to investigate the students' change in learning interest in both groups. Then, comparing the difference in two groups' data to gain information for answering the research question.

The third question examines the students' perceived learning difficulties. A survey used to measure students' perceived learning difficulties. Quantitative data analysis method by using excel will be adopted to analyze the data in survey. Furthermore, qualitative data analysis method will also be used, interview and self-reflection (an open-end question) after lessons to explore why Minecraft is difficult/easy. Students' interviews after the lessons will be conducted to investigate whether they feel challenging or manageable in Minecraft programming learning. Then, comparing these two groups' data will be for answering this question.

The fourth question investigates differences in computational thinking skills in two different programming interfaces. Data collection using pre and post test, which is a 5-point Likert scale questionnaires. The pre and post test is reference to "Computational Thinking scales (CTS)" adapted by Korkmaz et.al (2017) to measure computational thinking skills of secondary school students. The Computational Thinking scales consists of 29 items in five dimensions: creativity, algorithmic thinking, cooperation, critical thinking, and problem-solving. This study focus on computational thinking skill of problem-solving and creativity and evaluate students' computational thinking in 5-point Likert scale. The research question focused on two dimensions of Computational Thinking skills- creativity and problem-solving. Quantitative data analysis method using excel to analyze data in pre-test and post-test. For qualitative data analysis method, student interviews after the lessons will be implemented to discover whether students think they are improving computational thinking skills (creativity and problem-solving skills). Then, comparing these two groups' data will be for answering this question.

8. Research implementation

The research was conducted in Pentecostal Lam Hon Kwong School in Hong Kong in February to March 2022. The 24 Form 1-3 students were taking the STEM club as an extracurricular activity afterschool. The students were divided into two groups (Group A and Group B) randomly and each group had three 60-minutes Minecraft programming lessons, covering basic programming concepts, including command in agent movement, for loop and nested loop. Group A (Block-based group) took part in the 15:00-16:00 session, while Group B took part in the 16:15-17:15 session on the same day. All the students are beginners in programming. Both groups of students learnt the same programming content in Minecraft, however, Group A will use Block-based as an interface, while Group B will use Text-based as an interface. During the three lessons, students learnt programming concepts in Powerpoint, then the teacher assigned tasks for students to finish in Minecraft. In the first lesson of both Group A and Group B, the students spent 10 minutes answering pre-test questions on Google Form, then took 50 minutes to learn basic commands on the agent in Minecraft and finished 2 tasks. In the second lesson of both Group A and Group B, students learnt loops in Minecraft and finished 2 tasks. In the third lesson, the students spent 50 minutes learning nested loops in Minecraft and finished 2 tasks. In the last 10 minutes, students answered post-test questions on Google Form, then I randomly invited 3 students from each form to interview.

8.1 Teaching schedule

The three lessons are entry level programming lessons in the Minecraft education edition. The content of the lesson will include the following concepts: what is programming, how to program an agent(robot), how to write basic commands to control the agent(robot) using while loop and nested loop. In three lessons. Both groups will use Minecraft as a learning medium. For the Text-based group, I will use “Javascript” mode in Minecraft code builder

for teaching. For the Block-based, I will use “Block” mode in Minecraft code builder for teaching. PowerPoint will also be used as teaching materials in both groups for holding text or graphic teaching content. For example, the Block-based group’s PowerPoint displays a graphic of the while loop with text explanation to highlight the keys (Figure 1); while the Text-based group the code of the while loop is displaced with text highlighting the keys (Figure 2). Both lessons are of the same topics, but there are different programming interfaces.

Figure 1.An image with text explanation on Block-based group PowerPoint

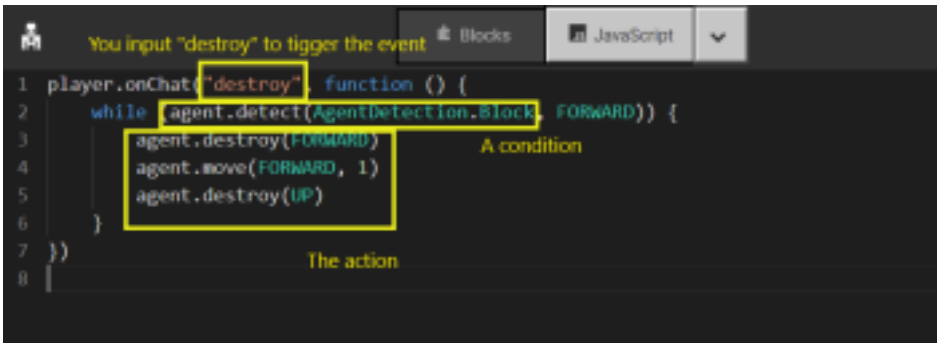

Figure 2. An image with text explanation on Text-based group PowerPoint

In each lesson, the teacher taught using PowerPoint and assigned programming tasks to students on Minecraft education edition. The students from Group A (Block-based group) and Group B (Text-based group) have the same teaching schedule and content (refer to Table 2).

Lesson number	Time	Teaching schedule and content
Lesson 1	10 minutes	Pre-test questionnaire and 3 open-ended self-reflection questions
	15 minutes	Teacher teach "What is programming" and the usage of chat command (programming tool in the game)
	15 minutes	Teacher teach and demonstrate move your agent forward by 5 units using chat command (programming tool in the game)
	20 minutes	**Task 1**-Move your agent move backward by 5 units using chat command
Lesson 2	20 minutes	Teacher demonstrate "Agent walk in Square"
	20 minutes	**Task 2**- "Agent walk in Square" using while loop to simplify the command
	20 minutes	**Task 3**-"Agent walk in Rectangle" using while loop
Lesson 3	20 minutes	Teacher teach nested loops and explain its benefits
	15 minutes	**Task 4**-Spawn Animals activity using nested loops
	15 minutes	**Task 5**-Plant carrots at the farm activity using nested loops
	10 minutes	Post-test questionnaire and 3 open-ended self-reflection questions

Table 2. Teaching schedule of Group A (Block-based group) and Group B (Text-based group)

9. Result and discussion

Q1. Does teaching programming using Minecraft increase the learning interest of students?

For the pre and post domain test, 10 students from Group A (Block-based group) and 10 students from Group B (Text-based group) attended both tests. Participants were Form 1

to 3 students. In Group A and Group B, both groups had 4 Form 1 students, 3 Form 2 students and 3 Form 3 students. None of the students studied any courses about programming before.

Table 3. Part A Pre-test and post results

	Group A (Block-based group) Pre-test result		Group B (Text-based group) Pre-test result		Group A (Block-based group) Post-test result		Group B (Text-based group) Post-test result	
	Mean	SD	Mean	SD	Mean	SD	Mean	SD
Q1	4.500	0.707	4.300	0.823	4.700	0.483	4.600	0.516
Q2	4.200	1.229	4.500	0.850	4.800	0.422	4.700	0.483
Q3	4.000	1.155	4.100	0.876	4.500	0.527	4.500	0.527
Q4	3.800	1.033	4.400	0.699	4.300	0.949	4.500	0.527
Q5	3.800	1.135	4.500	0.707	4.100	0.994	4.600	0.516
Q6	3.800	1.317	4.200	0.789	4.000	1.333	4.300	0.675
Q7	2.200	1.370	2.100	1.449	1.700	0.949	2.000	1.054

Part A Pre-test Q1 to Q7 are questions about students' interest in programming. In Q1 to Q6, the higher the score represents the higher the interest of students in programming before the Minecraft programming lesson. In Group A, the score is between 3.800 to 4.500, while the standard deviation is between 0.707 to 1.370. In group B, the score is between 4.100 to 4.500, while the standard deviation is between 0.707 to 1.447. In question 7, the question is "Do you think that programming is boring?)", the higher the score, the more boring the students thought about programming, the mean of both groups is 2.200 and 2.100 respectively. This shows that the Group A (Block-based group) and Group B (Text-based group) have a similar level of interest in programming. There is no significant difference between their interest in programming before the Minecraft programming lesson.

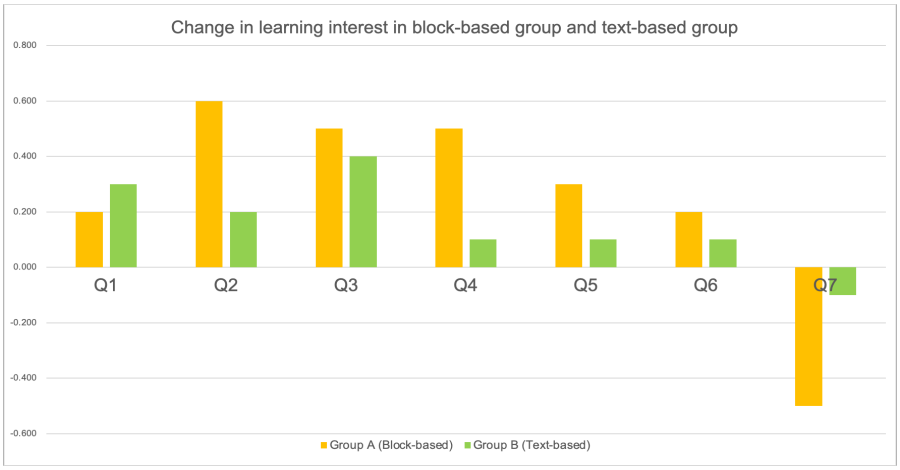

Figure 3. The change in learning interest in Block-based group and Text-based group

After the three lessons, students were required to finish the post-test for finding out their change in programming interest. For Part A Q1 to Q6, both Group A (Block-based group) and Group B (Text-based group) showed increased programming interest by 0.100-0.600 score respectively. In question 7, the question is "Do you think that programming is boring?)", the higher the score, the more boring the students thought about programming, both Group A and Group B showed a decrease in mean score, by 0.500 and 0.100 respectively. With reference to the change in learning interest in Group A and Group B (Refer to Figure 3) after the three lessons, the students' learning interest in programming increased in both Group A and Group B.

In the interview Q1, in Group A(Block-based group), all three students thought that learning programming by using Minecraft was interesting, because they could create a variety of products by themselves and there was not much geographic limitation in the game. In group B (Text-based group), two of the students are positive and one of the students is neutral about whether learning programming by using Minecraft is interesting. To conclude with, most of the students, no matter from Group A or Group B, agreed that learning programming by using Minecraft is interesting.

Group	Question	Interviewee			Quote
Group A (Block-based group)	Do you think that these three programming lessons are interesting? Why or why not?	Positive	Positive	Positive	"Yes. It's fun because the game is fun itself and I can use programming to build anything I wish." "Yes. I played Minecraft before and it's always very fun, especially when I can see different objects and creating something by myself."
Group B (Text-based group)		Positive	Neutral	Positive	Positive: "Yes. Minecraft has many variations and not much limitation in movement, which make it interesting and different from scratch." Neutral: "I think the game is okay, but it's quiet challenging and sometimes I cannot follow the instruction and it' hard."

Table 4. Interview result in Question 1

In the interview Q2, in Group A(Block-based group), all the three students believed that programming is more interesting than they thought in the past after the three lessons, because programming in Minecraft allowed more variations and creation of new items, which make them feel the programming learning is a game, unlike the programming lessons they learnt before. In Group B(Text-based group), two of the students believed that programming is more interesting than they thought in the past, because they enjoy programming with a game. While one of the students was neutral and did not agree that learning programming using Minecraft had much difference with other programming education tools. To conclude with, most of the students, from

both Group A or Group B, agreed that their interest in programming increased after the three lessons because the variation and game setting of Minecraft programming make students feel the learning is a game and thus increase their interest.

In conclusion, results from pre and post test and interviews showed that teaching programming using Minecraft increase the learning interest of students in both groups. The major reason is that the game Minecraft has variation in game setting, which sparked students interest. In addition, Minecraft programming make students feel the learning is a game, but not learning, and thus increase their interest.

Group	Question	Interviewee			Quote
Group A (Block-based group)	After three lessons, do you think that programming is more or less interesting than you thought in the past?	More interesting	More interesting	More interesting	"I think its more interesting because the setting is more variation and I feel like I am playing the Minecraft but not programming" "It's more interesting than Scratch and mbot because I can always make fancy items by using programming and enjoy playing the game"
Group B (Text-based group)		More interesting	More interesting	Neutral	Positive: "It's more interesting than before because Minecraft is one of my favourite game. I think its interesting game even with programming, because I can learn cooler movement using programming in Minecraft" Neutral: "I think its same in terms of interesting because the coding part is always hard in programming games"

Table 5. Interview result in Question 2

Q2. What are the differences in learning interest in Block-based programming teaching and Text-based programming teaching?

In the questionnaire Part A, most of the questions (i.e. Q2 to Q6) Group A (Block-based group) showed a larger increase in learning interest than Group B(Text-based group). In question 7, the question is “Do you think that programming is boring?)”, the lower the score, the less boring the students thought about programming than before, both Group A and Group B showed a decrease in mean score, by 0.500 and 0.100 respectively. By comparing these two scores, the 0.500 decrease in Group A showed a larger increase in learning interest than 0.100 decrease in Group B.

To conclude with, comparing the questionnaire results from two groups, Group A (Block-based group) showed a larger increase in learning interest than Group B (Text-based group).

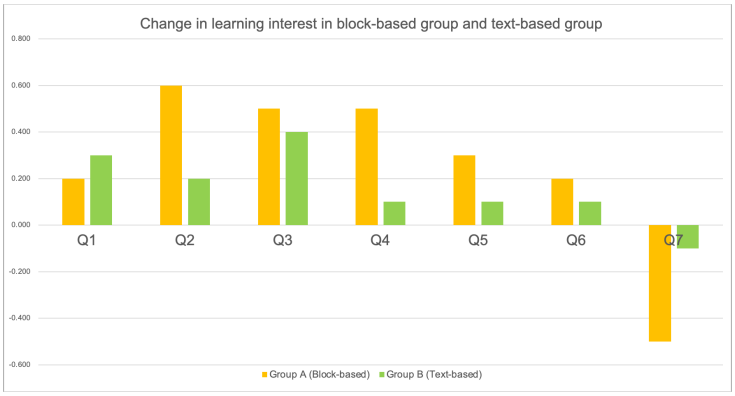

Figure. 4 The change in learning interest in Block-based group and Text-based group

In the self-reflection question Q1 (refer to Table 6), most of the students believed that Minecraft programming was interesting after the three lessons. Because the game setting is more variation and make the programming a game instead of programming lessons. While there were more students thought Minecraft programming was not interesting in Text-based group than the Block-based group. The reason Text-based group gave is that they thought Minecraft programming was challenging and they were unable to finish the tasks. These responses may stem from the more

challenging and low error tolerant features of Text-based programming interface (even a space would case error in Text-based programming interface). When students faced unsolved challenges and unable to finish the tasks, they may feel Minecraft programming is too challenging for them and deprived their interest in Minecraft programming.

To conclude with, Group A (Block-based group) showed a larger learning interest than Group B (Text-based group). Because students in Text-based group are more common to face unsolved challenges due to low error tolerant features of Text-based programming interface. Therefore, students feel Minecraft programming is too challenging for them and deprived their interest in Minecraft programming.

Table 6. Self-reflection question Q1 result

Group	Question	Answer		Quote
Group A (Block-based group)	Do you think Minecraft programming is interesting? Why?	Yes (8)	No (2)	Yes "I think its interesting because I can play Minecraft but not learning programming." "It's interesting and I wish to finish the task as soon as possible and do not want to stop even after the lesson, even though sometimes it is hard, it is still fun."
Group B (Text-based group)		Yes (6)	No (4)	Yes: "It's interesting because I can make different features and effects using programming in Minecraft." No: "I think it's challenging and hard to learn." "I think not very interesting because it is very hard to finish a task."

Q3. What are the impacts of Block-based programming teaching and Text-based programming teaching in learners' perceived learning difficulties?

For questionnaire Part B Q1 to Q5, the higher the score, the easier the tasks. The average score in the five questions for Group A (Block-based group) is 4.420 score and average score for Group B (Text-based group) is 3.640 marks, which illustrates that Group A group has a lower perceived learning difficulty than Group B. In Group A, the standard deviation is between 0.316 to 0.876 in the five questions. While the standard deviation in Group B is between 0.316 to 0.876 in the five questions. From the chart (refer to Figure.5), the perceived learning difficulty score is higher in Block-based group, which represent a lower perceived learning difficulty in Block-based group than Text-based group.

Table 7. Part B Pre-test and post results

	Group A (Block-based group) Post-test result		Group B (Text-based group) Post-test result	
	Mean	SD	Mean	SD
Q1	4.700	0.483	4.500	0.707
Q2	4.900	0.316	4.200	0.632
Q3	4.600	0.516	3.900	0.876
Q4	3.900	0.876	2.800	0.632
Q5	3.900	0.738	2.800	0.789
Average	4.420		3.640	

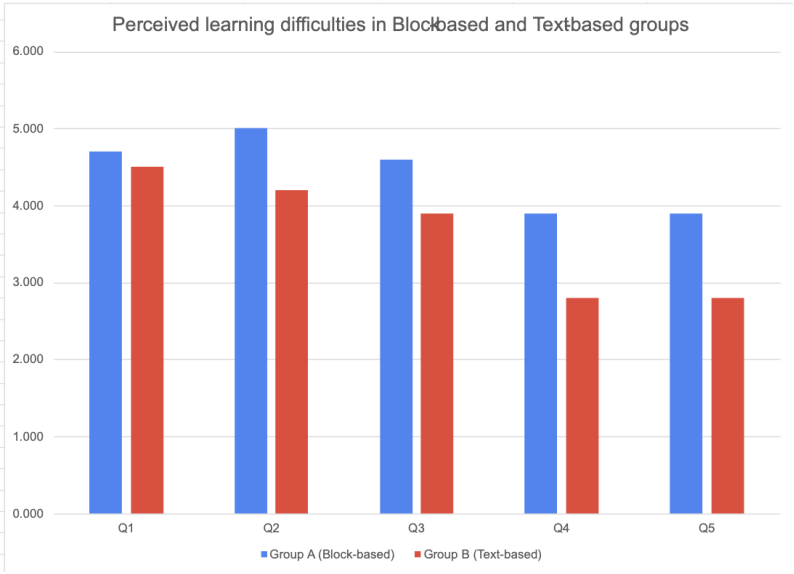

Figure 5. Perceived learning difficulties in Block-based and Text-based groups

In the self-reflection question Q2 (refer to Table. 8), eight students in Group A(Block-based group) believed that programming is easy because they thought programming using dragging block commands is easier than text programming. While two students from Group A believe that programming is hard because creating programming command is challenging. In Group B(Text-based group) five students believed that programming is hard because they usually make errors and cannot run the program, while five students believe that it is easy because they knew how to write command. From the result, students' perceived learning difficulty in Block-based group is lower than Text-based group, one of the major reasons is that Block-based learners drag the command boxes and are less frequent to make mistakes and errors, which may cause less frustration and a lower perceived learning difficulty. In contrast, Text-based group needs to input text command, which is case sensitive and more complex in syntax (include more rules). Then, students are easy to make mistakes and may cause frustration and a higher perceived learning difficulty.

To conclude with, the perceived learning difficulty is lower in Group A(Block-based group) than in Group B(Text-based group). The main reason is that Block-based group drag the command

box which has a lower chance of making error than Text-based group. Thus, resulting in a lower frustration and lower perceived learning difficulty.

Table 8. Self-reflection question Q2 result

Group	Question	Answer		Quote
Group A (Block-based group)	Do you think Minecraft programming tasks are difficult to complete? Why?	Yes (2)	No (8)	Yes: "Programming is hard because you it is difficult to think of command" No: "Programming is easy because you just need to drag the boxes" "Minecraft programming is easy because I do not need to type"
Group B (Text-based group)		Yes (5)	No (5)	Yes: "Programming is quite hard because it's easy to make mistakes and need to test again and again and this makes it so hard." "Programming is not easy because it's easy to have errors" No: "Programming is easy because I know how to write the codes" "Programming is easy because I can finish all tasks"

Q4. What is/are the changes in computational thinking skills in Block-based programming learners and Text-based programming learners?

In questionnaire Part C Q1 to Q3(refer to figure 6), the higher the score represents the larger increase in computational thinking skills. In Group A (Block-based group) and Group B (Text-based group) students both showed an increase in computational thinking skills (problem-solving skills and creativity).

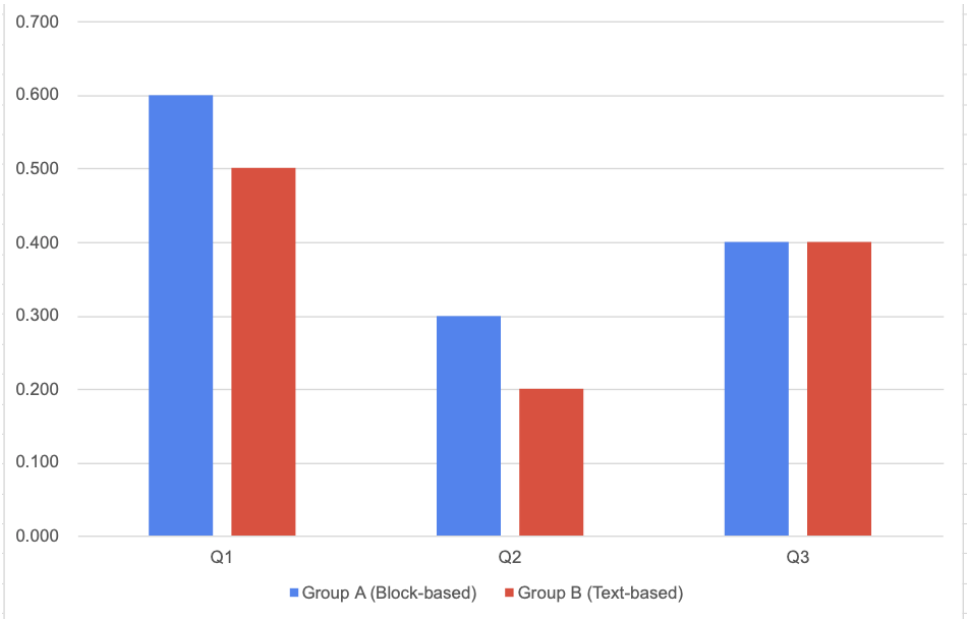

Figure 6. The changes in computational thinking skills in Block-based and Text-based groups Q1-Q3

In questionnaire Part C Q4 to Q8 (refer to figure 7), the larger the decrease in score represents the larger increase in computational thinking skills. In Group A (Block-based group) and Group B (Text-based group) students both showed a decrease in score by 0.100 to 0.700 respectively, this showed the improve in computational thinking skills (problem-solving skills and creativity) in both Group A and Group B.

To conclude with, students from both Group A (Block-based group) and Group B (Text-based group) had improvement in their computational thinking skills (problem-solving skills and creativity) after the three lessons.

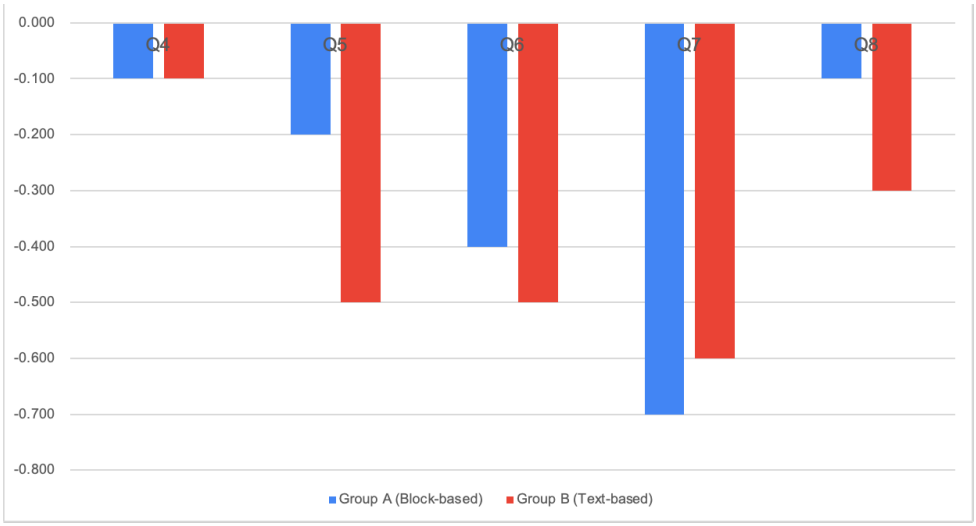

Figure 7. The changes in computational thinking skills in Block-based and Text-based groups Q4-Q8

In Interview(refer to Table 9 and 10), most of the students think that learning programming by using Minecraft improves their problem-solving skills and creativity (computational thinking skills). In interview question 4, all the students from Group A and Group B agreed that they become more creative after the three lessons. The main reason is that students thought they kept generating different ideas during solving the Minecraft programming tasks. Therefore, they trained their creativity to solve programming problems and became more creative. The other reason is that programming does not have a fixed answer, therefore students are free to attempt different ways to solve problems, which make them feel being more creative after the three lessons. In interview question 5, 2 students said he/she always break down the Minecraft tasks to smaller parts to help himself to code, which demonstrated his problem-solving skills is being exercised during the lessons. Three of the interviewees said that they are attracted by the game and wish to program the command as soon as possible, therefore encouraging them to divide the problems into small tasks, then create ways to solve the problem. For example by watching YouTube videos about Minecraft programming, then learn the skills from the videos to help them to solve the problem. To conclude with, Minecraft train students' computational thinking skills, including problem-solving skills and creativity, by encouraging students to generate ideas, free exploration of solutions, and the

attractiveness of the game encourage students to break down tasks to find solution.

Table 9. Interview question Q4 result

Group	Question	Interviewee			Quote
Group A (Block-based group)	After three lessons, do you think you become more creative? (i.e. able to create ideas to seek solution of a problem.)	Positive	Positive	Positive	"I think I become creative because I always want to find ways to solve the problem and keep thinking solutions. Also there can be many ways to solve a problem, so when I find a different way or see people think of another way, I feel I become more creative" "I always urge myself to think faster and complete the task, so my creativity increase when generating more ideas. I also feel I become more creative when I could code something by myself."
Group B (Text-based group)		Positive	Positive	Positive	Positive: "I think I become more creative because I would discuss with my friends and get inspired by them in how to input the command. I think both of us become more creative" "I think I become more creative because I would like to find ways to solve problems"

Table 10. Interview question Q5 result

Group	Question	Interviewee	Quote

Group A (Block-based group)	After three lessons, do you think your problem-solving skills have improved? (i.e. able to use variables such as X and Y in the solution of a problem.)	Positive	Positive	Positive	"I think problem-solving skills have improved because I always want to break down the coding command to smaller steps in order to create an object in Minecraft" "I always urge myself to think faster and solve the problem, so I divided a task to many sub tasks and solve them one by one. Therefore I think I improved my problem-solving skills"
Group B (Text-based group)		Positive	Positive	Positive	Positive: "I think my problem-solving skills improved because when I have difficulties in Minecraft programming, I really want to solve it now and watch YouTube videos to act as a reference. Then solve it step by step by making a huge tasks smaller chunks." Neutral: "I think my problem-solving skills is same because sometimes the tasks are too hard so I just copy the suggested answer"

10. Limitation

For the limitation of the research project, this research was not comprehensive enough because only 10 students participated in the project. Therefore, the deviation of the collected data was large because some of the interviewee might be the extreme case.

In addition, the study is conducted through Zoom meetings, which make it hard for the teacher to

identify and correct errors in students' programming codes, especially for the Text-based group (even an extra space may cause error), possibly make students feel hard to fix problems in programming and perceived programming difficulty is harder.

Thus, some students are using an Ipad to perform Minecraft programming instead of a computer, the differences in setting in the game make them difficult to follow the teacher's instruction and perceived programming difficulty is harder than others.

11. Conclusion and suggestion

In conclusion, the study applied a mixed research method to analyze programming interest, perceived programming difficulty and computational thinking skills in Block-based and Text-based programming learning in Minecraft. After the analysis of collected data, it shows that using Minecraft as an educational tool in learning programming may cause positive effects on students' learning interest. Also, perceived learning difficulties in Block-based programming is lower than Text-based programming, possibly because Block-based programming is less common to make programming errors. In addition, students' computational thinking skills can be improved in both Block-based and Text-based programming learners. Past research about using Minecraft for teaching programming to university students (Zorn et.al ,2015), it is not focused on secondary education.

After the present research, it shows that Minecraft programming is also suitable for secondary students in Hong Kong in increasing students' learning interest and Block-based programming learning has more benefit in increasing sharper learning interest and lower perceived learning difficulty, which may also be a better programming learning interface than Text-based programming learning. In this research, Minecraft provide a good platform for students to learn programming concepts (for example, loops) through game-based learning approach. However, further research could be conducted to understand how students perform or their learning interest

in learning more complex computational concepts, such as variables and operators in Minecraft. Also, what are the changes and difference in student's learning interest and outcome the two programming interface when learning more complex computational concepts on Minecraft.

12. Reference list:

Prensky, M. (2003). Digital game-based learning. Computers in Entertainment (CIE), 1(1), 21-21.

Ritterfeld, U., & Weber, R. (2006). Video games for entertainment and education. *Playing video games: Motives, responses, and consequences*, 399-413.

Loyens, S. M., & Gijbels, D. (2008). Understanding the effects of constructivist learning environments: Introducing a multi-directional approach. *Instructional science*, *36*(5), 351-357.

Papastergiou, M. (2009). Digital game-based learning in high school computer science education: Impact on educational effectiveness and student motivation. Computers & Education, 52(1), 1-12.

Choi, J., Mogami, T., & Medalia, A. (2010). Intrinsic motivation inventory: an adapted measure for schizophrenia research. *Schizophrenia bulletin*, *36*(5), 966-976.

Leng, E. Y., Baki, R., & Mahmud, R. (2010). Stability of the Intrinsic Motivation Inventory (IMI) for the use of Malaysian form one students in ICT literacy class. EURASIA Journal of Mathematics, Science and Technology Education, 6(3), 215-226. ISO 690

Huang, W. H. (2011). Evaluating learners' motivational and cognitive processing in an online game-based learning environment. Computers in Human Behavior, 27(2), 694-704.

McGonigal, J. (2011). *Reality is broken: Why games make us better and how they can change the world*. Penguin.

Qu, S. Q., & Dumay, J. (2011). The qualitative research interview. *Qualitative research in accounting & management*.

Marsden, E., & Torgerson, C. J. (2012). Single group, pre-and post-test research designs: Some. methodological concerns. *Oxford Review of Education*, *38*(5), 583-616.

Barzilai, S., & Blau, I. (2014). Scaffolding game-based learning: Impact on learning achievements, perceived learning, and game experiences. *Computers & Education*, *70*, 65-79.

Zorn, C., Wingrave, C. A., Charbonneau, E., & LaViola Jr, J. J. (2013). Exploring Minecraft as a conduit for increasing interest in programming. In FDG (pp. 352-359).

Callaghan, N. (2016). Investigating the role of Minecraft in educational learning environments. *Educational Media International*, *53*(4), 244-260.

Pusey, M., & Pusey, G. (2016). Using Minecraft in the science classroom. *International Journal of Innovation in Science and Mathematics Education*, *23*(3).

The Education Bureau, Hong Kong SAR Government (EDB). (2016). Report on STEM Education - Education Bureau. Retrieved from https://www.edb.gov.hk/attachment/en/curriculum development/renewal/STEM%20Education%20Report_Eng.pdf

Hiltunen, T. (2016). Learning and teaching programming skills in Finnish Primary Schools–The potential of games. University of Oulu, Oulu.

Nebel, S., Schneider, S., & Rey, G. D. (2016). Mining learning and crafting scientific experiments: a literature review on the use of minecraft in education and research. Journal of Educational Technology & Society, 19(2), 355-366.

Garris, R., Ahlers, R., & Driskell, J. E. (2017). Games, motivation, and learning: A research and practice model. In Simulation in Aviation Training (pp. 475-501). Routledge.

Bau, D., Gray, J., Kelleher, C., Sheldon, J., & Turbak, F. (2017). Learnable programming: blocks and beyond. Communications of the ACM, 60(6), 72-80.

Korkmaz, Ö ., Ç akir, R., & Ö zden, M. Y. (2017). A validity and reliability study of the computational thinking scales (CTS). *Computers in human behavior*, *72*, 558-569.

Weintrop, D., & Wilensky, U. (2017). Comparing block-based and Text-based programming in high school computer science classrooms. ACM Transactions on Computing Education (TOCE), 18(1), 1-25.

Pratama, L. D., & Setyaningrum, W. (2018). Game-Based Learning: The effects on student cognitive and affective aspects. In Journal of Physics: Conference Series (Vol. 1097, No. 1, p. 012123). IOP Publishing.

Kutay, E., & Oner, D. (2022). Coding with Minecraft: The development of middle school students'. computational thinking. *ACM Transactions on Computing Education (TOCE)*, *22*(2), 1-19.

13. Appendix 1- Google form of pre and post-test

Pre-test & Post-test

s1124984@s.eduhk.hk (未分享) 切換帳戶

Part A) Please answer the following questions.

1. I enjoyed the game.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

2. I had fun playing the game.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

3. Playing the game was pleasant.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

4. I like writing computer programs.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

5. Programming is enjoyable and stimulating.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

6. Once I start trying to work on a program, I find it hard to stop.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

7. Programming is boring.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

Self-reflection question

Q1. Do you think Minecraft programming is interesting? Why?

恐的含桑

Part B) Do you think these programming topic is/ are difficult?

1. Move your agent move backward by using chat command.

	1	2	3	4	5	
Very difficult	○	○	○	○	○	Very easy

2. Move your agent walk by square by using chat command.

	1	2	3	4	5	
Very difficult	○	○	○	○	○	Very easy

3. Move your agent walk by rectangle using chat command.

	1	2	3	4	5	
Very difficult	○	○	○	○	○	Very easy

4. Spawn Animals using "repeat" block.

	1	2	3	4	5	
Very difficult	○	○	○	○	○	Very easy

5. Plant carrots at the farm using nested loops.

	1	2	3	4	5	
Very difficult	○	○	○	○	○	Very easy

Self-reflection question

Q2. Do you think Minecraft programming tasks are difficult to complete? Why?

恐的含桑

Part C) Please answer the following questions.

1. I like the people who are sure of most of their decisions.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

2. I have a belief that I can solve the problems possible to occur when I encounter with a new situation.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

3. I trust my intuitions and feelings of “trueness” and “wrongness” when I approach the solution of a problem.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

4. I have problems in the demonstration of the solution of a problem in my mind.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

5. I have problems in the issue of where and how I should use the variables such as X and Y in the solution of a problem.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

6. I cannot apply the solution ways I plan respectively and gradually.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

7. I cannot produce so many options while thinking of the possible solution ways regarding a problem.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

8. I cannot develop my own ideas in the environment of cooperative learning.

	1	2	3	4	5	
Strongly disagree	○	○	○	○	○	Strongly agree

Self-reflection question

Q3. After the three Minecraft programming lessons, do you think that you become more creative? Why?

Appendix 2- Interview questions

Question No.	Question	Response to research topic
1	Do you think that these three programming lessons are interesting? Why or why not?	interest
2	After three lessons, do you think that programming is more or less interesting than you thought in the past?	interest
3	Do you think that the content in these programming lessons are difficult? Why or why not?	perceived programming difficulty
4	After three lessons, do you think you become more creative? (i.e. able to create ideas to seek solution of a problem.)	computational thinking skills
5	After three lessons, do you think your problem-solving skills have improved? (i.e. able to use variables such as X and Y in the solution of a problem.)	computational thinking skills