

## Effects of Annotations and Homework on Learning Achievement: An Empirical Study of Scratch Programming Pedagogy

Addison Y. S. Su<sup>1,2\*</sup>, Chester S. J. Huang<sup>2</sup>, Stephen J. H. Yang<sup>2</sup>, T. J. Ding<sup>3</sup> and Y. Z. Hsieh<sup>4</sup>

<sup>1</sup>Research Center for Advanced Science and Technology, National Central University, Taiwan // <sup>2</sup>Department of Computer Science and Information Engineering, National Central University, Taiwan // <sup>3</sup>Department of Materials and Energy Engineering, Ming Dao University, Taiwan // <sup>4</sup>Department of Management and Information Technology, Southern Taiwan University of Science and Technology, Taiwan // ncuaddison@gmail.com // shinjia.huang@gmail.com // jhyang@csie.ncu.edu.tw // tjding@mdu.edu.tw // yizeng.hsieh@gmail.com

\*Corresponding author

(Submitted July 20, 2014; Revised November 25, 2014; Accepted February 1, 2015)

### ABSTRACT

In Taiwan elementary schools, Scratch programming has been taught for more than four years. Previous studies have shown that personal annotations is a useful learning method that improve learning performance. An annotation-based Scratch programming (ASP) system provides for the creation, share, and review of annotations and homework solutions in the interface of Scratch programming. In addition, we combine the ASP system with the problem solving-based teaching approach in Scratch programming pedagogy, which boosts cognition development and enhances learning achievements. This study is aimed at exploring the effects of annotations and homework on learning achievement. A quasi-experimental method was used with elementary school students in a Scratch programming course over a 4-month period. The experimental results revealed that students' thoughts and solutions in solving homework assignments have a significant influence on learning achievement. We further investigated that only making annotations in solving homework activities, among all other variables (the quantity of annotations, the quantity of one's own annotations reviewed, the quantity of peers' annotations reviewed, the quantity of one's own homework solutions reviewed, and the quantity of peers' homework solutions reviewed), can significantly predict learning achievements.

### Keywords

Annotation, Scratch programming, Learning performance, Homework

### Introduction

Students engage in an especially useful practice by making annotations to learning materials. The positive effects of making and sharing annotations on learning Scratch programming have been presented in the study of Su, Yang, Hwang, Huang and Tern (2014). The use of making annotations has a positive influence on learning achievement and it stimulates students' motivation to learn (Su, Yang, Hwang, & Zhang, 2010). Hsiao and Brusilovsky (2010) thought that annotations could support community feedback in the process of learning Lego. Garner (2009) indicated that Scratch programming is akin to Lego, which is ideal for primary school students.

We developed an annotation-based Scratch programming system, ASP, which allowed students to highlight important points and key concepts. Furthermore, it supports the storage of making, sharing, and reviewing annotations, so students can subsequently find their previous ideas and thoughts. The mechanism provides opportunities for students to critically recall past memories and to think about how to use them. Making and sharing annotations can expand students' ideas and thoughts, which may lead to better solutions to solving problems. However, students face various problems in learning Scratch, and invariably students try to find particular ways to solve homework problems. To learn problem-solving-based learning methods is a study in learning itself. The incorporation of ASP, within the context of problem-solving-based teaching, enables students to become aware of and to determine their problem-solving skills and Scratch learning needs (Su, Yang, Hwang, Huang, & Tern, 2014). The innovative approach also allows students to be able to enhance their learning potential in the area of Scratch programming. All the issues described above have been explored in this paper by conducting an empirical study of the Scratch programming class using ASP tool-supported problem-solving-based teaching approach.

## Literature review

### Annotation systems supported by Scratch programming pedagogy

The objective of Scratch programming pedagogy is to encourage students to think about their own practical experience. Garner (2009) and believe that Scratch programming should have a “low floor” (easy to get started) and a “high ceiling” (opportunities to create increasingly complex projects over time). Thus, we conducted an empirical study of Scratch programming pedagogy at a Taiwanese elementary school, and students could easily understand the Chinese interface of Scratch programming when first introduced to Logo programming.

A variety of annotation systems have been developed for helping teachers and students to teach and learn digital learning materials. Microsoft has published OneNote that is a personalizable annotation tool. Teachers and students can use a digital pen, a keyboard, or a multimedia recording to input annotations. However, the system annotations can only annotate Microsoft documents and be shared through e-mails. Yang, Chen, and Shao (2004) proposed a personalized annotation management system, and they have demonstrated that personalized annotation enhances knowledge sharing and learning in students' online learning activity. Hwang, Wang, and Sharples (2007) developed a multimedia annotation system that enables students to annotate multimedia objects as well as create multimedia content, such as pictures, audio and video as annotations. Yang, Zhang, Su, and Tsai (2011) proposed a collaborative multimedia annotation system that allows users of audio and video to make and share personal and group annotations that are synchronized with the multimedia. The system permits sharing of personal knowledge and collaboration among learners; annotations can be accessed by other users remotely through the Internet.

Following the previous studies, we found that existing annotation systems do not support the Chinese interface of Scratch programming, and users cannot make, review, and share annotations directly inside the Scratch programming editing environment. In this study, we modified existing annotation systems to provide for creating, reviewing, and sharing annotations in the interface of Scratch programming. The annotation system supported Scratch programming pedagogy is defined as the process wherein students make, review, and share annotations so they can subsequently review their previous lecture notes and homework solutions. This mechanism provides opportunities for students to critically re-evaluate and possibly improve their thoughts and solutions. Finally, we aimed to use the ASP system supported problem solving-based teaching approach in learning Scratch programming.

### Influence of annotations and homework on learning achievement

Slotte and Lonka (1999) indicated that students who created annotations generally attained greater levels of academic achievements than those who did not; students understood the learning material deeper and they performed better on a test. Yeh and Lo (2009) conducted an experiment for giving feedback on second language writing of college freshmen by creating, sharing and reviewing annotations in homework activities. Moreover, the result showed that students who were arranged in an annotation tool group had significantly better error recognition learning performance than those who were arranged in a traditional paper-based annotation group.

As for annotations reviewed by other peers, Slotte and Lonka (1999) emphasized that reviewing annotations have the advantage of allowing for informal knowledge sharing related to a concept. Nokelainen, Miettinen, Kurhila, Florén, and Tirri (2005) thought that reviewing annotations allows learners to collaborate by sharing their thoughts about the learning material with other students accessing the same content. Reviewing homework solutions in this study is defined as the process when students reviewed peers' homework. By reviewing annotations and homework of other peers, students may benefit from peer learning via critical peer feedback and reflection.

As for homework, Power, Dombrowski, Watkins, Mautone, and Eagle (2007) reported that homework in elementary school may be completed in school and at home after school. Power et al. (2007) argued that homework has positive effect because homework has an immediate effect on the retention and understanding of the material it covers. Based on previous studies, homework is not only helpful for enhancing individual learning, but it is a useful way to facilitate peer learning through sharing annotation progresses.

In this study, we applied the developed ASP system to support students with making, reviewing, and sharing annotations and solving homework during learning Scratch programming activities. This study aimed to explore the

effects of creating annotations and homework solutions, reviewing own and peers' annotations, reviewing own and peers' homework solutions on learning achievement.

### **Problem-solving method supported Scratch programming pedagogy**

Papert (1980) indicated that students need to possess a suitable problem-solving method, in which people try to make sense of a problematic situation using previous knowledge and practical experience by attempting to acquire relevant information and solutions about that situation until they can resolve the ambiguity. The problem-solving-based instructional method plays an important role in helping students learn better; it helps understand how the task is performed (Hung, 2008). The empirical studies of existing problem-solving-based instructional methods include at least four distinct steps: (1) understanding and defining the problem; (2) planning solutions and collecting necessary information; (3) designing and implementing the solution; and (4) verifying and presenting results.

Hung (2008) have contended that existing problem-solving-based instructional methods need to correspond with school children in terms of Scratch programming studies. According to existing problem-solving-based teaching methods, we refined a way to accomplish the objective in four distinct steps. Step 1: Explain the objective of an example. Step 2: Understand the requirement of the example. Step 3: Propose a plan to solve the design of the example. Step 4: Implement the example and debug it. Adherence to these steps strongly promoted students' problem-solving skills. Hung (2008) thought that an appropriate selection of problem-solving instructional strategies plays an important role in helping younger schoolchildren learn; it helps to understand how the programming task is performed. The appropriate method could be used to help schoolchildren establish their learning objectives and assess their progress toward these objectives.

In this study, we sought to facilitate problem-solving abilities and cognitive developments among students by providing a means to which students could create, share, and review annotations and work on homework. We focused on study cognitive skills that would show when students review their annotations and homework solutions using the ASP-supported problem-solving-based teaching approach.

### **Perceived satisfaction of using ASP-supported problem-solving-based teaching approach**

Perceived usefulness and perceived ease-of-use proposed by Davis (1986) have been widely used to predict user attitudes toward information technology. A learner's behaviour toward an information system was determined by his/her attitude concerning the perceived usefulness and perceived ease-of-use. Perceived usefulness and perceived ease-of-use were evaluated with the initial version of ASP (Yang et al., 2011; Cheng, Lou, Kuo, & Shih, 2013), and our results showed that students perceived the ASP system to be both useful and easy to use. Perceived usefulness refers to the belief that using ASP will boost one's performance. Perceived ease-of-use refers to the belief that using ASP will be relatively effortless. In this study, a questionnaire, adapted from Hwang, Wang, and Sharples (2007), was employed to gain an understanding of the students' perceived satisfaction with using ASP-supported problem-solving-based teaching approach.

### **Research objectives**

The initial version of ASP was a web-based annotation system and it was developed by Su, Yang, Hwang, and Zhang (2010). The subsequent version was modified into a multimedia-based annotation system in the following years (Su et al., 2014). The experiment was conducted with the ASP-supported problem-solving-based teaching approach, and the results show that students perceived the system to be both useful and easy to use. Moreover, the students were satisfied with using ASP-supported problem-solving-based teaching approach in individual and collaborative learning scenarios. The learning achievements of students in the experimental group (using ASP-supported problem-solving-based teaching approach) were significantly better than those of students in the control group (using traditional pen and paper) in the two learning scenarios; our results revealed that students were satisfied with using ASP in two learning scenarios.

Based on the previous studies, we conducted a real study (e.g., one Scratch programming course) at a primary school in order to explore the effects of annotations and homework on learning performance. At the end of this course, the questionnaire was given to evaluate the students' attitudes -- their perceived satisfaction with using ASP-supported problem-solving-based teaching approach.

### The operation of the Annotation-based Scratch Programming (ASP) system

To efficiently conduct the ASP-supported problem-solving-based instructional approach, four distinct steps are given to students to understand how their homework is solved. The homework example with the topic "A Sprite Walking Staircase" was uploaded to the ASP and made available to students.

Step 1: Explain the goal of a sprite walking staircase, as shown in Figure 1, the teacher reminds students to recall and employ their past knowledge experience in performing the sprite walking staircase.

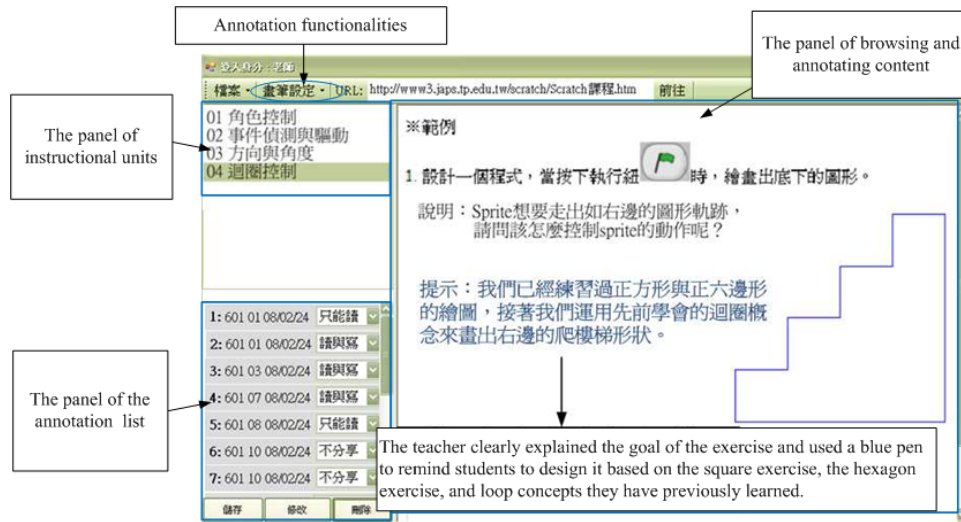


Figure 1. Explaining a sprite walking staircase programming exercise

Step 2: Understand the input and output requirements of a sprite walking staircase, as shown in Figure 2, the teacher made notes in red to provide clues regarding the requirements of the sprite walking staircase.

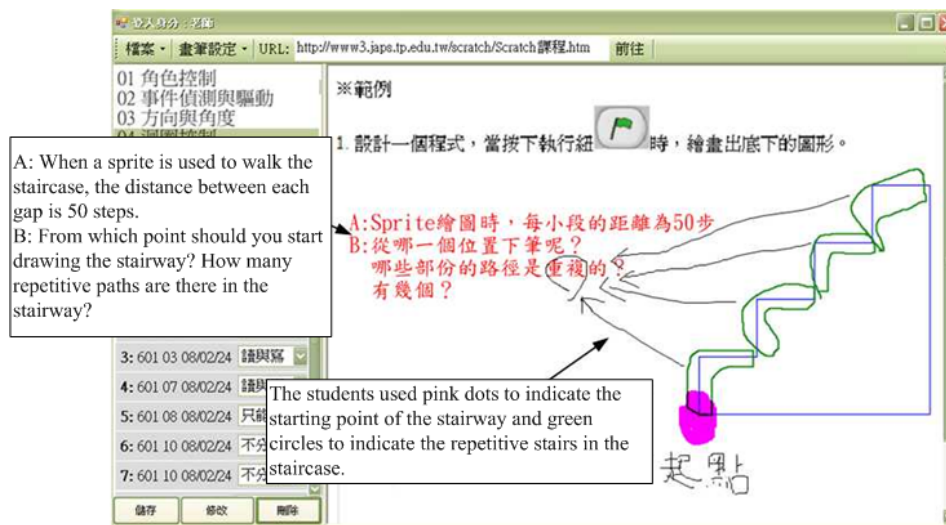


Figure 2. Demonstrating a student's understanding requirements of the sprite walking staircase

Step 3: Collect necessary information and propose a plan to finish the design of a sprite walking staircase, as shown in Figure 3, students need to understand how loop commands are interpreted by Scratch programming with operations in order to accomplish the repetitive stairs in the example. Furthermore, students indicate the starting point and the repetitive parts of the stairway in light of the demonstrative teaching of the teacher.

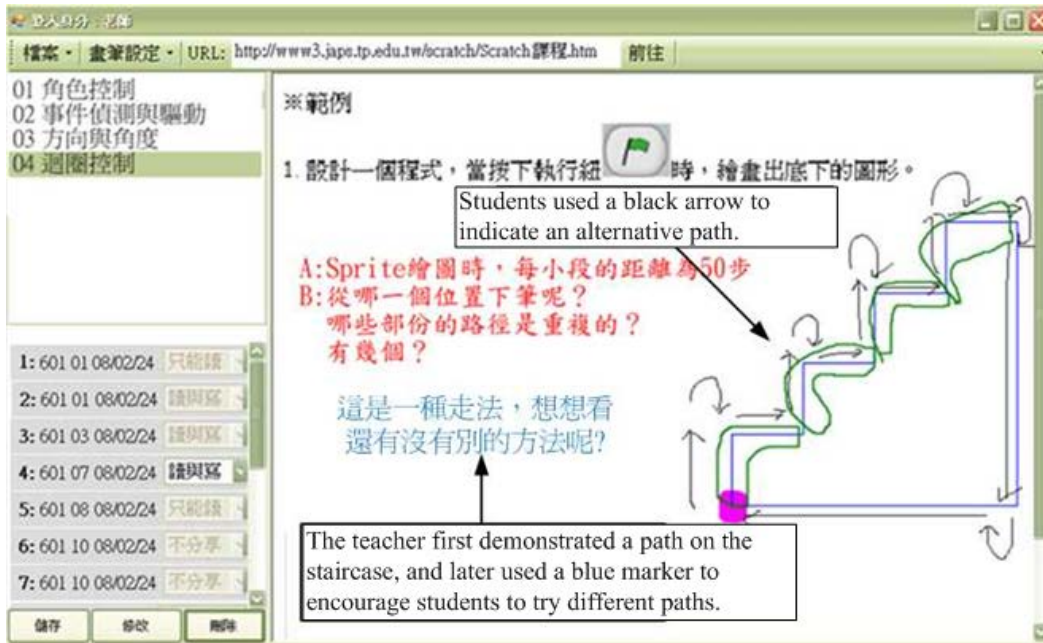


Figure 3. Showing students' proposing a solution plan for designing the sprite walking staircase

Step 4: Design a sprite walking staircase and debug it, as shown in Figure 4, the teacher answers a student's question in red ink, such as indicating the source of Scratch programming errors in the example.

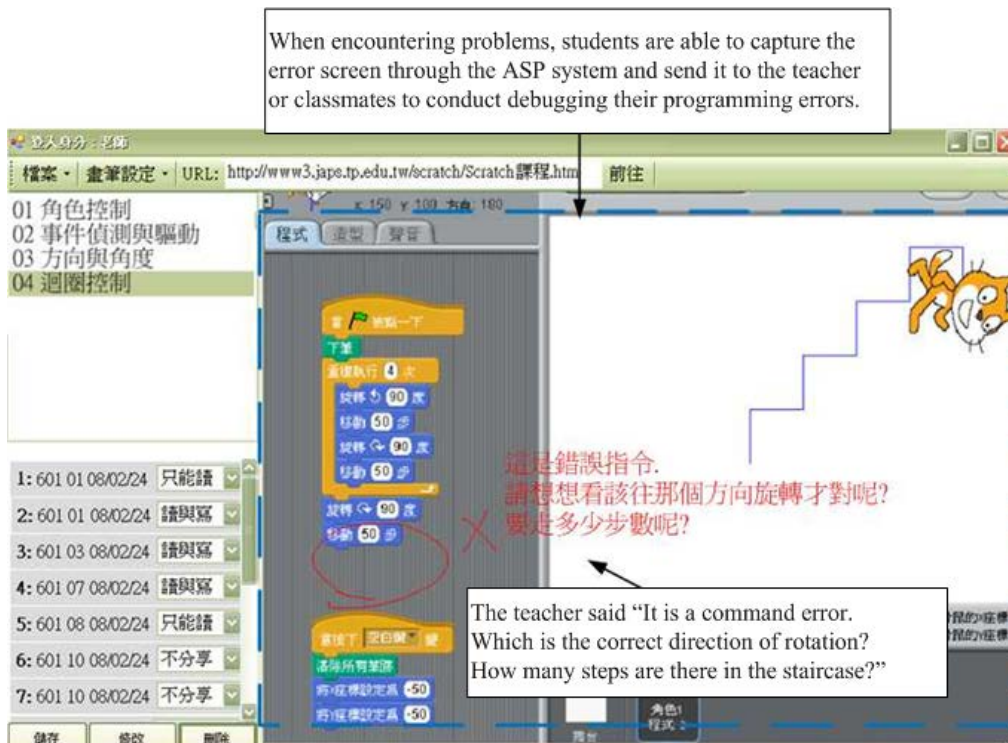


Figure 4. Students' writing the sprite walking staircase and debugging it

The ASP provided straightforward access control over annotations that were made and retrieved. For example, a student may be granted read and write privileges to annotations. This access control mechanism provided multiple choices for sharing and reviewing annotations with others; students were able to grant or deny access rights to their peers. In addition, the ASP automatically saved manipulation portfolio data created by all students to our database, which included the students' comments and thoughts, the students' answers to each question, the time duration for each operation, and so on. By exploring students' manipulation of portfolio data, we were able to analyse the detailed students' learning behaviours while studying Scratch programming.

## **Methodology**

### **Participants**

Thirty-seven students (the average age was 12 years old) from one sixth-grade class of a northern Taiwanese elementary school participated in this experiment. Most of students understood basic computer concepts, but they had no experience in Scratch programming. The teacher, with 12 years of teaching experience, led a course entitled "Scratch Programming." Chinese was used as a medium of instruction. The course ran three hours per week in a computer classroom for four months, from March to June 2013.

### **Instruments**

#### *Materials*

The course covers the Introduction to Scratch programming for primary school students. The teacher prepared two topics of Scratch programming in the following sequence. The first topic included event triggers and variables using the block palette combined with the Sensing and Pen functions in Scratch programming to create a hexagon shape. The second topic included conditionals and loops combined with the Control function in Scratch programming to create staircase and windmill shapes. Due to teaching appointment periods and limitations, the teacher focused on two topics. The teacher designed the instrument used to serve as a Scratch programming achievement test.

#### *Scratch programming achievement test*

The test aimed to evaluate the learning achievements of the students at the end of the semester. It consisted of 6 multiple choice-items (10 points each) and 2 problem-solving implementation items (20 points each). The multiple choice-items described the structure, syntax, and implementation concept of Scratch programming, and the correctness of the answer to each problem-solving implementation item was assessed by the teacher. If the item is completely correct, 20 points was given for the correctness criterion. If the item was partially correct, 2 points was deducted per "bug." At last, if the item could not be executed due to syntax errors, zero points were given. According to Brown (1996) and Su et al. (2014), Kuder Richardson 20 (KR-20) was used to evaluate the internal consistency reliability of the test. The KR-20 coefficient of the test was 0.72, showing adequate internal consistency in evaluating the students' learning performance.

#### *Questionnaire*

The questionnaire mainly surveyed students' perceived satisfaction of using ASP-supported problem-solving-based teaching approach. During the process of designing a valid questionnaire, applicable items were verified and validated by the two education experts. Some ambiguous or unsuitable items were modified, alternated, or removed accordingly. The questionnaire was composed of response questions with seven items in a 5-point Likert and one open-ended question. According to the reliability evaluation of the questionnaire, the Cronbach  $\alpha$  coefficient was 0.83, implying that the questionnaire had a high reliability (Carmines & Zeller, 1979). Thirty-seven students were asked to complete the questionnaire after the post-test.

## Procedures

During learning Scratch programming, all young students were asked to make annotations, to solve homework exercises and to review annotations and solutions. School children were volunteered and curious to use the ASP system in the context of learning Scratch programming because it was easy to use, with friendly user interface and many students considered that using annotations were very interesting. The teacher was motivated to use the ASP-supported problem-solving-based teaching approach to teach Scratch programming.

A quasi-experimental method was used with elementary school students in a Scratch programming course over a 4-month period, and the experimental procedures were conducted as follows. Students were asked to study the learning material unit-by-unit and to make annotations on the content. Each unit of learning materials consisted of Scratch programming homework exercises, such as a hexagon shape, a staircase and a windmill shape. Students were asked to complete and solve their homework problems after the teacher taught a unit; they could do it immediately during class. Students were then asked to review their own annotations and homework solutions. Furthermore, the students were asked to share their annotations and homework solutions with other students and review the annotations and homework solutions of their peers.

At the end of the semester, a post-test was administered to evaluate students' learning performance. After the post-test, the questionnaire was given to understand students' perceived satisfaction of using ASP-supported problem-solving-based teaching approach.

## Data collection and analysis

We collected data to response to the research objectives. The data comprised students' manipulation portfolio data, post-test scores, and questionnaire data. In class, the students' manipulation portfolio data was stored in our server database. The students' manipulation portfolio data provided statistical information for each student on the number of annotations made, the number of one's own annotations reviewed, the number of peers' annotations reviewed, the number of homework solutions made, the number of one's own homework solutions reviewed, and the number of peers' homework solutions reviewed. The statistical information helped us to understand annotation and homework contributions of each student. Two education experts were involved in content analysis to perform coding for the number of annotations and homework solutions. Content analysis resulted in an inter-rater reliability ( $k = 0.76$ ) of more than 0.7, and the analysis differences were resolved by the two experts through discussion. At the end of the semester, a post-test was conducted to obtain students' learning performance. Furthermore, a questionnaire was given to understand students' perceived satisfaction with using ASP-supported problem-solving-based teaching approach.

## Research variables

Students were taught within the same learning period by the same teacher with the same tools and learning materials provided in this study. Moreover, additional variables were controlled. The number of annotations and homework were explored to show their influence on students' learning achievement. The stepwise multiple regression analyses were used to predict learning achievement from annotations and homework solutions. The independent variables are: (1) the quantity of annotations, (2) the quantity of one's own annotations reviewed, (3) the quantity of peers' annotations reviewed, (4) the quantity of homework solutions, (5) the quantity of one's own homework solutions reviewed, and (6) the quantity of peers' homework solutions reviewed. The dependent variable is (7) learning achievement.

The following variables related to the number of annotations and homework solutions were defined, and we explored the relationships of these variables among each other and with students' learning performance:

- (1) The quantity of annotations: The number of annotations made by a student on learning materials.
- (2) The quantity of one's own annotations reviewed: The number of times that a student reviewed their own annotations on learning materials.
- (3) The quantity of peers' annotations reviewed: The number of times that a student reviewed peers' annotations on learning materials.

- (4) The quantity of homework solutions: The number of annotated homework solutions created by a student on homework assignments.
- (5) The quantity of one's own homework solutions reviewed: The number of times that a student reviewed their own homework solutions on homework assignments.
- (6) The quantity of peers' homework solutions reviewed: The number of times that a student reviewed peers' homework solutions on homework assignments.
- (7) Learning achievement: These were measured by the post-test scores earned by the students in the final examination at the end of the course.

## Results

### Correlation analysis: Annotations, homework, and learning achievement

According to annotations and homework recorded by students' manipulation portfolio data, we found that the proportion of annotations to learning materials was 0.38, and the proportion of annotations to homework was 0.62. The Pearson's correlation was utilized to measure the relationships between variables related to annotations, homework, and learning achievement. The results are shown in Table 1 and Table 2 and some findings are discussed below.

#### *Correlation analysis of annotations and learning achievement*

The quantity of annotations ( $r = 0.552^{**}$ ) and the quantity of their own annotations reviewed ( $r = 0.433^{**}$ ) were significantly correlated with learning achievement. This finding was explored deeply via content analysis. From content analysis, we found that students who created annotations focused on clarifying important points of the learning materials. Moreover, students usually recollected their own previous knowledge when they experienced difficulties understanding learning material content. Students looked back to their records to recall their memories and to acquire new ideas, which in turn led to a deeper comprehension of learning material content. Meanwhile, an interesting phenomenon found was that a significant correlation existed between the quantity of annotations made and the quantity of the students' own annotations reviewed ( $r = 0.460^{**}$ ), implying that students would like to look back records because those comments were very meaningful to them. Furthermore, students would review their comments or ideas to find important points or solve problems when studying learning materials.

No significant correlation was found between the quantity of peers' annotations reviewed and learning achievement ( $r = -0.152$ ). From content analysis, we found that some school children annotated the learning material overly, such as mark annotations. Furthermore, students reported that they could not easily recall what was understood based on these mark annotations when they returned to peers' annotations. During the interview, students mentioned that peers' annotations helped them to understand important hints and concepts of the learning material content.

*Table 1. Correlation statistics of annotations and learning achievement*

	Learning achievement	The quantity of annotations	The quantity of one's own annotations reviewed	The quantity of peers' annotations reviewed
Learning achievement	1	0.552 <sup>**</sup>	0.433 <sup>**</sup>	-0.152
The quantity of annotations	0.552 <sup>**</sup>	1	0.460 <sup>**</sup>	-0.110
The quantity of one's own annotations reviewed	0.433 <sup>**</sup>	0.460 <sup>**</sup>	1	0.215
The quantity of peers' annotations reviewed	-0.152	-0.110	0.215	1

*Note.* <sup>\*\*</sup> $p < .01$ .

Second, the relationships between variables related to homework and learning achievement were analysed using Pearson's correlation. The results are shown in Table 2 and some findings are discussed below.



*Correlation analysis of homework and learning achievement*

The quantity of homework solutions was significantly correlated with learning achievement ( $r = 0.711^{**}$ ), implying that the annotations of homework assignments were strongly associated with learning achievement. Students thought that using ASP-supported problem-solving-based teaching approach could boost a deeper understanding of their homework problem. The quantity of students' own homework solutions reviewed had a significant correlation with learning performance ( $r = 0.527^{**}$ ). Students indicated that reviewing their own annotated homework inspired them to find potential and relevance solutions. The quantity of one's own homework solutions reviewed had a significant correlation with the quantity of homework solutions ( $r = 0.424^{**}$ ). Through content analysis, it was found that students preferred to review their annotated homework so that solving homework problems became a good learning process. This process could then help students gather relevant knowledge to acquire their potential solutions needed, which belong to a high level of cognition for students' problem-solving abilities. The quantity of peers' homework solutions reviewed had no significant correlation with learning achievement ( $r = -0.152$ ). Students reported that they reviewed the homework solutions of their peers when they encountered difficulties in solving homework problems. However, some students reviewed a high frequency of peers' solving homework problems that mostly consisted of a peripheral review of the ideas. We found that some school children preferred to review and copy the ideas and thoughts of the teacher and classmates when they improved upon their homework solutions. The interviews revealed that students preferred to review peers' homework solutions to deep classmates' intelligence of various solving strategies to help solve new homework.

*Table 2. Correlation statistics of homework and learning achievement*

	Learning achievement	The quantity of homework solutions	The quantity of one's own homework solutions reviewed	The quantity of peers' homework solutions reviewed
Learning achievement	1	0.711 <sup>**</sup>	0.527 <sup>**</sup>	-0.152
The quantity of homework solutions	0.711 <sup>**</sup>	1	0.424 <sup>**</sup>	-0.110
The quantity of one's own homework solutions reviewed	0.527 <sup>**</sup>	0.424 <sup>**</sup>	1	0.210
The quantity of peers' homework solutions reviewed	-0.152	-0.110	0.210	1

*Note.* <sup>\*\*</sup> $p < .01$ .

*Effects of annotations and homework on learning achievement*

Multiple regression analyses were employed to further predict the impact of annotations and homework solutions on learning achievement. The results of the stepwise multiple regression analyses are shown in Table 3 and Table 4. In Table 3, the R square value ( $R^2 = 0.505$ ,  $F = 35.736$ ,  $p < .01$ ) revealed that the model predictability is 50.5%. That is, six independent variables could be used to predict learning achievement: annotations, one's own annotations reviewed, peers' annotations reviewed, homework solutions, one's own homework solutions reviewed, and peers' homework solutions reviewed.

*Table 3. The summary of the multiple regression model*

Model	R	R <sup>2</sup>	Adjusted R square	F	p
1	0.711	0.505	0.491	35.736	.000 <sup>**</sup>

*Note.* <sup>\*\*</sup> $p < .01$ , dependent variable: learning achievement.

The results of stepwise multiple regression analyses are shown in Table 4. It was found that six independent variables related to annotation and homework solutions were in this regression analysis, and only values with significant potentiality were included into the stepwise multiple regression model. Our results showed that the standardized coefficients  $\beta$  of "the quantity of homework solutions" was 0.711 ( $t = 5.978$ ,  $p < .01$ ), which was the highest of all. Only the quantity of homework solutions out of all other variables (the quantity of annotations, the quantity of one's own annotations reviewed, the quantity of peers' annotations reviewed, the quantity of one's own homework solutions reviewed, and the quantity of peers' homework solutions reviewed) was the significant predictor of learning

achievement. Therefore, the teacher should encourage students to use ASP-supported problem-solving-based teaching approach to search and collect relevant knowledge to finish homework assignments. The teacher should also inspire students to increase the quantity of homework solutions, which can help foster the learning achievement of school children into Scratch programming activities.

*Table 4. Stepwise multiple regression model of predicting learning achievement*

Model	Independent variables	$\beta$	$B$	$t$	$p$
	The quantity of homework solutions	0.711	0.079	5.978	.000**

*Note.* \*\* $p < .01$ , dependent variable: learning achievement.

### Results of questionnaire

Based on the data collected from the questionnaire, we examined some meaningful information and findings from the educational point of view. With respect to students' perceived satisfaction shown in Table 5 (mean score of all items is higher than 4.0), most students agreed that the ASP-supported problem-solving-based teaching approach could satisfy the needs of learning Scratch programming. The mean score of items 1, 2, 3, 5, 6, and 7 is higher than 4.2, implying that the ASP-supported problem-solving-based teaching approach could help students to easily express their thoughts and to clearly explain their homework problem process. The annotation tool can also support annotation storage and share, so students can subsequently look back their previous comments; it provides opportunities for them to critically find potential solutions and to think about how to improve them.

The average score of item 4 is the lowest in Table 5. It was found that understanding classmates' homework problems were too difficult and few students could give helpful clues. Furthermore, the teacher might need to provide more help to the few students who did not achieve an understanding of classmates' homework problems. Based on students' higher agreements on these items, they expressed higher interests in using ASP-supported problem-solving-based teaching approach for study with their friends after class.

*Table 5. Students' perceived satisfaction of using ASP-supported problem-solving-based teaching approach*

#	Item	SA	A	U	D	SD	Avg.
1.	I think that the ASP-supported problem-solving-based teaching approach can help me to easily express my annotations and to clearly explain my homework problem process.	13	20	4	0	0	4.24
2.	I think that the ASP-supported problem-solving-based teaching approach can give me a suitable problem-solving approach to learn Scratch programming.	15	20	2	0	0	4.35
3.	I think that the ASP-supported problem-solving-based teaching approach can help me to clear my homework content.	14	19	4	0	0	4.27
4.	I think that understanding classmates' homework problems are easy and I can provide helpful comments.	12	15	10	0	0	4.05
5.	I think that the annotation tool can provide opportunities for me to critically review my annotations and to think about how to improve the quality of my homework solutions.	13	21	2	1	0	4.24
6.	Compared with online homework that did not provide the ASP-supported problem-solving-based teaching approach, homework with the innovative approach used in my class did satisfy my functional demands. I had got more happiness in learning Scratch programming.	18	13	6	3	0	4.49
7.	Comparing with online homework that did not provide the ASP-supported problem-solving-based teaching approach, homework with the innovative approach used in my class did improve my learning achievement.	14	19	4	0	0	4.27

In this questionnaire, there was one open-ended question to allow students to present their perceived satisfaction of using ASP tool-supported problem-solving-based teaching approach. To evaluate the students' intentions to accept

the innovative approach, a question was presented: “Do you like to learn Scratch programming via using ASP tool-supported problem-solving-based teaching approach?” The following statements reflect the students’ responses:

“The innovative approach was helpful for learning Scratch programming. When I encountered homework problems, I usually asked the teacher or my classmates to get assistance in solving my problems. Moreover, my annotations made in class gave me important clues to solve my problems.”

“I know I could find some friends online. I took some annotations for my friends when I found something wrong in their homework. To remind my friends take care of mistakes in their homework. If I find someone’s annotations are valuable, I will repeatedly recall his/her comments and update my homework solutions in the other way.”

“I want to view my classmates’ homework solutions to check whether they are better than mine. Also, I think this helps me to find the drawbacks of my homework methods and inspire me to improve the quality of homework solutions.”

“Although I often do not fully understand others’ annotations in Scratch programming studies, I think that the activity will do me good.”

## **Discussion**

In this study, we aimed to investigate the effects of annotations and homework on learning achievement. The findings are summarized as follows.

We investigated the effects of annotations and homework on learning achievement; interpretations of why these results were obtained are presented. The results of stepwise multiple regression analyses showed that only the quantity of homework solutions, out of all other variables (the quantity of annotations, the quantity of one’s own annotations reviewed, the quantity of peers’ annotations reviewed, the quantity of one’s own homework solutions reviewed, and the quantity of peers’ homework solutions reviewed), is the significant predictor of learning achievement. Although other variables did not significantly predict students’ learning performance, most students believed that they reviewed annotations to recall the content of learning materials and to understand it better. Besides, reviewing peers’ annotations and homework solutions did not correlate with students’ learning performance. This is in accordance with past studies (Hwang et al., 2007; Su, Yang, Hwang, & Zhang, 2010; Tam, 2009) that reviewing peers’ annotations were not found to improve students’ learning performance. The primary reason for this is because peers’ mark annotations are not easy for others to understand; the meaning and understanding of annotations are strongly connected to their creators. Therefore, it is recommended that the teacher should consider using the ASP-supported problem-solving-based teaching approach that can boost problem-solving skills and cognitive development into learning Scratch programming process, such as encouraging school children to take annotations, solving homework, and sharing them.

From the results of the questionnaire, it was found that most students were satisfied with using ASP-supported problem-solving-based teaching approach in the context of learning Scratch programming. The students also had strong desires to use the innovative approach to do their homework, to explain how they solve problems, to recall their acquired learning knowledge, to simulate their new thoughts, and to provide helpful suggestions to others. Students reviewed annotations to reinforce a deeper comprehension of the learning material content and homework problems. However, few students thought that understanding classmates’ homework mistakes were a barrier to giving helpful suggestions. Thus, the teacher would give more help to weaker students by providing useful suggestions and appropriate prizes. Moreover, the innovative approach was crucial for helping students find right peers and right information and for inspiring constructive ideas in order to complete homework.

With respect to the findings related to the open-ended question of this questionnaire, most students mentioned that there were a number of benefits from using ASP tool-supported problem-solving-based teaching approach in learning Scratch programming. Most students perceived that the innovative approach could help them to improve the quality of their homework solutions to explore how to overcome their homework difficulties and mistakes reflected on their learning performance. Furthermore, most students indicated that they liked to view suggestions to recall their

memories and to inspire fresh ideas for resolving their difficulties. Our results revealed that most of the students were positive and considered the innovative approach to be convenient and helpful to learning Scratch programming.

## Conclusion

This study developed an annotation-based Scratch programming system, ASP, which was helpful in creating, sharing, and reviewing annotations and homework. We aimed at exploring the effects of annotations and homework on students' learning achievement in an empirical study of Scratch programming pedagogy. Our results are consistent with some earlier studies (Hwang et al., 2008; Nokelainen et al., 2005; Slotte & Lonka, 1999; Su & Yang, 2010; Su, Yang, Hwang, & Zhang, 2010; Su et al., 2014), which indicated that making annotations are significantly correlated with students' learning achievement. Meanwhile, the quantity of one's own annotations reviewed and the quantity of one's own homework solutions reviewed are significantly correlated with students' learning performance. An implication of this is that a deeper understanding of learning materials and homework problems are directly correlated with reviewing products of study, and furthermore, that homework is also helpful in individual learning. The review of annotations and homework solutions written by other peers was found to have no significant influence on students' learning achievement. Students reported that a primary reason for this phenomenon is the fact that it is hard to understand mark annotations made by others; annotations are usually most meaningful to their creators. We found that some students annotated the learning material overly, such as mark annotations. Furthermore, students reported that they could not easily recall what was understood based on these mark annotations when they returned to peers' annotations. Only the quantity of homework solutions among all other variables was found to be a significant predictor of students' learning achievement. Student reported that they preferred to annotate homework problems to deep peers' intelligence of various solving strategies to help solve new homework. The quantity of peers' homework solutions reviewed had no significant correlation with learning achievement. However, students said during the interview that reviewing peers' homework solutions was helpful for them to find some helpful clues. Therefore, further investigation of this phenomenon is necessary.

According to the findings of this questionnaire, most of the students maintained an overall positive perception of using, and were generally satisfied with, the ASP-supported problem-solving-based teaching approach. Most of the students perceived that the innovative approach could stimulate them to figure out more main points and to perceive more incomprehensible problems, which may make them pay more attention during a lecture. Furthermore, doing homework with the innovative approach could increase students' interest, happiness, and achievements in learning Scratch programming. The innovative approach could help students to easily express thoughts and to clearly explain their process for solving homework assignments, in addition to improving interaction among students, annotations, and homework solutions. However, a few students thought that it was hard to understand annotations made by other classmates; they felt that the teacher might need to provide more help and suggestions. Finally, most of the students thought that the innovative approach could satisfy the needs of learning Scratch programming. From the findings of the open-ended question of this questionnaire, most students liked to make and review digital annotations while learning Scratch programming and doing homework assignments. Through the use of the innovative approach, students felt they could more readily find potential resources and information to help solve problems. These findings are consistent with some past studies (Cheng, Lou, Kuo, & Shih, 2013; Hsiao & Brusilovsky, 2010; Su, Yang, Hwang, & Zhang, 2010; Su et al., 2014), which indicated that the innovative approach could enable students to deeply understand the content of the learning material and homework presented in this course.

Based on the findings of the statistical analyses and questionnaire, it is recommended that teachers consider incorporating the innovative methods into their educational environment in order to foster students' cognitive development in the area of Scratch programming. This can be achieved by encouraging students to make annotations in the homework assignments, followed by subsequent reviews. The creation and review of annotations and homework were found to be beneficial to learning Scratch programming and doing homework assignments; therefore, it might be useful for encouraging students to create and review their annotations more frequently and more deeply. The results of this study may shed light on higher levels of knowledge understanding with the ultimate aim of improving students' understanding and practice of Scratch programming. The key issue is we provided a tool for sharing annotations and homework but the study focused on benefit of annotations and homework to individual learning only. In future studies, we will adopt collaborative problem-solving strategy to solve group homework, we will evaluate if this strategy is able to improve school children' learning performance. Furthermore, we hope to see from our future studies that such annotations will stimulate students to make more annotations and to receive more

benefit from the outputs of their studies. The effects of using annotations within the collaborative problem-solving strategies will be investigated in future study.

## Acknowledgements

This work is supported in part by Ministry of Science and Technology, Republic of China, Taiwan under contract numbers MOST 104-2517-S-008-001 and MOST 104-2511-S-008 -011.

## References

- Brown, J. D. (1996). *Testing in language programs*. Upper Saddle River, NJ: Prentice Hall Regents.
- Carmines, E. G., & Zeller, R. A. (1979). *Reliability and validity assessment*. Beverly Hills, CA: Sage.
- Cheng, Y. M., Lou, S. J., Kuo, S. H., & Shih, R. C. (2013). Investigating elementary school students' technology acceptance by applying digital game-based learning to environmental education. *Australasian Journal of Educational Technology*, 29(1), 96-110.
- Davis, F. D. (1986). *A Technology acceptance model of empirically testing new end-user information systems: Theory and results* (Unpublished doctoral dissertation). Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA.
- Garner, S. (2009). Learning to program from Scratch. In *Proceedings of the 9th IEEE International Conference on Advanced Learning Technologies* (pp. 451-452). doi:10.1109/ICALT.2009.50
- Hung, Y. C. (2008). The Effect of problem-solving instruction on computer engineering majors' performance in Verilog (HDL) programming. *IEEE Transaction on Education*, 51(1), 131-137.
- Hsiao, I. H., & Brusilovsky, P. (2010). The Role of community feedback in the student example authoring process: An Evaluation of AnnotEx. *British Journal of Educational Technology*, 42(3), 482-499.
- Hwang, W. Y., Wang, C. Y., & Sharples, M. (2007). A Study of multimedia annotation of web-based material. *Computers & Education*, 48(4), 680-699.
- Papert, S. (1980). *Mindstorms: Children, computes, and powerful ideas*. New York, NY: Basic Books.
- Nokelainen, P., Miettinen, M., Kurhila, J., Floréen, P., & Tirri, H. (2005). A Shared document-based annotation tool to support learner-centered collaborative learning. *British Journal of Educational Technology*, 36(5), 757-770.
- Power, T. J., Dombrowski, S. C., Watkins, M. W., Mautone, J. A., & Eagle, J. W. (2007). Assessing children's homework performance: development of a multi-dimensional, multi-informant rating system. *Journal of School Psychology*, 45(3), 333-348.
- Slotte, V., & Lonka, K. (1999). Review and process effects of spontaneous note-taking on text comprehension. *Contemporary Educational Psychology*, 24, 1-20.
- Su, A. Y. S., & Yang S. J. H. (2010). Improving annotation categorization performance through integrated social annotation computation. *Expert Systems with Applications*, 37(12), 8736-8744.
- Su, A. Y. S., Yang, S. J. H., Hwang, W. Y., & Zhang, J. (2010). A Web 2.0-based collaborative annotation system for enhancing knowledge sharing in collaborative learning environments. *Computers & Education*, 55(2), 752-766.
- Su, A. Y. S., Yang, S. J. H., Hwang, W. Y., Huang, C. S. J., & Tern, M. Y. (2014). Investigating the role of computer-supported annotation in problem solving based teaching: An Empirical study of a Scratch programming pedagogy. *British Journal of Educational Technology*, 45(4), 647-665.
- Tam, C. W. (2009). Homework involvement among Hong Kong primary school students. *Asia Pacific Journal of Education*, 29(2), 213-227.
- Yang, S. J. H., Chen, I., & Shao, N. (2004). Ontological enabled annotations and knowledge management for collaborative learning in virtual learning community. *Educational Technology & Society*, 7(4), 70-81.
- Yang, S. J. H., Zhang, J., Su, A. Y. S., & Tsai, J. J. P. (2011). A Collaborative multimedia annotation tool for enhancing knowledge sharing in CSCL. *Interactive Learning Environments*, 19(1), 45-62.
- Yeh, S. W., & Lo, J. J. (2009). Using online annotations to support error correction and corrective feedback. *Computers & Education*, 52(4), 882-892.