

The Development and Application of Computational Fairy Tales for Elementary Students

Jungho Park¹

¹ Department of Computer Education, Chinju National University of Education 3, Jinyangho-ro 369beon-gil, Jinju-si, Gyeongsangnam-do, Republic of Korea

Correspondence: Jungho Park, Department of Computer Education, Chinju National University of Education 3, Jinyangho-ro 369beon-gil, Jinju-si, Gyeongsangnam-do, Republic of Korea. E-mail: jhpark0154@gmail.com

Received: April 4, 2018

Accepted: May 28, 2019

Online Published: May 31, 2019

doi:10.5430/ijhe.v8n3p159

URL: <https://doi.org/10.5430/ijhe.v8n3p159>

Abstract

In the field of K-12 education, the demand for effective coding education is gradually expanding with various coding tools such as Scratch being popularly used as an effective learning environment. However, an answer to the question of what constitutes appropriate computing concepts for children (e.g. elementary school students) has not been fully answered. In this regard, this study worked to develop computational fairy tales (CFT) for coding beginners and applied it to the elementary school classroom environment in Korea. This CFT was developed by extracting the concepts of computer science through literature analysis, developing a plot/episode and creating a story. The final CFT developed is composed of 15 core computational concepts. 152 elementary students participated in the experiment where students read the CFT and solved related problems over two weeks period. We analyzed the effects of the CFT on the acquisition of basic computing concepts and coding attitudes. The results of the study showed that both the score of computing concept comprehension and attitude was enhanced significantly ($p < .001$). This study demonstrates the positive educational effects of CFTs in fostering understanding of basic computing concepts before students begin to code with algorithms directly.

Keywords: computational fairy tales, unplugged activities, computing concepts, computing attitude introduction

1. Introduction

In the 1990s, people began to talk about the Knowledge Society, noting that information is useful only when it is transformed into knowledge. But, in these days knowledge alone is not enough. The development of computing technology is changing our daily lives rapidly. In today's rapidly-changing world, people must continually come up with creative solutions to unexpected problems.

Success is based not only on what we know or how much we know, but on how think and act creatively. In other words, just acquiring knowledge is not enough; being able to create solution using technology is essential to finding a job.

Across the globe, countries are trying to foster competitive talent for the future in expectation of massive social and economic changes due to the 4th industrial revolution (Rim and Lim, 2018), (Choi, 2017). The Partnership for 21st Century Skills (2013) reports that students will require an assortment of skills, including creative thinking, communication, critical thinking and collaboration, all known to be essential skills for the 21st century (The Partnership for 21st Century Skill, 2013).

Unlike Information Communication Technology (ICT) education based on software applications, computer science education helps to develop computational thinking skills as well as digital skills in order to understand the world and enhance problem solving competency. Moreover, early exposure to computer science for K – 12 students produces students with a better grasp on the advantages and limitations of computing. This can have the added benefit of aiding student's decision making ability and preparing them for situations where they should consider how the technology affects their privacy, how autonomous systems impact their lives and how information and computing can be leveraged for the benefit of the disadvantaged. In this way, far from being just programming, early access to computer science classes fosters "computational thinking" skills.

For this reason, among developed countries, computer science education is a requisite course of study in primary and secondary education as a means to increase Computer Science (CS) skills among students.

For example, in the UK, beginning in 2014 the Department for Education began shifting from a conventional ICT based education to a computing-based education which includes computer science, information technology, and digital skills and has now implemented this for all K-12 educational institutions (Department for Education, 2014), (Bell et al, 1998).

In the US, computer science is involved in the International Society for Technology in Education's education course and the American Computing Machinery education course models (The Joint Task Force on Computing Curricula, 2013), (Han and Han, 2009). In line with this trend, South Korea has announced a 'Software Education Plan', which requires all elementary school students to learn programming as part of their regular curriculum beginning in 2019. Consequently, all elementary students will have to learn coding and algorithms as part of their computer science curriculum, for 17 hours starting 2019 while middle school students will have already taken 34 hours yearly by 2018.

The educational aims of the Software Education Plan at the elementary school level is to develop computational thinking (CT) skills through experiential activities with algorithms and programming in order to develop skills applicable to real life problems. CT is a kind of problem-solving process based on computing education. Although CT is essential to the development of computer applications, but it can also be used to support problem solving across all disciplines (Baek et al, 2008).

An effective activity for learning CT is computer programming, also known as coding, is the process of creating software. Coding is the act of using languages to instruct a computer to perform functions. More countries are seeing the importance of teaching this skill to young students all over the world.

Kelleher and Pausch (2005) say that programming skills can be difficult to acquire for beginning students across the age spectrum. Beginner programmers must familiarize themselves with commands and terms which may not be easy to fathom in addition to the already challenging task of learning to develop structured solutions to problems and how to implement them (Kelleher and Pausch, 2005), (Todman and Dick, 1993).

It is also known that students often experience learning difficulties due to the abstract nature of knowledge in the field of computer science (Scott and Ghinea, 2013). Yet, few practical applicable educational strategies or approaches have been identified that may increase student's interest toward higher education or careers in computer science, especially among elementary and middle school students (Moore, 2008).

Currently in Korea many classrooms use educational programming languages, unplugged activities and physical computing as a tool for Software Education (SW) classes.

Software is a set of instruction for a computer to perform and programming is writing instructions for a computer to perform. The instructions, called source code, are written in special language that are unlike natural human languages. It also involves planning, testing, and debugging source code.

There are a lot of resources help K-6 students learn how to code. Some examples include Scratch and the Hour of Code in code.org which are quite useful for someone new to get acquainted with programming. Through these platforms, students write simple programs that make graphical creature move, build games and learn the basic tools of programming. The major strength of these platforms is the visual feedback from the platform, which helps the students stay constantly engaged with the project.

However, tech-centric teaching methods may increase anxiety and pressure on elementary school students who have no experience in algorithms and programming. Pedagogy and classroom activities must be re-conceived to provide context and a hands-on approach in order to make CT relevant to students.

According to Furner (2017), to address the issue of anxiety among their students, the inclusion of literature and writing have proven effective (Furner, 2017; Mokhtar, 2017; Mustafa, Tourkia, and Ramadan, 2017; Nair and Hui, 2018; Gumusgul, 2018; Kögce and Aslandag, 2018; Kyrychenko, 2018). It is also suggested that teachers use children's literature to combat anxiety by using these materials in a shared setting to foster connections in a more meaningful ways along with the inclusion of "writing, discussion, and the use of manipulatives as they relate to the story in the children's literature book".

Storytelling as a pedagogical tool has been used for years in the humanities and in childhood education (Russell, 1994). It can also be used as a tool to introduce computer science topics as well. Metaphorical elements in such stories make concepts more interesting, while also making them more concrete and memorable. Because of this, the use of stories can make technical concepts more accessible to students who would otherwise be turned off by the study of computing concepts.

Unplugged activities are interactive activities aimed at promoting computer science skills in young people through the use of non-electronic methods. Examples can include board and card games, puzzles and other activities which use physical materials. They allow students to engage in computational thinking exercises without the potential anxiety of engaging with unfamiliar technologies, much in the way that CFTs do.

Therefore, this study develops and applies computational fairy tales and unplugged activities as an introductory teaching method for children to learn the concepts of computing easily. This study is significant as it sheds new light on the link between computer science education and fairy tales and unplugged activities (Suryadibrata and Kim, 2017).

2. Theoretical Background

At their core, fairy tales use narrative as a tool to impart a cautionary or moral lesson. On this basis alone, they are able to provide a vehicle to transmit other types of thinking and lessons beyond the traditional. Fairy tales are ubiquitous across cultures and can be enjoyed regardless of gender or age. Children in particular are able to easily immerse themselves in the narratives of fairy tales, making them an ideal means to transmit ideas about computer science while simultaneously triggering the learner's interest.

According to Russel (1994), children are able to "experience others' adventures in their imagination", "experience insight to solve a problem", "understand others' difficulties and expand their empathy", and "experience interest and emotional escape" (Russell, 1994), (Suryadibrata and Kim, 2017). In other words, through fairy tales, children can learn a language while simultaneously absorbing knowledge, information and emotions. Fairy tales do not offer any direct delivery, demonstration, explanation, or description of knowledge information. Rather, these things are acquired meaningfully and in context.

The ability of fairy tales to impart knowledge on various subjects such as mathematics, science and Chinese characters speaks to just one of the advantages of using them as an educational tool. For example, Furner (2017) says that "using children's literature and fairy tales to teach math may help in addressing the improvement of attitudes toward mathematics and help students to be more creative and imaginative in a world with endless possibilities and with advancing technologies" (Furner, 2017), (Shin and Jung, 2017). Therefore, it is reasonable to conclude that applying the concepts and principles of computer science to the attributes of fairy tales would be equally effective. These are known as computational fairy tales.

There are some stories written with computer science in mind such as Computational Fairy Tales, Hello Ruby, and Lauren Ipsum. Davic Touretzky's book on Lisp includes a story called "Martin and the Dragon" which teaches the concept of recursion (Gerçeker, 2018). In the story, a boy finds a list problem and asks for help from a dragon wise in the ways of computing. The dragon declares that he will not work on an entire list, since he's not in an agreeable mood, but will work on one element as many times as the boy wants. The clever Martin tricks the dragon into helping him with the entire task through the use of recursion. Through this simple story, the concept of recursion is introduced and demonstrated.

Especially notable is Computational Fairy Tales by Kubica (2012) which includes over 70 stories which cover a range of different computer science concepts from introductory programming, to high level CS concepts, to data structures and algorithms, computational complexity and practical programming tips (Ghanney, 2018). He also developed computer science episodes available in three levels. The beginner level focuses on general concepts (e.g. very high-level algorithms) and simple programming concepts.

These are good for people with little to no programming experience, including those who are learning to program for the first time and people with no interest in programming who may merely need to understand the core concepts. At the intermediate level the books introduce algorithms, data structures, and practical programming techniques.

These are concepts that are generally introduced in the latter half of an introductory CS course and in an intro to algorithms course. This would be appropriate for those with some experience programming. The advanced level covers advanced algorithms, pointers and computational theory. These are topics commonly found in algorithm courses and are good for those who have a higher degree of experience or have taken computer science courses.

Despite the multiple advantages of using fairy tales in the classroom and the many computational fairy tales that have been published, studies that examine the use and effectiveness of computational fairy tales are rare.

According to Duval (2008), university students' application of computational fairy tales have helped to garner a positive reaction in terms of their understanding of the concepts of computer science and memory retention, and was especially effective in aiding the visualization of abstract concepts, memorization of syntax rule, and understanding

of complicated algorithms (Ha and Tran, 2018). He gives many examples of stories that can be deployed in computing classes and analyses the advantages and possible disadvantages of doing so.

Sullivan (2012) made use of the fairy tale ‘Santa’s Dirty Socks’ to introduce a divide and conquer algorithm to elementary school students (Halder and Chandra, 2012; Hussein, 2017; Lawal and Samy, 2017; Marangoz, 2018; Mbunda-Nekang, 2018). In Korea, Park (2017) applied a computational fairy tale based unplugged activities approach to prospective teachers, and the positive learning in computer science was demonstrated to be more meaningful than simply using unplugged computer science activities by themselves (Hassan and Kommers, 2018). In addition, research found that both SW motivation and computational thinking was improved after conducting classes using CFTs.

However, in spite of these studies, more quantitative and qualitative research needs to be conducted in order for CFTs to be recognized for their educational effectiveness. In addition, it is necessary to develop computational fairy tales which take into account the development level of elementary school students as well as to develop a curriculum for elementary SW education with an aim to conduct experimental research using them.

3. Method

3.1 The Development of a CFT and Unplugged Activities

In this study, a CFT was designed to lay the foundation for computational thinking by connecting basic concepts of computer science naturally into a story. In other words, the CFT is a fairy tale that is appropriate for the developmental stage of elementary school students and is designed to contain a variety of computing concepts mean to stimulate interest. The CFT was developed using the procedure that follows:

First, an overall concept for the fairy tale was developed by analysing domestic and foreign SW education curricula and relevant literature while considering the developmental stage of elementary school students. Through this process, 15 core concepts of computing were selected (e.g. algorithms, binary numbers, conditions, sequences, recursion, events etc.)

Next, a plot for the fairy tale, character and background were designed and story lines for each concept were developed.

Third, unplugged activities and illustrations associated with the story lines were developed. Illustrations were used to help make the CFT more understandable. The following Fig. 1-2 are examples of illustrations.



Figure 1. First version of illustration samples of Computational Fairy Tale

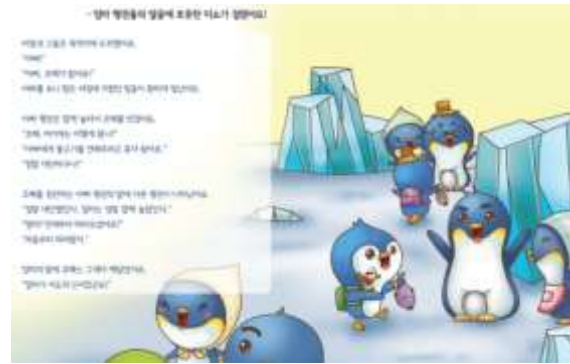


Figure 2. Final version of illustration samples of Computational Fairy Tale

Finally, to create a story, the characters in the fairy tale were set up in detail. By connecting the smaller story lines and unplugged activities, one whole story was developed.

The main theme of the computing fairy tale is presented as follows: The setting of the fairy tale is Antarctica, and the main character is a penguin names a portmanteau of Coding and Penguin (Co-pe).

He is set to take a long trip to Antarctica to deliver fish quickly to his father, who is caring for his eggs at the coldest point in Antarctica. Throughout the journey, Co-pe visits several villages and is supposed to learn the concepts and principles of computer science, such as sequences, patterns, decomposition, repetition, abstraction, conditionals, functions, algorithms, binary numbers, searching, sorting, coding events, etc.

For example, the main plots in the town of ' Binary ' are: Both numbers and letters are binary in town as in the following Fig. 2. with the goal being to understand binary and communicate through binary numbers.

Co-pe observes the calculation of the value of goods in the village and discovers the principle of binary numbers. Then he finds a sign that says how to get out of town through binary calculation. Other episodes in the book contain similar concepts and related unplugged activities.

1	2	3	4	5	6	7	8	9	10	11	12	13
a	b	c	d	e	f	g	h	i	j	k	l	m
14	15	16	17	18	19	20	21	22	23	24	25	26
n	o	p	q	r	s	t	u	v	w	x	y	z

Figure 3. The table of number and letters for reference

■	■	□	■	□	e
□	□	■	■	■	x
■	□	■	■	□	i
□	■	□	■	■	t

Figure 4. Text representation through binary decoding and encoding

There are secondary characters as well, such as the krill shrimp Pongpong-e, a friend who helps out Co-pe on occasion. There are also seals, predators of penguins like Co-pe and his friend, who act as antagonists in the story, providing more opportunities to introduce CS concepts.

Since Bell (1998) first introduced the unplugged activities based on play and games to help his students learn computer science (Holloway et al, 2018), many researchers have used unplugged activities as a way of allowing students to experience computer science (Horzum, et al, 2018) (Touretzky and Lisp, 2012).

There were two types of unplugged based activities applied in this study.

First, board games in the form of a “roll and move games” were employed as shown in Fig. 5.

Roll and Move Games involve rolling dice (or spinning a wheel or drawing cards) to determine the number of spaces you may move in a primarily linear direction. Players may be racing from a start point to a finish line or moving to control resources. This type of game lent itself to teaching the concept of conditionality, where landing on different spaces resulted in different outcomes. For example, “Storm” spaces send the player back 3 spaces and encountering seals sends the player back to the nearest igloo.



Figure 5. Unplugged activities: dice board game

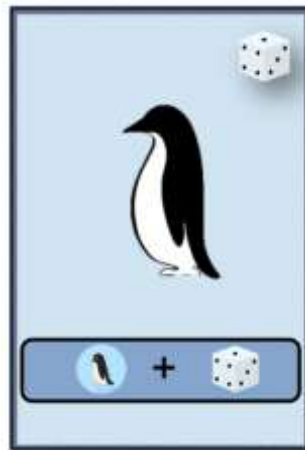


Figure 6. Unplugged activities: coding instruction card

Another type of unplugged activity was to solve problems using CS concepts. Fig. 8 is an algorithm problem activity where the player gives instructions to either move spaces or color spaces. In figure 9 we see the arrow (➡) which means ‘MOVE’ and the 🎨 symbol which means ‘PAINT’.

Students attempted to color the board to match the given pattern and were then encouraged to revisit their solution to find a more compact set of instruction through analysis of repetition and patterns. By repeating this process, students get an optimized algorithm. For example, repeated commands can be called continuously by defining them with a new function.

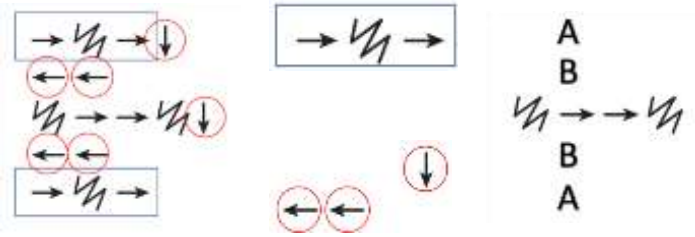


Figure 7. Unplugged activities: optimizing algorithm

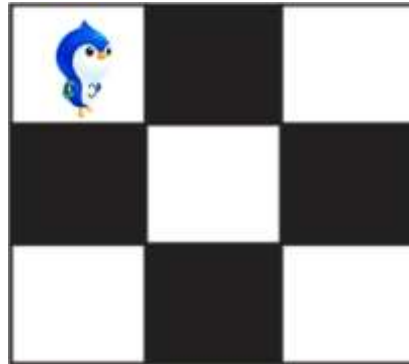


Figure 8. Unplugged activities: algorithm problem board

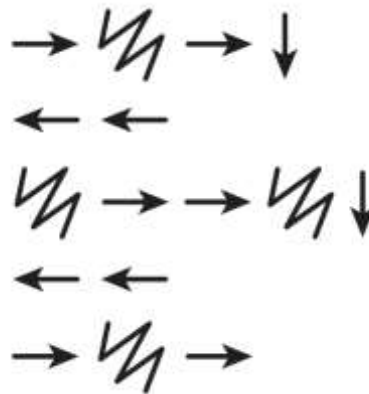


Figure 9. Unplugged activities: solution for problem

H. The application of CFT and unplugged activities

This study was carried out in SW classes for two weeks every day for 40 minutes from December 11 to December 22, 2017. The study covered 152 elementary school students (5th - 6th graders) in South Gyeong-sang Province (Male: 79 and Female: 73).

The children who participated in the experiment read and shared the story with each other for 25 minutes. Then, an opportunity was provided to experience the concepts of computing in practice through unplugged activities for 15 minutes.

In order to evaluate the effect of computational fairy tales, we conducted evaluations on computing concepts, evaluations on computing attitudes, as well as interviews.

Computing Concept Evaluation: To see how much students understood the computing concepts presented to them, researchers and three other elementary school teachers developed a total of 60 assessment questions, four for each concept. Afterwards, five expert groups consisting of professors and researchers reviewed it for validity on the Likert 5-step scale. After selecting two questions with good scores for each concept, the pilot study was conducted on 15

elementary school students for 1 week. Based on the pilot study results, the items were revised and supplemented. Finally, 15 questions were selected.

Computing Attitude Instrument: To assess students' computing attitude, we modified and used the Computer Attitude Instrument, originally developed by Todman and Dick (1993) for elementary school students (Duvall, 2018). The CAI consists of 11 Likert scale items in three areas: fun, usefulness, and ease of use. Each item was rated on a 5-point Likert-type scale. The questionnaire was translated into Korean, and five elementary school teachers examined its validity. Three items (*) were phrased negatively, and reverse scored for statistical analysis.

Table 1. Computing Attitude Instrument

Category		Description
Fun	1	I think Computing is interesting activity.
	2	I think that computing make study funny.
	3	I would like to study computing after school.
	4	I think computing is attractive activity.
Usefulness	5	I think that it is stupid thing to study computing (*)
	6	I don't want to know computing any more(*)
	7	I think that computing is more interesting than other activity
Easy of use	8	I think computing is useful activity.
	9	I think that it is hard to understand computing (*)
	10	I think that computing is a difficult activity
	11	I think that it is hard to learn how to use computing

Interview: To examine how the CFT affected the children's attitude, 35 students were randomly selected and took part in semi-structured interviews after the study to assess 'the fun of CFT', 'the hard work', and 'what they learn from classes'. The interview took 11 to 15 minutes per student. The interviewee was asked by the researcher to explain in detail, asking additional questions if the answers were not understood. All interviews were recorded and then transcribed for analysis after obtaining agreement from the interviewees.

I. Data Analysis

Data analysis was done using a Paired *t* Test, which compares the means of two of the same subject at two different times, in this case test scores before and after the CFT intervention. The purpose of this is to determine if there is a statistical difference between the pairs that significantly different from zero. The Paired *t* Test analyses were conducted using SPSS 11.0

4. Result

J. Evaluation for Computing Concepts

A paired-sample *t*-test was conducted to compare students' computing concepts in Table 2. Table 2 is the result of a Paired-sample *t* Test comparing student's computing concept literacy. Result indicates a statistically significant difference in computing concept scores between pre-test ($M=4.92$, $SD=2.13$) and post-test ($M=11.21$, $SD=1.95$), $t=-25.37$, $p=0.000$.

The increase in score to 11.21 after the experimental class compared to 4.92 before class indicates that students were able to develop a greater understanding of computing concepts through the CFT. In other words, the use of the CFT played a positive role in teaching computing concepts.

Table 3. The Result for Computing Concepts Before-After

Category	N	M	SD	T	df	Sig.(p)
pre-test	152	4.92	2.13	-25.37	151	.000(**)
post-test	152	11.21	1.95			

K. The Result of Computing Attitude

A paired-sample t-test was conducted to compare students' computing attitude in Table 3. The comparative result revealed that students had significantly higher attitude on the post-test (M=4.13, SD=0.36) than the pre-test (M=3.08, SD=0.51), $t=-21.07$, $p=0.000$.

The increase in score to 4.13 after class compared to 3.08 before class indicates that students developed a positive computing attitude because of the CFT. In other words, the use of the CFT played a positive role in shaping computing attitude.

This result shows that the stories can make computing more attractive to those who are not traditionally attracted to computing. It is generally known that many students do not find computer science interesting. The stories gave a creative flair to a computing class, and this made the class as a whole more appealing to the elementary school students.

Table 2. The Test for Computing Attitude Before-After

Category	N	M	SD	T	df	Sig.(p)
pre-test	152	3.08	0.51	-21.07	151	.000(**)
post-test	152	4.13	0.36			

L. The Results of Interview

Most of the students who participated in the experiment had previous experience with SW education. Given the significant improvement in scores and attitude, this suggests that they found it easier and more enjoyable to read the CFT to learn new computing concepts (as opposed to SW education).

They also replied that working with colleagues in the unplugged activities improved their enjoyment. Most interviewees (32/35) showed satisfaction with the use of the CFT, because they found it fun (N=33), Useful (N=31), and easy to use (N=28).

The following are some of the typical interviews.

10) The fun of CFT:

"It was similar to people when penguins travelled to get their dad fish."

"Whenever penguins went through the village, they solved a problem one by one, and it was very fun to solve the same problem in pairs."

"For the first few weeks, it was difficult to understand the rule of game and concepts of programming, but I became accustomed to playing the game and moving it with coding cards. I think learning it with a board game is very fun."

"I found that computer science helps solve problems. I want to study more computer science in the future."

"The unplugged activity felt more realistic. The story became more and more fun and interesting. It's easy to understand the solution better."

11) The hard work:

"At first, it was difficult because it was complicated and unfamiliar but after reading a fairy tale, I played games with my friend."

"It was easy to understand the sequence, the pattern and the repetition, but it was hard to learn the array, the variable and the binary number."

"What I didn't understand was solved with the help of partner."

12) *What they learn from class:*

"I learned about the concept of computing through computational fairy tales and games."

"I heard many computing words for the first time, but I was amazed that they were used in our life every day."

"In the previous coding class, we listened to teachers or watched their demonstration and carried out the exercises according to the teacher's demonstration. However, the class using board game required a lot of strategies and competitions with friends"

"I had to work with my colleagues to optimize a given algorithm. And we learned that collaboration has better results."

5. Conclusions

In order to survive in a competitive society, it is important to foster CT skills early on. Computing activities such as programming are an effective manner of fostering CT skill and many countries have already begun to integrate them into their curriculum.

Stories can help students understand computing concepts easily. In computing, teachers often discuss abstract concepts such as algorithms, and using a story can give a better mental image of what is being explained (Teaching, 2012). Because the story has a plot and characters, it becomes memorable. The story gives the students a shared vocabulary and an easy way to refer to and remember various computing concepts.

Given the above, this study developed a CFT and applied it to the elementary school classroom. It also examined the effectiveness of the CFT through experimental research in computing concepts and computing attitude. The result of this study showed a significant improvement in both attitude and concept achievement in the post assessment.

The main results of the study are as follows:

First, significant improvements were demonstrated in the results of the pre-post computing concept assessment. In addition, the result of pre-post attitude tests revealed a statistically significant difference in all areas of fun, usefulness and ease of use. These results were also confirmed in further interviews.

Ultimately, computational fairy tales demonstrated several educational benefits for children in acquiring computing concepts. First, compared to learning coding directly with computers, it is more effective to learn basic programming concepts such as sequence, repetition, conditionals, and variables through a CFT in advance because it can remove cognitive burdens. Second, experiencing unplugged activities after reading the CFT provides not only a synergy in concept formation but also is effective in developing positive computing attitudes.

Finally, talking about the concepts of computing and solving problems in pairs created opportunities for communication, sharing and collaboration.

This study has important implications for the use of fairy tales in elementary computing education. The result of this study provides certain experimental evidence in formal curriculum that the combination of storytelling and unplugged activities has a positive effect on attitude and concept achievement in computing education.

This study had several limitations and the results of this study suggests several directions for future research. This study was conducted as a pre- and post-comparative study with 152 students in a formal class setting.

However, students who participated in this research were not randomized because of certain environmental constraints. A randomized sample across several school districts may increase the generalizability of the findings (Park, 2017) (Bell et al, 1998).

Acknowledgments

Acknowledgments: This work was supported by the Ministry of Education of the Republic of Korea and the National Research Foundation of Korea (NRF-2017S1A5A8019141)

References

- Baek, S. R., Song, J. B., Park, J. H. & Lee, T. W. (2008). Development and application of algorithm teaching materials centered on plays for problem-solving abilities. *Journal of Korean Association of Computer Education*, 11(1), 85-95.
- Bell, T., Witten, I. H. & Fellows, M. (1998). Computer science unplugged: Off-line activities and games for all ages. Available from: <http://unplugged.canterbury.ac.nz>.

- Choi, H.H. (2017). Adaptive and prioritized random access and resource allocation schemes for dynamic TDMA/TDD protocols. *JICCE.*, 15(1), 28-36.
- Department for Education, (2014). The national curriculum in England. Available from: <https://www.gov.uk/government/publications/national-curriculum-in-england-primary-curriculum>.
- Duvall, S. (2018). Computer science fairy tales. *Journal of Computing Sciences in Colleges*, 24(2), 98-104.
- Furner, J. M. (2017). Using fairy tales and children's literature in the math classroom: Helping all students become einstein's in a STEM world. *Journal of Advances in Education Research*, 2(2), 103-12. <https://doi.org/10.22606/jaer.2017.22006>
- Ger çeker, C. S. (2018). A study on developing special teaching methods attitude scale for pre-service music teachers. *Asian Journal of Education and Training*, 4(4), 356-362.
- Ghanney, R. A. (2018). How parental education and literacy skill levels affect the education of their wards: The case of two schools in the Effutu municipality of Ghana. *International Journal of Education and Practice*, 6(3), 107-119. <https://doi.org/10.18488/journal.61.2018.63.107.119>
- Gumusgul, O. (2018). Comparison of Time Management Perception of Students Studying at Department of Physical Education and Sports Teaching and Program in Primary School Education. *Asian Journal of Education and Training*, 4(3), 224-227. <https://doi.org/10.20448/journal.522.2018.43.224.227>
- Ha, T. P. T. & Tran, M. D. (2018). Review of impacts of leadership competence of project managers on construction project success. *International Journal of Emerging Trends in Social Sciences*, 4(1), 15-25. <https://doi.org/10.20448/2001.41.15.25>
- Halder, S. N. & Chandra, S. (2012). Users' attitudes towards institutional repository in Jadavpur University: A critical study. *International Journal of Management and Sustainability*, 1(2), 45-52.
- Han, H. S. & Han, S. K. (2009). A case study on information education for pre-service teacher using unplugged computing. *Journal of The Korean Association of Information Education*, 13(1), 23-31.
- Hassan, M. I. A. & Kommers, P. (2018). A review on effect of social media on education in Sudan. *International Journal of Educational Technology and Learning*, 3(1), 30-34. <https://doi.org/10.20448/2003.31.30.34>
- Holloway, S. S., Romme, A. G. L. & Demerouti, E. (2018). Crafting values in organizational change processes. *International Journal of Social Sciences Perspectives*, 3(1), 7-20. <https://doi.org/10.33094/7.2017.2018.31.7.20>
- Horzum, T. & Izci, K. (2018). Preservice Turkish teachers' views and perceived competence related to inclusive education. *Journal of Education and e-Learning Research*, 5(2), 131-143. <https://doi.org/10.20448/journal.509.2018.52.131.143>
- Hussein, L. M. (2017). The Effectiveness of Teaching Educational Research Course on the Development of Scientific Research Skills, Academic and Personal Integrity among Female Students of Al-Qassim University. *International Journal of Asian Social Science*, 7(5), 392-409. <https://doi.org/10.18488/journal.1.2017.75.392.409>
- Kelleher, C. & Pausch, R. (2005). Lowering the barriers to programming: A taxonomy of programming environments and languages for novice programmers. *ACM Computing Surveys (CSUR) Surveys*, 37(2), 83-137. <https://doi.org/10.1145/1089733.1089734>
- Kögçe, D., & Aslandag, B. (2018). The Views of Classroom Teachers on Project Tasks. *Asian Journal of Education and Training*, 4(3), 233-245. <https://doi.org/10.20448/journal.522.2018.43.233.245>
- Kyrychenko, V. (2018). Indonesias Higher Education: Context, Policy, and Perspective. *Asian Journal of Contemporary Education*, 2(2), 159-172. <https://doi.org/10.18488/journal.137.2018.22.159.172>
- Lawal, O., & Samy, M. (2017). Perception of Stakeholders in Funding of Medical Education in Nigeria. *International Journal of Asian Social Science*, 7(6), 521-534. <https://doi.org/10.18488/journal.1.2017.76.521.533>
- Marangoz, I. (2018). The Determination of the Relationship between Somatotypes and Speed of Sub-Elite Athletes. *Asian Journal of Education and Training*, 4(3), 220-223. <https://doi.org/10.20448/journal.522.2018.43.220.223>
- Mbunda-Nekang, F. M. (2018). Preventing Conflict and Ensuring National Peace through Folklore: Option for Cameroon. *International Journal of English Language and Literature Studies*, 7(2), 22-31. <https://doi.org/10.18488/journal.23.2018.72.22.31>

- Mokhtar, S. B. (2017). Teaching-Learning Model of Islamic Education at Madrasah Based on Mosque in Singapore. *International Journal of Asian Social Science*, 7(3), 218-225. <https://doi.org/10.18488/journal.1/2017.7.3/1.3.218.225>
- Moore, D. (2008). *National science foundation offers \$2.7 million to foster kids' interest in computer science*. News Center (Dallas, TX).
- Mustafa, H. M., Tourkia, F. B., & Ramadan, R. M. (2017). An Overview on Evaluation of E-Learning/Training Response Time Considering Artificial Neural Networks Modeling. *Journal of Education and e-Learning Research*, 4(2), 46-62. <https://doi.org/10.20448/journal.509.2017.42.46.62>
- Nair, S. M., & Hui, L. L. (2018). An Analysis of Common Errors in ESL Descriptive Writing among Chinese Private School Students in Malaysia. *International Journal of Education and Practice*, 6(1), 28-42. <https://doi.org/10.18488/journal.61.2017.61.28.42>
- Park, J. H. (2017). The effect on unplugged computer science edutainment activity based on computing fairy tales. *Journal of Korea Entertainment Industry Association*, 11(8), 281-288. <https://doi.org/10.21184/jkeia.2017.12.11.8.281>
- Rim K. & Lim, D. (2018). DoS attack control design of IoT system for 5G Era. *JICCE.*, 16(2), 93-98.
- Russell, D. L. (1994). *Literature for children: A short introduction*. New York & London: Longman.
- Scott, M. J. & Ghinea, G. (2013). Educating programmers: A reflection on barriers to deliberate practice. Proc. 2nd HEA Conf. Learning Teaching STEM Discipl. pp: 028P.
- Shin, C.H. & Jung, C.Y. (2017). An optimal algorithm for enhancing the contrast of chest images using the frequency filters based on fuzzy logic. *Journal of Information and Communication Convergence Engineering*, 15(2), 131-136.
- Suryadibrata, A. & Kim, K.B. (2017). Ganglion cyst region extraction from ultrasound images using possibilistic C-means clustering method. *Journal of Information and Communication Convergence Engineering*, 15(1), 49-52.
- Suryadibrata, A. and Kim, K. B. (2017). Ganglion cyst region extraction from ultrasound images using possibilistic c-means clustering method. *Journal of Information and Communication Convergence Engineering*, 15(1), 49-52.
- Teaching. (2012). Teaching the principles of computer science to primary-aged children. Available from: https://www.youtube.com/watch?v=RWEAv1B1_qA, (2012).
- The Joint Task Force on Computing Curricula. (2013). IEEE Computer Society, Association for Computing Machinery.
- The Partnership for 21st Century Skill. (2013). [Online]. Available from: <http://www.p21.org>.
- Todman, J. & Dick, G. (1993). Primary children and teachers' attitudes to computers. *Computers & Education.*, 20(2), 199-203. [https://doi.org/10.1016/0360-1315\(93\)90088-Z](https://doi.org/10.1016/0360-1315(93)90088-Z)
- Touretzky, D. & Lisp, C. (2012). A gentle introduction to symbolic computation. Redwood City, CA: Benjamin Cummings, 1989. J. Kubica, Computational Fairy Tales, 2012.