Hirashima,T. et al. (Eds.) (2010). Workshop Proceedings of the 18th International Conference on Computers in Education. Putrajaya, Malaysia: Asia-Pacific Society for Computers in Education.

The Effects of Programming Using Collective Intelligence

Kil-Mo KIM[a], Seong-Sik KIM[b]

[a]*Department of Computer Education, Korea National University of Education, Korea*

[b]*Department of Computer Education, Korea National University of Education, Korea*

*kimkilmo@knue.ac.kr

Abstract: A programming education has a favorable influences on creative and logical thinking and problem solving abilities of students. However, students typically have spent too much effort in learning basic grammar and using the model of programming language, which negatively affect their eagerness in learning. In this respect, the purpose of this study is to investigate problem solving and learner's attitude towards collective intelligence in the context of secondary school students' programming classes and to verify the possible application of a new instructional method. The result of collective intelligence showed a positive effect in the attitudes of the students towards learning and problem solving.

Keywords: Collective Intelligence, Programming Attitude, Problem Solving Ability

Introduction

Information has become the most important element in knowledge-based societies. The Internet which was brought about by the development of information and communications technology has caused many changes all over the world particularly in the educational environment. Interests on computers have increased because it can change the traditional learning environment.

Learning how to use computers has been developed differently compared with traditional face-to-face learning in which teachers and learners share limited time and space. The emergence of Internet in the change of educational environment had become a system of administering knowledge which is implemented and eventually brings a significant change inside the classroom. In the Center of cultural phenomenon, a new paradigm of Web 2.0 has been created and called "User participation in the open space". The Web 2.0 provides the foundation that users can interact directly, and have a variety of sharing and spreading of knowledge by a direct connection. The aforementioned phenomenon can be explained by "*Collective Intelligence*"[1].

1. Literature Review

1.1 Scratch Programming Language

The effect of existing programming education has shown a limitation coming from its methodology. Scratch is an easy-to-learn and intuitive Educational Programming Language (EPL) that helps improve the problem solving ability of the class [4][9].

Scratch is developed by the Lifelong Kindergarten Research Group at the MIT Media Lab(http://llk.media.mit.edu)[10].

Scratch is a new programming language that makes it easy to create your own interactive stories, games, music and animations and share your creations with others on the web[13]. Scratch is designed to help young people (ages 8 and up) develop 21st century learning skills. As they create and share Scratch projects, young people learn important mathematical and computational ideas, while learning to think creatively, reason systematically, and work collaboratively. They can create many different types of projects with Scratch[10][14].

1.2 Collective Intelligence(CI)

The *Collective Intelligence*(CI), also called "The wisdom of crowds", or "swarm intelligence"[5], has been recognized as a new value with the advent of Web 2.0.

Collective Intelligence is a shared or group intelligence that emerges from the collaboration and competition of many individuals. Collective Intelligence appears in a wide variety of forms of consensus decision making in bacteria, animals, humans, and computer networks.

Collective Intelligence can also be defined as a form of networking enabled by the rise of communications technology, namely the Internet. Web 2.0 has enabled interactivity and thus, users are able to generate their own content. Further, The *Collective Intelligence* draws on this to enhance the social pool of existing knowledge[2].

Pierre Lévy (1994) has defined *Collective Intelligence* as "distributed everywhere, and is given the value of continuous, real-time adjustments, and the practical ability to bring intelligence"[1]. Lévy and de Kerckhove consider *Collective Intelligence* from a mass communications perspective, focusing on the ability of networked ICT's to enhance the community knowledge pool[2].

The cyberspace is a venue where various people meet and interact with each other. Furthermore, it is where diverse knowledge and information are being discussed. In effect, the more participants joined in cyberspace discussion, the greater the value and space of knowledge will uncover.

Also, James Surowiecki(2004) has defined the *Collective Intelligence* as a moving power in the economy and society. In some situations, a smart group discussion will lead to a wise decision and is better than it could have been made by any single member of the group. For this reason, a man does not need to dominate this group. He said that "The wisdom of crowds" and the like[3].

According to Don Tapscott and Anthony D. Williams(2008), *Collective Intelligence* is mass collaboration. In order for this concept to happen, four principles need to exist. These are openness, peering, sharing and acting globally[6][10][11].

1.3 Design of Collective Intelligence Programming

In this study, the content of *Design of Collective Intelligence Programming* is based on the *Problem-Based Learning model*. And we used modification and supplementation. The validity of *Collective Intelligence Programming* contents according to was 10. Test results were valid. The following Table 1 is *Collective Intelligence programming* contents.

Table 1: *Collective Intelligence Programming* contents

Sessions	**Subjects**	**Contents and Tools**	
1	How to use Web2.0 Tools	Wiki, Mind Map tools, Messenger Program et al.	
2	How to use Scratch	Block, Sound, Motion, et al. (Scratch)	
3	Problem recognition	Learning Objectives and Problem Set (Wiki)	
4	Problem solving planning	Current contents arrangement (Mind manager), The establishment of a plan to solve the problem.	
5	Searching	Data Searching and Data Saving	Wiki, Messenger Program, Scratch's Web
6	Solution	Creative ideas and programming	
7		Programming Analysis and Exchange	
8		Applications of Programming	
9	Presentation and Evaluation	Discussion, Sharing, Review	
10	Publication	Online Publication and Print (Scratch's Web site)	

Also, *Collective Intelligence Programming Learning* has the following effects[8].

- Learners can solve problems by implementing algorithms.
- Teachers are facilitators in programming learning.
- Interaction between learners for effective problem solving.

Figure 1 shows the environment for *Collective Intelligence programming Learning*.

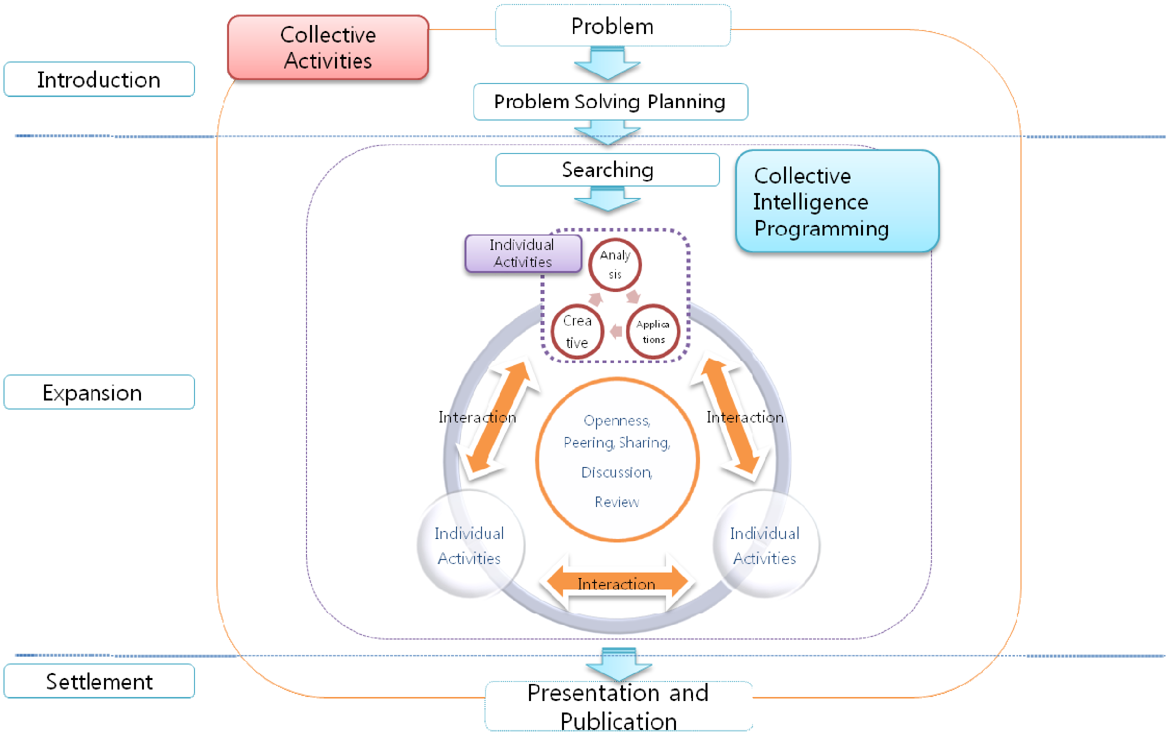

Figure 1: *Collective Intelligence Programming Learning*

2. Methodology

2.1 Samples

There were 73 students in the secondary school who participated in this research. Table 2 presents two research variables (Treatment and Control Group) with the gender information of the variables.

Table 2: General Information of Participants

Group	**Subject**		**Total**	
	Male	**Female**		
Treatment Group(G_1)	22	14	36	73
Control Group(G_2)	22	15	37	

2.2 Design

They were divided into a Treatment Group (G_1), which consisted of students using the *Collective Intelligence programming learning* and Control Group(G_2),which had students using the *Traditional Programming Learning* for comparison. Table 3 illustrates the design of the study.

Table 3: Design

Treatment Group(G_1)	O_1	X_1	O_3
Control Group(G_2)	O_2	X_2	O_4

O_1, O_2: Pre-test (problem solving ability, programming attitude)
X_1: Collective Intelligence Programming Learning
X_2: Traditional Programming Learning
O_3, O_4: Post-test (problem solving ability, programming attitude)

In order to measure learners' achievements, pre- and post-tests were performed before and after the five-week (March 15th to April 16th). They learned the way of using Scratch for two sessions. Subsequently, they learned simple a game programming project in a different way for a total of 8 sessions.

2.3 Test items

- *Problem Solving Ability* test items was conducted by the OECD PISA(Program for International Student Assessment) in 2003. Problem solving ability test items had 19 questions in the area of public. We were used modify 12 questions. Question has been verified of expert group. The result of the pilot test reliability was Cronbach's alpha = .824(n=73).

- *Programming Attitude* test items was conducted by Cho (2008) in the attitude of programming[15]. We were used modify and supplement. Programming attitude test items were verified by the expert group. The result of the pilot test reliability was Cronbach's alpha = .921(n=73).

3. Analyses

In this study, we used SPSS 12.0 for statistical analysis. And the pre-test and post-test was conducted for the analysis of the *Problem Solving Ability* and *Programming Attitude*. Based on this, Independent-Samples t-test and Paired Sample t-test were conducted.

3.1 Problem Solving Ability(PSA)

Pre-test results of *Problem Solving Ability* are shown in Table 4. As the table shows, there are no statistically significant difference between the two groups in either part(p>.05). Therefore, G_1 and G_2 are in the same group.

Table 4: Pre-test results of *Problem Solving Ability*(n=73)

Group	n	Mean	S.D.	df	t	p
Treatment Group(G_1)	36	80.28	11.956	71	.572	.569
Control Group(G_2)	37	78.65	12.365			

P<.05

Post-test results of *Problem Solving Ability* are shown in Table 5. As shown in the table, there are statistically significant differences between the two groups(p<.05). And the mean of the Treatment Group was higher then the Control Group. Therefore, *Collective Intelligence Programming Learning* shows positive influence in enhancing students' *Problem Solving Ability*.

Table 5: Post-test results of *Problem Solving Ability*(n=73)

Group	n	Mean	S.D.	df	t	p
Treatment Group(G_1)	36	87.11	10.725	71	2.234	.029
Control Group(G_2)	37	80.86	13.022			

To know statistically significant difference between the two groups using Paired Sample t-test. As the result are shown in Table 6.

Table 6: Pre-test and post-test Paired sample t-test of PSA(n=73)

Group	**n**	**Pre Problem Solving Ability – Post Problem Solving Ability**			
		Mean	**S.D.**	**T**	**p**
Treatment Group(G_1)	36	-8.677	5.448	-7.525	.000
Control Group(G_2)	37	-3.515	3.895	-3.461	.001

The results of analyses show that the Treatment and Control Group's *Problem Solving Ability* has improved (p<.05). However, the Control Group's improvement of mean was lower than the Treatment Group. The results of the study show that *Collective Intelligence Programming Learning* has positive influence in enhancing students' *Problem Solving Ability.*

3.2 Programming Attitude (PA)

Pre-test results of *Programming Attitude* are shown in Table 7. As can be seen, there are no statistically significant difference between the two groups in either part(p>.05). Therefore, G_1 and G_2 are in the same group.

Table 7: Pre-test results of *Programming Attitude*(n=73)

Group	**n**	**Mean**	**S.D.**	**df**	**t**	**p**
Treatment Group(G_1)	36	58.67	12.708	71	.427	.671
Control Group(G_2)	37	57.46	11.428			

Post-test results of *Programming Attitude* are shown in Table 8. As can be seen, there are statistically significant difference between the two groups(p<.05). And Treatment Group's mean was higher then Control Group. Therefore, *Collective Intelligence Programming Learning* was positive influence in enhancing student's *Programming Attitude*.

Table 1 8: Post-test results of *Programming Attitude*(n=73)

Group	**n**	**Mean**	**S.D.**	**df**	**t**	**p**
Treatment Group(G_1)	36	66.31	10.601	71	2.980	.004
Control Group(G_2)	37	58.24	12.417			

To known statistically significant difference between the two groups using Paired Sample t-test. As the result are shown in Table 9.

Table 2 9: Pre-test and post-test Paired sample t-test of PA

Group	**n**	**Pre Programming Attitude – Post Programming Attitude**			
		Mean	**S.D.**	**T**	**p**
Treatment Group(G_1)	36	-7.639	7.579	-6.048	.000
Control Group(G_2)	37	-.784	2.359	-2.021	.051

The results of analyses show that the Treatment Group’s *Programming Attitude* has improved. But Control group’s *Programming Attitude* has not changed(p>.05). The results of the study show that *Collective Intelligence Programming Learning* has positive influence in enhancing students’ *Programming Attitude*.

4. Conclusion and Discussion

Web 2.0 has attained attention in terms of the flexibility and diversity providing users with various teaching and learning materials. Programming education has favorable influence on creative, logical thinking and problem solving abilities of students. However, students typically have to spend too much effort in learning basic grammar and the usage model of programming language, which negatively affects their eagerness in learning. In this respect, the purpose of this study is to investigate problem solving and learner’s attitude of the *Collective Intelligence programming learning* on Secondary school student’s programming classes and to verify the possible application of this now instruction method. In the results, the *Design of Collective Intelligence Programming learning* was positive influence in enhancing learner’s *Programming Attitude* and *Problem Solving Ability*. In particular, the *Programming Attitude* of students has changed from negative attitude to positive attitude. Through this research, the researcher findings to be bases for a more active participation of student’s in computer field.

References

[1] Pierre Lévy (1994). L' intelligence Collective: Pour une anthropologie de cyberspace. La Decouverte.

[2] Flew, Terry (2008). *New Media: an introduction*. Melbourne: Oxford University Press.

[3] James Surowiecki (2005). *Wisdom Of Crowds*. Random House.

[4] Tony Jenkins (2002). On the Difficulty of Learning to Program. [On-line], <http://www.psy.gla.ac.uk/~steve/localed/jenkins.html>

[5] Wheeler William M’s (2009). *Ants: Their Structure, Development, And Behavior*. Bibliolife.

[6] http://en.wikipedia.org/wiki/Collective_intelligence.

[7] Tapscott, D., & Williams, A. D. (2008). *Wikinomics: How Mass Collaboration Changes Everything*. USA: Penguin Group

[8] Morrison, G. R, Lowther, D.L., and DeMeulle. L. (2005). *Integrating Computer Technology into the Classroom*. New Jersey: Prentice Hell.

[9] Bae, H et al., (2009). A Problem Based Teaching and Learning Model for Scratch Programming Educatin. *The Journal of Korean Association of Computer Education. 12*(3), 11-21.

[10] Lifelong Kindergarten Group, MIT Media Lab. [On-line], <http://scratch.mit.edu>

[11] Jean-François Noubel (2006). Collective Intelligence, The Invisible Revolution. The transitioner.

[12] CSTA(2003). A Model Curriculum for K-12 Computer Science. [On-line], <http://www.acm.org/education/curric_vols/k12finall022.pdf>

[13] J. Maloney, L. Burd, Y. Kafai, N. Rusk, B. Silverman, and M. Resnick(2004). Scratch: A Sneak Preview. In Second International Conference on Creating, Connecting, and Collaborating through Computing, 104-109.

[14] Kafai, Y. B. (in press). Constructionism. In k. Sawyer(Ed.), Cambridge Handbook of the Learning Science. Cambridge, MA: Cambridge University Press.

[15] Cho, S et al., (2008). The effect of CPS-based Scratch EPL on problem solving ability and programming attitude. *The Korean Association of Information Education, 12*(1), 77-88.