International Journal of Education and Development using Information and Communication Technology (IJEDICT), 2015, Vol. 11, Issue 2, pp. 153-167

Poem Generator: A comparative quantitative evaluation of a microworlds-based learning approach for teaching English

Craig Jenkins
University of South Wales, UK

ABSTRACT

This paper is a comparative quantitative evaluation of an approach to teaching poetry in the subject domain of English that employs a ‘guided discovery’ pedagogy using computer-based microworlds. It uses a quasi-experimental design in order to measure performance gains in computational thinking and poetic thinking following a microworld-based intervention in English lessons. Preliminary findings reveal a distinct increase in computational thinking and poetic thinking performance for learners who participated in the intervention. There is also some evidence, though this requires further research, to suggest a relationship between high performance in computational thinking and high performance in poetic thinking.

Keywords*: microworlds, constructionism, computational thinking, english, byob.*

INTRODUCTION

In Wales, the National Literacy and Numeracy framework (LNF) (Welsh Government, 2013) sets a legal requirement for teachers to embed literacy into all lessons. In addition, the recent curriculum review for Wales by Professor Donaldson (2015) recommends a third statutory cross-curriculum responsibility of Digital Competence and the ICT Steering Group (2013) report to the Welsh Government recommends that all learners should be given the opportunity to learn programming.

Possible future changes in educational policy raise a very important question for educators in Wales: how should educational practice adapt in order to integrate digital competence more deeply? One possible response to this question is to employ a microworld-based ‘guided discovery’ approach.

There is, then, a strong and current rationale for examining the link between computational thinking and poetic thinking through the lens of a microworlds-based teaching approach. This paper reports on a comparative quantitative evaluation of the implementation of a microworlds-based approach in the realistic curriculum setting of English. It makes use of a pre-test/post-test using quasi-experimental design in order to collect quantitative data within a real-life educational setting.

Dating back to Papert (1980) and Turtle Graphics, a microworld as a programming environment for ‘guided discovery’ in mathematics has been well documented. With the exception of authors such as Sharples (1985), there has been less coverage of the microworlds-based approach that has focused on the curriculum area of English. The study aims to contribute to research in this area by means of two key objectives:

(1) To examine the potential of a microworld-based teaching approach for the development of computational thinking skills.
(2) To examine the potential of a microworld-based teaching approach for the development of poetic thinking in English.

LITERATURE REVIEW

What is a microworld?

In the 1960s, Papert and his team at the Massachusetts Institute of Technology (MIT) developed a programming language called LOGO. Underpinning this development was a profound new philosophy of how learning happens with computers: a microworld-based approach to learning (see Papert, 1980). For Papert, microworlds provided a means for learners to acquire knowledge in a *natural* way. An example of natural learning commonly used by Papert and other constructionists is early language development in infants which does not apparently require didactic instruction: language develops naturally through immersion within a linguistic environment.[1] Papert thought that his own computer-based microworld, *Turtle Graphics*, provided an environment within which mathematical ideas could develop in a similar way.

As Mitchel Resnick (1997) puts it, 'microworlds are simplified worlds, specially designed to highlight (and make accessible) particular concepts and particular ways of thinking' (p. 50). Turtle Graphics, for Papert and Resnick, provided a paradigmatic microworld where mathematical ideas could develop in a more 'natural' manner, in much the same way as early language learning in the pre-school infant but with well-defined boundaries and constraints. Rieber (1992) uses the phrase *guided discovery* to articulate the microworlds-based learning approach clearly:

> Educational computing […] has much to gain by the infusion of constructivism into instructional design […] The compromise is reached largely through a *guided discovery* [italics in original] orientation to learning in which the nature of the learning activity and experience is naturally constrained by the parameters of the microworld (p. 94).

What is computational thinking?

Wing (2006) explains that computational thinking is 'a fundamental skill for everyone, not just for computer scientists' (p. 33). It is the ability to apply the principles of computer science as a tool for thought in different subject domains far beyond the boundaries of the discipline itself. The ability to use decomposition when facing large problems, to modularize, to think recursively; these are all key tools to be borrowed from the computer science thinking toolkit.

Scratch (Resnick et al., 2004) has become an ubiquitous programming tool that follows a similar format of guided discovery. It was developed by the Lifelong Kindergarten Group at MIT. Resnick and Brennan (2012) have proposed a tripartite framework for describing the different dimensions of computational thinking that are used when programming with *Scratch*. First, there are *computational concepts* such as *sequencing* when breaking down a program step-by-step; *conditionals* for decision making; *operators* for mathematical expressions. Second, there are *computational practices* such as *debugging* for solving problems and *modularizing* for sorting code into different stacks. Finally, there are *computational perspectives* such as *questioning* (interrogating, not naturalizing technology) and *expressing* (computation for design purposes).

Educationalists have begun designing exploratory computer-based environments that specifically target the development of such computational thinking mechanisms. Weintrop and Wilensky's

---

[1] It is worth noting that this is not entirely true since there is an inherent didacticism in early language learning. The idea of immersion in an active and interactive environment is the most important feature, though this rests on underlying developmentalist ideas about natural learning.

(2013) programming game *Robobuilder* is being designed to do exactly that. Their poster proposal points to the potential development of computational mechanisms such as ‘algorithmic thinking’ (a practice of modularizing by decomposing a problem into its smallest parts) and ‘debugging’ (an experimental and cyclical practice of adaptation). The project is speculative to date and has not yet been implemented in an educational setting.

Microworlds for English

In the 1980s, Papert’s Turtle Graphics microworld provided an environment for guided discovery within the domain of mathematics. Robobuilder, when the project is completed, may be thought of as a microworld for computational thinking. In 1985, Sharples created a series of microworld activities with boundaries set within a different subject domain: written English. Sharples (1985) implemented three programs (later in LOGO) that were aligned with this philosophy of exploratory learning and the subject domain of written English. Sharples’ first program was *PAT*: a word pattern generator. Learners first specify a structure of sentence components and a bank of different words. The computer, in turn, generates random phrases and displays them according to the specified sentence structures. By specifying different grammars and then generating words at random to create sentences, learners manipulate language and play experimentally in order to gain an understanding of its linguistic rules (pp. 63 – 64).

In order to evaluate the efficacy of his microworlds, Sharples carried out a feature analysis of essays produced by learners before and after the microworlds-based teaching scheme. The experimental group post-essays revealed a greater increase in the development of ‘mature’ writing features, along with a greater reduction in the use ‘immature’ techniques, than that shown by the control group (*ibid.*, pp. 103 – 104). There is already some evidence, then, that thinking computationally – through learning to program – can have a positive effect on written English.

A fundamental problem shared by both Papert’s Turtle Graphics and Sharples’ PAT, however, is that textual programming languages can prove problematic to the uninitiated learner. Syntax is an important issue with textual programming because, as Rieber (2004) reminds us, microworlds need to be understandable to the learner. Scratch and Robobuilder employ a building-block programming approach (*see* Resnick et al., 2009) that was specifically designed to address the syntactic difficulties encountered with textual programming languages.

This paper reports on the design and implementation of a *Poem Generator* microworld that draws heavily on Sharples’ PAT. Poem Generator is an updated, block-based microworld created using the *Build Your Own Blocks* (BYOB) extension of Scratch (Mönig and Harvey, 2009). Developed by a team at the University of California, the BYOB platform enabled custom programming blocks to be designed that were tailored to the specific curriculum area of English. The microworld was accessed by learners using *Snap!*, a browser-based iteration of the BYOB platform, and was designed in conjunction with a subject-specialist English teacher. The aim was to create a microworld for the development of (i) computational thinking and (ii) English; more specifically poetic thinking.

Further, it is helpful to look at previous academic work to consider other possible limitations of the study. Papert, in *Microworlds* (1980), claimed that learners could improve their problem solving skills by using his own Turtle Graphics microworld. Pea (1984) investigated this claim and suggested that it is the guidance provided by the enthusiastic teacher, not the microworld *per se*, that can lead to such an outcome. Papert (1987), in turn, argued that Pea’s research focused too narrowly on specific indicators of problem solving, therefore omitting other improvements.

Pea’s criticism and Papert’s counter-criticism are equally relevant when considering the limitations of this study. First, this paper has reported on an intervention led by one subject-specialist English teacher with an enthusiasm for the project. Gains in computational thinking and poetic thinking, therefore, may be higher than what could be reasonably expected of other teachers. Second, gains may also be quite understated due to the restrictions imposed by the test designs. All findings, therefore, need to be treated with an appropriate level of academic caution.

Selwyn (2014) warns academics that a more critical approach to educational technology is needed that moves beyond a ‘disdain of formal education’ and the assumption that ‘education is best organized along informal lines of discovery, play and “hard fun”’ (p. 161).[2] Tracing the routes of such tendencies in educational technology to Papert, Selwyn finds an apparent diminishing role of the teacher and classroom that is evident in his writing. What this findings of this research have indicated, however, is that it may in fact be possible to build on Papert’s work effectively. By implementing the microworld as a pedagogical tool in a formal classroom setting, modest improvements were recorded.

METHODOLOGY

Microworld design

In the scripts area (*Figure 1*), learners must first (i) decide what grammar component each box will represent and (ii) populate these boxes with examples of each grammar. Learners achieve this by slotting together a series of pre-built programming blocks with space to add their own text.

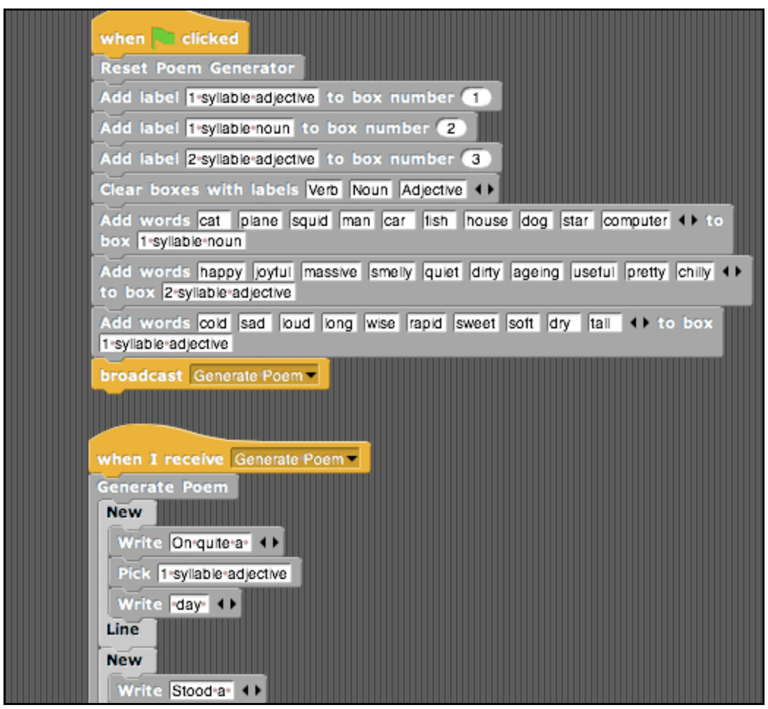


Figure 1: *Screen grab of the scripts area of Poem Generator*

---

[2] See Papert, 2008 for an account of the phrase ‘hard fun’ that Selwyn refers to here.

Learners adopt a similar process to specify the word and line structure for the type of the poem they would like to generate. First, learners add blocks to specify where they would like constant words and random picks from the boxes to appear. This is then used to generate random poems when the program is run.

When they run the poem generator, the outcomes are shown on the stage area (see *Figure 2*). The computer generates lines and entire poems depending on the sequence and input variables specified with the *write* and *pick* blocks. Learners, in turn, are able to observe both the regularities and irregularities of written English.

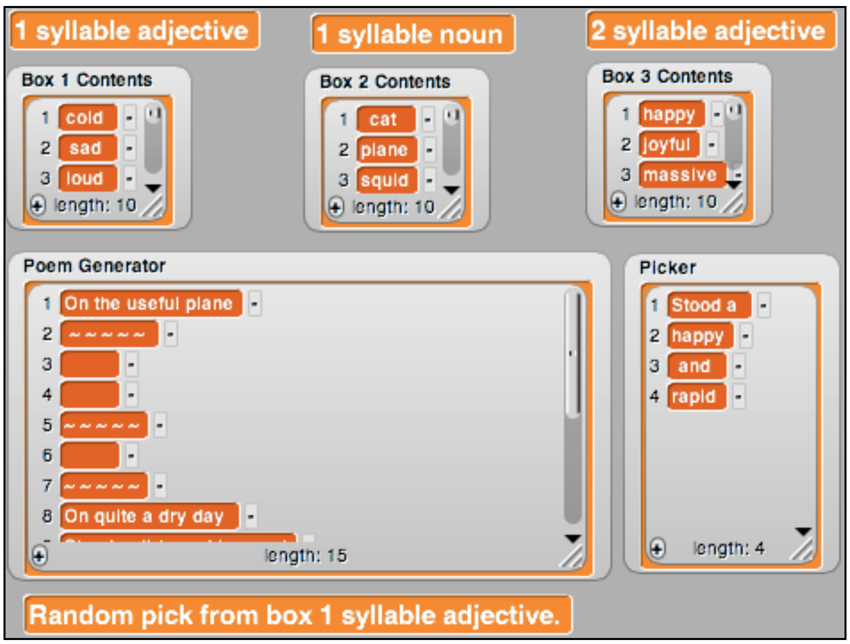


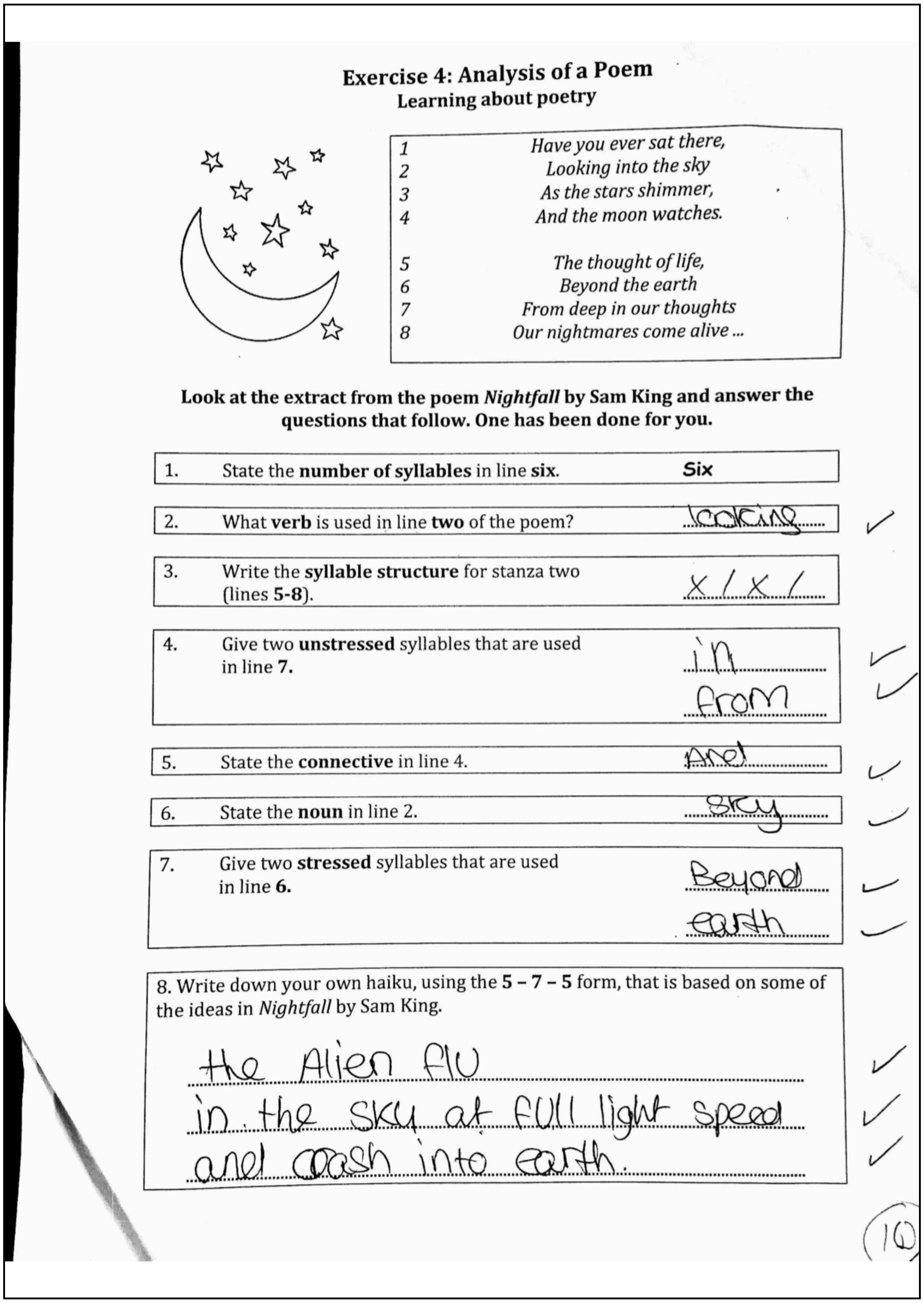
Figure 2: *Screen grab of the stage area of Poem Generator*

To facilitate the ‘low floor’ (see Rieber, 2004) and ease-of-access that is a pedagogic characteristic of the microworlds-based learning approach, learners were provided with two versions of the microworld. The first, shown in *figures one* and *two,* provided a working exemplar of the microworld pre-populated with sample poetic structures and words. Learners were then given blocks in a ‘blank’ microworld to encourage experimentation.

Instrumentation and Schedule

To investigate the research questions, an experimental nonequivalent comparison group design (*see* Cohen, Manion and Morrison, 2011, Locations 14569-14595 of 33832) was used with pre-tests and post-tests for computational thinking and poetry. The pre-tests and post-tests were designed by the author in conjunction with an English-specialist teacher. The difference between pre-test and post-test outcomes provided a means by which to measure any difference in computational thinking and poetic thinking following the microworld intervention with the experimental group. Fully-informed consent was secured from all participants.

Figures 3 and *4* show, respectively, exemplar exercises taken from the poetic thinking and computational thinking tests.



Exercise 4: Analysis of a Poem
Learning about poetry

Have you ever sat there,
Looking into the sky
As the stars shimmer,
And the moon watches.

The thought of life,
Beyond the earth
From deep in our thoughts
Our nightmares come alive ...

Look at the extract from the poem *Nightfall* by Sam King and answer the questions that follow. One has been done for you.

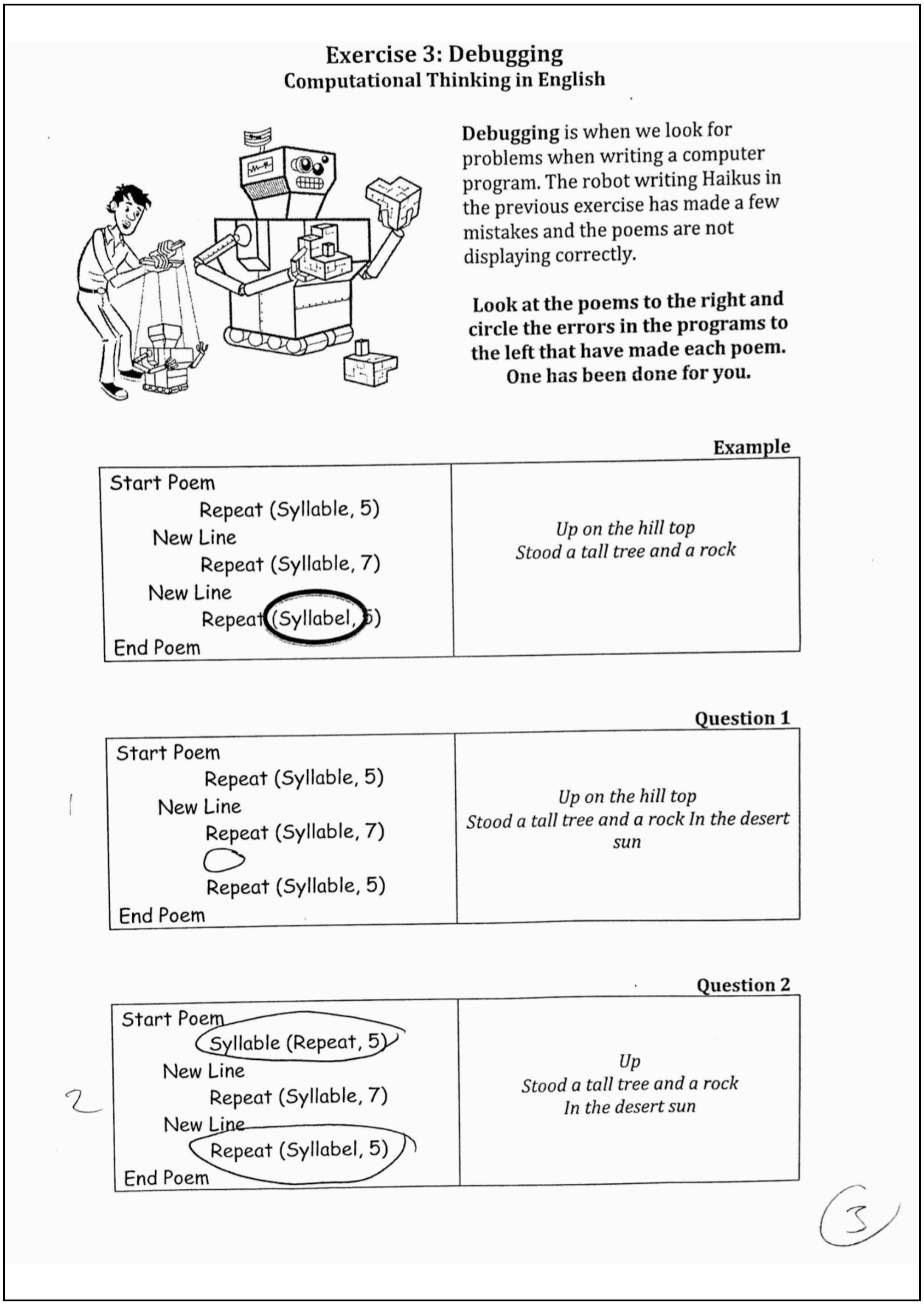
1.	State the **number of syllables** in line **six**.	**Six**
2.	What **verb** is used in line **two** of the poem?	looking
3.	Write the **syllable structure** for stanza two (lines **5-8**).	x / x /
4.	Give two **unstressed** syllables that are used in line **7.**	in from
5.	State the **connective** in line 4.	And
6.	State the **noun** in line 2.	sky
7.	Give two **stressed** syllables that are used in line **6.**	Beyond earth

8. Write down your own haiku, using the **5 – 7 – 5** form, that is based on some of the ideas in *Nightfall* by Sam King.

the Alien flu
in the sky at full light speed
and crash into earth.

10

Figure 3: *Exemplar exercise in the English test*



Exercise 3: Debugging
Computational Thinking in English

Debugging is when we look for problems when writing a computer program. The robot writing Haikus in the previous exercise has made a few mistakes and the poems are not displaying correctly.

Look at the poems to the right and circle the errors in the programs to the left that have made each poem. One has been done for you.

Example

Start Poem Repeat (Syllable, 5) New Line Repeat (Syllable, 7) New Line Repeat (Syllabel, 5) End Poem	*Up on the hill top* *Stood a tall tree and a rock*

Question 1

Start Poem Repeat (Syllable, 5) New Line Repeat (Syllable, 7) Repeat (Syllable, 5) End Poem	*Up on the hill top* *Stood a tall tree and a rock In the desert sun*

Question 2

Start Poem Syllable (Repeat, 5) New Line Repeat (Syllable, 7) New Line Repeat (Syllabel, 5) End Poem	*Up* *Stood a tall tree and a rock* *In the desert sun*

Figure 4: *Exemplar exercise in the computational thinking test*

The pre-tests/post-tests for poetic thinking consisted of five exercises mapped to the 'adapt structures in writing, 'use a wide range of sentence structures' and 'use knowledge of word roots and families' aspects defined in the statutory literacy cross-curriculum responsibilities set out by the Welsh Government (2013). For computational thinking, five exercises were mapped to the programming concepts of sequences, loops and events that are defined in Resnick and Brennan's (2012) framework for computational thinking. In order to avoid the problem of pre-test/post-test equivalency, an identical set of tests were used for pre-tests and post-tests. Gain (difference between pre-test and post-test scores) was measured instead of post-test performance.

The study took place over the course of one half term at a secondary school in South Wales; the implementation of the microworld intervention was as follows. At the beginning of the half term, both the experimental and comparison groups were issued with identical paper-based computational thinking and poetry assessments that were co-authored by the author and the English subject-specialist teacher. Identical assessments were re-issued to both groups at the end of the half term. These assessments took place in the standard teaching room for both groups.

The microworld intervention was carried out over three one-hour lessons in a bookable ICT room during normal timetabled English lessons. The intervention was led by the English teacher after receiving training from the author and took place in a bookable ICT suite at the school. The author was available throughout in a technical support capacity. The comparison group made their own paper-based haiku poetry in their standard teaching rooms whilst the intervention group undertook the computer-based microworld activities.

Both groups followed the scheme of work (SoW) unit for 'Year 8 poetry' that was devised by the school. The SoW unit devised by the school aimed to teach about different word classes and their use in common poetry forms, with a particular emphasis on the haiku form. Learners were first required to study the haiku poetry form by looking at existing poems and were then asked to write their own haikus according to the structures they have studied throughout the unit. The unit linked to the following areas of the KS3 programme of study for English in the National Curriculum for Wales: (i) 'use the standard forms of English: nouns, pronouns, adjectives, adverbs, prepositions, connectives and verb tenses'; (ii) 'experiencing and responding to a wide range of texts that include [...] traditional and contemporary poetry' (DCELLS, 2008).

Participants

The sample consisted of an experimental group made up of 69% boys (n = 9) and 31% girls (n = 4) and a comparison group made up of 57% boys (n = 8) and 43% girls (n = 6). Classes were selected in an attempt to control for external factors as much as possible, whilst also maintaining a natural school setting and class dynamic. To control for external factors, the two English classes were: (i) from the same cohort – Year 8; (ii) following the same scheme of work - poetry; (iii) taught by the same English teacher; (iv) of a similar low-ability academic set; (v) made up of learners from comparable ethnic and socioeconomic backgrounds.

A recent Estyn (2010) inspection report provides an indication of the context of the school population from which the samples were taken. The school is a large school with over 1,500 learners on roll at the date of the last inspection. Using free school meals (FSM) as an indicator of socioeconomic disadvantage, in 2009/10 eligibility was 36.8%. This is significantly greater than the national comparator of 17.1%. A high number of learners (68%) were from homes where languages are spoken other than English or Welsh. As such, there was a high level of English as an additional language (EAL) support provision in place. The percentage of learners appearing on the register for special educational needs (SEN) is 28.6%. This is significantly higher than the national comparator of 20.9%.

Data Analysis

Data analyses were carried out using IBM SPSS. Following initial univariate analyses, independent t-tests were used alongside Cohen's D to compare the performance of the groups. Pearson's r was calculated alongside an F-test to look for a relationship between high scores in *computational thinking* and *English*. Finally, a regression model investigated the relationship between predictors *group* and *computational thinking gain* and the dependent *English gain*.

FINDINGS AND DISCUSSION

Univariate analysis

Table 5 shows the mean pre-test and post-test raw scores (and calculated gains) for both the English (poetic thinking) and computational thinking (CT) tests.

Table 5: Comparison of means

			Computational Thinking			English		
Group	Gender		CT pre-test	CT post-test	CT gain	English pre-test	English post-test	English gain
Experimental	Boy	Mean	14.6667	17.8889	3.2222	21.1111	25.0000	3.8889
		N	9	9	9	9	9	9
		Std. Deviation	3.53553	2.36878	3.15348	3.14024	3.87298	4.70225
	Girl	Mean	9.5000	12.5000	3.0000	20.0000	24.0000	4.0000
		N	4	4	4	4	4	4
		Std. Deviation	4.20317	5.06623	1.41421	3.36650	4.08248	4.24264
	Total	Mean	13.0769	16.2308	3.1538	20.7692	24.6923	3.9231
		N	13	13	13	13	13	13
		Std. Deviation	4.34859	4.10597	2.67227	3.11325	3.79440	4.38675
Comparison	Boy	Mean	12.8750	15.0000	2.1250	20.7500	23.2500	2.5000
		N	8	8	8	8	8	8
		Std. Deviation	4.76408	2.77746	3.27054	5.36523	1.90863	4.44008
	Girl	Mean	17.5000	19.0000	1.5000	25.0000	26.5000	1.5000
		N	6	6	6	6	6	6
		Std. Deviation	2.07364	1.54919	.83666	3.03315	1.64317	2.42899
	Total	Mean	14.8571	16.7143	1.8571	22.5714	24.6429	2.0714
		N	14	14	14	14	14	14
		Std. Deviation	4.41775	3.04905	2.47626	4.87875	2.40535	3.62607

The table shows that, between taking the pre-test and the post-test, learners who did not receive the microworld intervention improved by 2.5 raw marks on average in computational thinking. Learners who did receive the microworld intervention improved by 2.7 raw marks. This means that learners following a microworld-based teaching scheme showed an improvement in their post-test scores that was 5.64% higher.

For poetic thinking in English, the comparison group improved by an average of 3.6 raw marks whereas the microworld group improved by an average of 4.4 marks. This produced a similar difference in performance. Learners following a microworld-based teaching scheme saw a modestly greater improvement, by 5.61%, than their counterparts in the comparison group.
This initial analysis is promising with an increase in performance of around 5-6% in both computational thinking and poetic thinking for those following the microworld-based teaching scheme. It is important to note, however, that the standard deviations are quite restricted in places. This indicates that, in future experiments, there is a need to look again at the tests used in order to ensure a more robust set of outcomes.

It is also important to note two further limitations resulting from variables that were not controllable. First, the well-documented Hawthorne Effect[3] could have resulted in an atypical higher level of performance. Second, the experimental and comparison groups were comprised of two low-ability English sets. The English sets were made up of students that also belonged to a mix of discrete ICT sets. A variability in ICT teaching and exposure to the Scratch programming environment, along with the different ICT capability of learners, delimits the findings of the study since sampling only accounted for English ability.

Bivariate analysis: independent samples t-test

Two independent samples t-tests were carried out *(see table six)* in order to assess the confidence of the changes that were measured. Both the change in computational thinking ($p = 0.202$) and poetic thinking ($p = 0.242$) produced close to an 80% confidence level for change. This was particularly pleasing given the small sample size that made achieving statistical significance and a pre-defined *p* value difficult.

Further, Cohen's effect size value was calculated in order to measure the strength of the effects. Following the microworld intervention, a modest-to-moderate effect size ($d = 0.50$) was reported in computational thinking and a modest effect size ($d = 0.46$) in poetic thinking. Potentially, then, the data reveals a positive effect of the microworld intervention in terms of both computational thinking and poetic thinking in English.

---

[3] See *Macefield, 2007* for more on the Hawthorne Effect phenomenon.

Table 6: *Independent samples t-test*

		Levene's Test for Equality of Variances		t-test for Equality of Means						
									95% Confidence Interval of the Difference	
		F	Sig.	t	df	Sig. (2-tailed)	Mean Difference	Std. Error Difference	Lower	Upper
CT gain	Equal variances assumed	.305	.585	1.309	25	.202	1.29670	.99072	-.74373	3.33714
	Equal variances not assumed			1.305	24.429	.204	1.29670	.99363	-.75214	3.34555
English gain	Equal variances assumed	.249	.622	1.199	25	.242	1.85165	1.54422	-1.32873	5.03203
	Equal variances not assumed			1.190	23.373	.246	1.85165	1.55546	-1.36322	5.06652

Bivariate analysis: Pearson's r

A subsidiary analysis was carried out in order to examine whether a large improvement in computational thinking was also associated with a large improvement in English. This analysis incorporated results from both the microworld intervention group and the comparison group. The result of this calculation is shown in *Table 7*.

Table 7: *Pearson's r calculation*

		English gain	Computational thinking gain
English gain	Pearson Correlation	1	.341
	Sig. (2-tailed)		.082
	N	27	27
Computational thinking gain	Pearson Correlation	.341	1
	Sig. (2-tailed)	.082	
	N	27	27

The analysis revealed that there was a modest-to-moderate positive correlation (r = 0.341) between improvement in computational thinking and improvement in poetic thinking. This finding is supported by a reasonable level of confidence that is greater than 90% (p = 0.082). These results suggest the possibility of a link between poetic thinking and computational thinking, though this may merely indicate high gains in both tests. According to the results, those who improved the most in the post-test for computational thinking also improved the most in the post-test for poetic thinking.

Multivariate analysis

A multiple linear regression model was used to examine whether (i) being part of the microworld intervention group and (ii) making an higher improvement in computational thinking served as combined predictors of higher performance in poetic thinking. *Tables eight* and *nine* show the results of these calculations.

Table 8: *Model summary*

Model	R	R Square	Adjusted R Square	Std. Error of the Estimate
1	.373[a]	.139	.068	3.90368

Table 9: *Standardized coefficients*

Model		Unstandardized Coefficients		Standardized Coefficients	t	Sig.
		B	Std. Error	Beta		
1	(Constant)	1.204	1.186		1.015	.320
	CT gain	.467	.304	.301	1.540	.137
	Experimental group	1.246	1.554	.157	.801	.431

Looking at the standardized coefficients for the two variables individually, making a high improvement in computational thinking (B = 0.301) seemed to have a stronger effect on poetic thinking than being part of the microworld intervention group (B = 0.157). The stronger comparative effect size when using computational thinking performance as a predictor is also matched with a higher level of confidence.

These results suggest that making an improvement in computational thinking had a larger effect on improvement in poetic thinking than following a microworlds-based teaching scheme. It is important to note, however, that the combined effect suggests a poor overall fit of the model to the data (R^2 = 0.139, adjusted R^2 = 0.068). The combined effect is also not statistically significant.

CONCLUSIONS

The findings of the study reveal a number of important results for educators to consider. There are potential quantifiable gains in performance that could be achieved as a result of incorporating elements of a microworld-based pedagogy into classroom practice. It is clear, with close to 80% confidence, that learners who received the microworld-based intervention in teaching practice made a higher improvement in computational thinking and poetic thinking than their counterparts who did not. Further, the effect size of this was modest-to-moderate in both cases.

The effect of the *Snap!* microworld-based intervention upon poetic thinking is consistent with an earlier study by Sharples (1985) that took place using the LOGO programming language. Whereas Sharples recorded an increase in 'mature' writing features, this study has recorded a greater gain in poetic thinking tests for 'sentence structures' and 'word roots and families'. Both studies, though using different instruments, saw larger – though not statistically significant - gains in aspects of written English following a computer-based programming intervention.

The effect of the intervention upon computational thinking is interesting because this contradicts previous research carried out by Pea (1984), which suggests that problem solving skills were not improved by programming in LOGO. It is important to note, however, that computational thinking and problem solving – though related – are far from synonymous. The post-tests in this study measured specific computational thinking concepts of sequences, loops and events. The earlier study measured generic problem solving skills. It may also be the case that visual block-based programming environments, such as *Snap!*, are more powerful at improving thinking skills than text-based programming languages such as LOGO. This, however, requires further research as this study provides no evidence to support this.

The study reports, with 90% confidence, that a positive correlation between performance gains in computational thinking and gains in poetic thinking. It is important to note that this may simply indicate a high performance in both tasks. The notion of a link between computational thinking and poetic thinking is an interesting element that requires further research coverage.

The quasi-experimental design used here, though useful at embedding the findings within a classroom context, did not allow for random sampling. It is for this reason that a non-equivalent *comparison* group was used in lieu of a *control* group. Nevertheless, as the methodology states, every effort was made to control for external factors such as ability level and scheme of work.

Further experiments need to be carried out with different samples of learners, within a different school, in order to evaluate the findings of this paper further. There are two key problems that need to be addressed. First, the standard deviations of the test scores indicate that further refinement of the quantitative assessment instruments are necessary to ensure that a robust set of differentiated outcomes is achieved. Second, there is a need to report on the use of qualitative techniques, such as questionnaire and observation, in order to gain deeper insights into the microworlds-based teaching approach and any link that may exist between computational thinking and poetic thinking.

REFERENCES

Cohen, L., Manion, L. and Morrison, K. (2011) *Research Methods in Education*. 7th edn. Kindle format [e-book reader]. Available at http://www.amazon.co.uk (Accessed 6 November 2013).

Department for Children Lifelong Learning and Skills. (DCELLS) (2008) *English in the National Curriculum for Wales.* Cardiff: Welsh Assembly Government.

Donaldson, G. (2015) *Independent Review of the Curriculum and Assessment Arrangements in Wales*. Available at: http://gov.wales/docs/dcells/publications/ 150317-successful-futures-en.pdf (Accessed: 13 May 2015).

Estyn. (2010) *Inspection under section 10 of the Schools Inspection Act 1996: School X.* Cardiff: Estyn.

ICT Steering Group (2013) *The ICT Steering Group's Report to the Welsh Government*. Available at: http://learning.wales.gov.uk/docs/learningwales/publications/ 131003-ict-steering-group-report- en.pdf (Accessed: 7 November 2013).

Macefield, R. (2007) 'Usability Studies and the Hawthorne Effect', *Journal of Usability Studies*, 2(3), pp. 145-154.

Monig, J. and Harvey, B. (2009) 'Bringing "No Ceiling" to Scratch: Can One Language Serve Kids and Computer Scientists?', *Constructionism 2010*, Paris, France, 16-21 August. Paris, France: AUP [Online]. Available at: http://www.eecs.berkeley.edu/ ~bh/BYOB.pdf (Accessed 29 October 2013).

Muijs, D. (2011) *Doing Quantitative Research in Education with SPSS.* 2nd ed. London: Sage.

Papert, S. (1980) *Mindstorms: Children, Computers, and Powerful Ideas*. New York: BasicBooks.

Papert, S. (1987) 'Information Technology and Education: Computer Criticism Vs. Technocentric Thinking', *Educational Author*, 16(1), pp. 22–30.

Papert, S. (2008) *Hard Fun*. Available at http://www.papert.org/articles/HardFun.html. (Accessed: 5 June 2012).

Pea, R. (1984) 'Symbol Systems and Thinking Skills', *Proceedings of 1984 National Logo Conference, Cambridge, MA,* 26–29 June. Cambridge, MA: MIT [Online]. Available at: http://www.stanford.edu/ ~roypea/RoyPDF%20folder/A19_Pea_84.pdf (Accessed: 27 October 2013).

Resnick, M. (1997) *Turtles, Termites, and Traffic Jams: Explorations in Massively Parallel* Microworlds. Massachusetts: MIT.

Resnick, M., Kafai, Y., Maloney, J., Burd, L., Rusk, N. and Silverman, B. (2004) 'Scratch: A Sneak Preview', *Second International Conference on Creating, Connecting and Collaborating through Computing,* Kyoto, Japan, 29–30 January. Kyoto, Japan: C5'04 [Online]. Available at: http://llk.media.mit.edu/projects/scratch/ ScratchSneakPreview.pdf (Accessed: 27 October 2013).

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E. Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B. and Kafai, Y. (2009) 'Scratch: Programming for All', *Communications of the ACM*, 52(11), pp. 60-67.

Resnick, M., Maloney, J., Monroy-Hernández, A., Rusk, N., Eastmond, E. Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B. and Kafai, Y. (2009) 'Scratch: Programming for All', *Communications of the ACM*, 52(11), pp. 60-67.

Resnick, M. and Brennan, K. (2012) 'New Frameworks for Studying and Assessing the Development of Computational Thinking', *Proceedings of the 2012 Annual Meeting of the American Educational Research Association*, Vancouver, Canada, 13-17 April. Washington, DC: AERA [Online]. Available at: http://web.media.mit.edu/~kbrennan/ files/Brennan_Resnick_AERA2012_CT.pdf (Accessed 28 October 2013).

Rieber, L. (1992) 'Computer-Based Microworlds: A Bridge Between Constructivism and Direct Instruction', *Educational Technology Research and Development,* 40(1), pp. 93–106.

Rieber, L. (2004) 'Microworlds' in Donassen, D. (ed.) *Handbook of Research for Educational Communications and Technology.* 2nd ed. Mahwah, NJ: Lawrence Erlbaum Associates, pp. 583-603.

Selwyn, N. (2014) *Distrusting Educational Technology: Critical Questions for Changing Times*. London and New York: Routledge.

Sharples, M. (1985) *Cognition, Computers and Creative Writing*. West Sussex: Ellis Horwood.

Weintrop, D. and Wilensky, U. (2013) 'RoboBuilder: A Computational Thinking Game', *Proceedings of the 44th SIGCSE Technical Symposium on Computer Science Education*, Denver, CO, 6–9 March. New York: ACM [Online] Available at: http://ccl.northwestern.edu/papers/2013/ RB_Comp_Thinking_Game.pdf (Accessed: 27 October 2013).

Welsh Government (2013) *National Literacy and Numeracy Framework*. Available at: http://learning.wales.gov.uk/resources/browse-all/nlnf/?lang=en (Accessed 26 December 2014).

Wing, J. (2006) 'Computational Thinking', *Communications of the ACM*, 49(3), pp. 33–35.

Original article at: http://ijedict.dec.uwi.edu/viewarticle.php?id=1972