e-Bangi Faculty of Social Sciences and Humanities Universiti Kebangsaan Malaysia
Journal of Social Sciences and Humanities

Vol. 16. No.7 (1-12) ISSN: 1823-884x

ALGORITHM AS A PROBLEM SOLVING TECHNIQUE FOR TEACHING AND LEARNING OF THE MALAY LANGUAGE

Faridah binti Nazir, Zanariah Jano & Norliza Omar

ABSTRACT

Computational thinking or CT refers to the thought processes involved in expressing solutions as computational steps or algorithms that can be carried out by a computer. CT is not merely using software or computer programs alone but is also interlinked with information and communication technology (ICT) and computer science based on problems; tools; action (praxis); solutions; concepts; and social needs. Hence, the aim of the study was to determine the effectiveness of using scratch program to teach Malay language subject. As part of the methodology, 32, 4th year students in a primary school of Seremban district participated in the study. A case study was used. The teachers teach Malay language subject using scratch programs with the theme of agriculture and title of food product from agriculture. The students were to arrange the basic words into grammatical sentences based on the scratch program and write a cake recipe using an algorithmic technique, a series of instructions contained in the scratch program. This conceptual teaching and learning algorithm was conducted in five steps namely the induction set; step 1; step 2; step 3; and enrichment and recovery. The result of this study shows that students could understand the concepts of algorithms taught and did not require recovery activities. This implies that the algorithm concept of how to prepare a cake through a scratch program has been fully understood by the students. Students are also excited about the algorithmic techniques and the scratch program generated. As a conclusion, the student's reaction as they remain focused throughout the session shows that the use of algorithms and scratch programs is better than conventional methods in teaching the Malay Language.

Keywords: Scratch program; Algorithm; Basic words; Grammatical; Instruction

INTRODUCTION

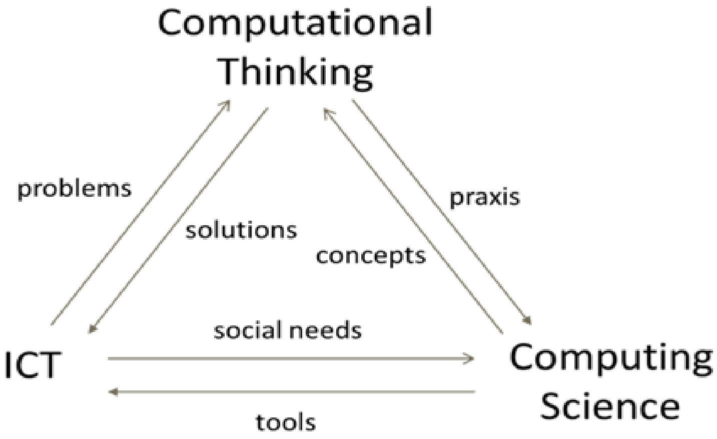
Computational Thinking (CT) is a new approach in solving problems. This approach uses the methods in Computer Science to solve problems or execute a task (C.A.R. Hoare and N. Wirth, 1972). Computational Thinking (CT) is a problem solving process that includes a number of characteristics and dispositions. CT is essential to the development of computer applications, but it can also be used to support problem solving across all disciplines, including the humanities, math, and science. Students who learn CT across the curriculum can begin to see a relationship between academic subjects, as well as between life inside and outside of the classroom. Some of the elements of CT, including:

a) Decomposition: Breaking down data, processes, or problems into smaller, manageable parts.
b) Pattern Recognition: Observing patterns, trends, and regularities in data
c) Abstraction: Identifying the general principles that generate these patterns
d) Algorithm Design: Developing the step by step instructions for solving this and similar problems

A complex problem is one that, at first glance, we don't know how to solve easily. Computational thinking involves taking that complex problem and breaking it down into a series of small, more manageable problems (decomposition). Each of these smaller problems can then be looked at individually, considering how similar problems have been solved previously (pattern recognition) and focusing only on the important details, while ignoring irrelevant information (abstraction). Next, simple steps or rules to solve each of the smaller problems can be designed (algorithms). Finally, these simple steps or rules are used to program a computer to help solve the complex problem in the best way.

Nevertheless, computational thinking refers to the thought processes involved in expressing solutions as computational steps or algorithms that can be carried out by a computer. (Cuny, Snyder, & Wing, 2010; Aho, 2011; Lee, 2016). Computational thinking or CT is not merely using software or computer programs alone but is also interlinked with information and communication technology (ICT) and computer science based on problems; tools; action (praxis); solutions; concepts; and social needs as shown in Figure 1, the Triangle of Computational Thought.

Figure 1: Triangle of Computational Thinking



Source: C.A.R. Hoare and N. Wirth (1972)

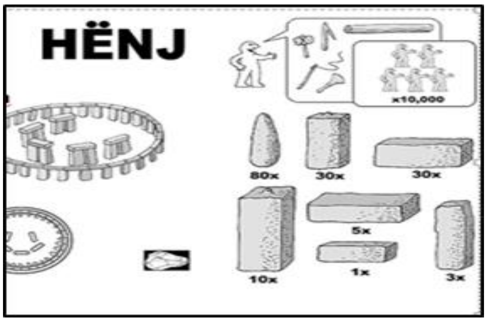
According to N. Wirth (1973), CT is a combination of Algorithms and Data Structures that produces the Program. Thus, CT is a combination of information basing on computing work to solve problems. The most difficult part of CT is characterizing a problem. Four main techniques are utilized in characterizing CT-based problems, namely:

a) Decomposition
b) Pattern recognition
c) Pattern generalisation/abstraction
d) Algorithm design

Decomposition

Decomposition refers to the identification of information needed to solve a problem. Information is summarized into multiple fractions. Furthermore, the breakdown of the information is identified to address the breakdown of the problem as shown in Figure 2.

Figure 2: Decomposition



Source: N. Wirth (1973)

Pattern Recognition

Pattern recognition comprises two aspects: to identify patterns in the problem; identify patterns in the information. When identifying patterns in the problem, two questions must be asked namely,

a) Have I seen or experienced such problems before?
b) How was this problem different from my previous problem ?.

Accordingly, four questions are asked on the pattern contained in the information:

a) How is the structure of this information?
b) Is there any relevant links in this information?
c) Have I seen this information in such pattern before?
d) How is this information different from the information I have ever seen?

Pattern Generalization

Pattern generalisation or Abstraction needs two questions and sub-questions as follows:

a. What is the main issue in this problem?
   i. Which part is static and how are the parts in the problem / sub problem arranged?
   ii. Which part is changed?
b. What is the information pattern in the problem?
   i. Which information is static?
   ii. Which information is changed?

Algorithm Design

Clear CT instruction to solve problems needs algorithm as follows:

a) What are the steps contained in the information from the beginning until the end stage of problem solving?
b) How do these problem parts link to each other?
c) How does the information change in every step?

CT focuses on problem solving ranging from information, computing to the information structuring instead of computer programs like decomposition; abstraction; patterns; and algorithm. CT begins with a concrete assumption of a problem until the problem is solved using the algorithm set instruction. The objectives of this study are as follows:

a. To determine the effectiveness of using scratch program in teaching Malay language in ways that students are able to:
   i. write algorithm set instruction using a single sentence using scratch program.
   ii. produce a cake recipe using an algorithmic technique, a series of instructions contained in the scratch program.

Hence, the research questions are as follows:

a. How effective are the scratch program in teaching Malay language?
   i. Are students able to write algorithm-based instructions using single and plural sentences based on scratch program?
   ii. To what extent students can produce a cake recipe using the algorithm set instruction in the scratch program?

REVIEW OF LITERATURE

The CT approach began in schools in the Europe when the students compared the events that occur in the life of ancient Rome children with their own lives. These students also recorded a series of instructions in the life of the ancient Roman children. Teachers also emphasized role-playing and simulation methods for the students to better understand the series of instructions available. Teachers also guided students to apply the good values of ancient Rome children's lives, into the lives and careers of their future students (Barr, D, Harrison, J, Conery, L. (2011). The instruction series produced by students was an algorithm in CT approach.

There have been several literary style reviews and overview's written on the current state of CT in schools in specific countries as well as suggestions for frameworks for it. The following section presents several of these which were found during the search process. Qualls & Sherrell (2010) presents a brief overview of some examples of how CT has been used and gives evidence as to why CT should be integrated into the curriculum. In the study, some examples are given of CS courses that are adapted to introduce CT, interdisciplinary attempts to integrate CT as well as standalone CT college courses are presented. The authors discuss the Alice programming language and how it is a popular choice in college courses as well as describing the Computer Science for High Schools program developed by Carnegie Mellon University, one of the first major attempts to spread CT.

Schulte et al. (2012) present an overview of a working-group report on CS at school conducted in 2012. This was an international effort which included a workshop at the Koli Calling conference in 2011 and an online survey which was analysed. They emailed the survey to known experts as well as a variety of CS education mailing lists. They received 84 respondents from 22 countries and included a variety of institutional backgrounds including school, industry and university. Based on an initial survey of the results they used a SWOT (Strengths, Weaknesses, Opportunities and Threats) inspired analysis and some of their findings were:

a. CS is most often available in upper secondary school
b. In upper secondary school the most relevant topics were rated as:
   i. Introductory programming (rated much higher than all other topics)
   ii. Algorithms
   iii. Advanced programming
   iv. Programming project

v. HTML

c. For upper secondary school the top goals of CS were:
   i. Developing thinking skills
   ii. Developing problems solving skills
   iii.Learning programming
   iv.Improving algorithmic thinking
   v. Databases: design and queries

d. For primary school, most topics weren't seen as important with applications being the only moderately important one followed by ethics & privacy.

e. For all levels of teaching, the following teaching methods are listed in order of importance:
   i. Classroom based
   ii. Using standard applications like Microsoft Word
   iii. Email
   iv. Individual and small group work
   v. Programming, projects

f. Suggested problems related to CS were:
   i. A lack of trained teachers
   ii. The perception that CS is the same as ICT

g. Trends with CS:
   i. Many noted that new CS curricula were coming in in the next 5-10 years
   ii. There is more CS in high school and an increasing demand for it

h. Teaching education goals:
   i. Basic concepts of CS
   ii. CS education goals, Education/Pedagogy, CS specific teaching approaches, the nature of CS

i. Problems with CS teacher education:
   i. Lack of existing teachers in the subject
   ii. Very little teacher education to teach CS in schools
   iii. No CS in curriculum

j. Trends with CS teacher education:
   i. Technology will evolve - This includes an increase in the variety of ways computers Are used at schools,and the new ways of development of learning materials
   ii. CS becoming recognised
   iii. Teachers being trained in CS

Algorithm refers to a series of commands or set of rules to execute an assignment. MDEC and Jag Systems (2016) have categorized algorithms as online activities (i.e, scratch program) or offline (unplugged activity). The offline activity is to create a monster face and appearance through a series of algorithm instructions, namely the algorithm 1 is about the the face and the giant form and the algorithm command 2 is how to paint the monster as follows:

Algoritma 1 is about my monster face and appearance:

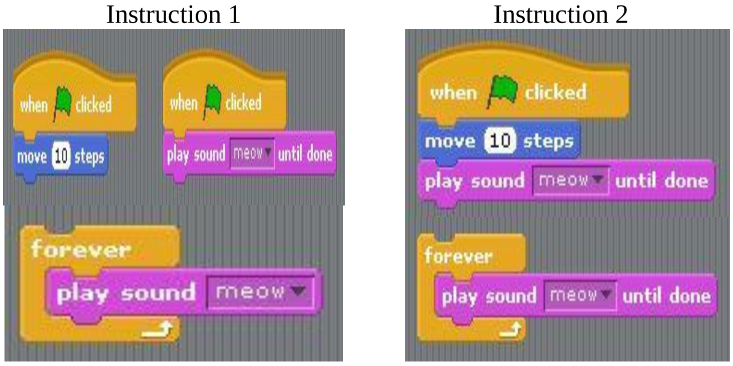
a. ____
b. ____

Algoritma 2 is on how to paint my monster:

a. ____
b. ____

The scratch program is ideal for algorithmic concepts application in CT approach. Sprite or character in the scratch program is moved through a series of algorithm instructions as shown in Figure 3. The computer program also uses an abstract concept to solve a problem. Through the abstract concept, only a series of important commands are presented, while non-essentials are eliminated (Csizmadia, A and et al., 2015).

Figure 3: Scratch Program Algorithm Instructions Series



Instruction 1

```
when 🏁 clicked
move 10 steps

when 🏁 clicked
play sound meow until done

forever
    play sound meow
```

Instruction 2

```
when 🏁 clicked
move 10 steps
play sound meow until done

forever
    play sound meow until done
```

Source: Researcher

Natasha Nesiba and Enrico Pontelli (2015) have carried out a project known as DISSECT (DIScover SciEnce through Computational Thinking). The aimed of this project is to introduce students to computer science principles by establishing computational thinking (CT) as a problem-solving technique within middle school and high school Science, Technology, Engineering, and Mathematics (STEM) courses. While DISSECT has shown successful integration of CT into middle school and high school STEM curricula, illustrating the pervasive nature of CT, a question remained; "can CT also be infused into humanities courses (e.g., English, Art, History) in addition to scientific courses (e.g., Chemistry, Biology, Computer Science)?" The answer is positive. The objective of this research is to present one approach to bridge the gap between CT and humanities through the curriculum of a 12th-grade English Literature course. The course blends CT practices with composition and literature to provide students with the ability to write critical and comparative analyses of selected literature. This research also describe multiple modules that integrate computational thinking into the course, and discuss the results and assessment tools used to measure student competency in computational thinking.

RESEARCH METHODOLOGY

A survey was conducted with the observation technique among 32, 4th year students at a Seremban district primary school. Teachers taught Malay Language subjects using scratch programs. The theme was agriculture, and the title was food products from agriculture. This teaching and learning session had an objective of being able to arrange the words into grammatically single and plural sentences based on the scratch program based on the following Teaching Plans (RPH):

Table 1: Teaching Plans

Lesson Name	Agriculture
Theme	Agricultural-based Food
Duration	60 mins.
Standard Content	3.3 Constructing and writing words, phrases, and sentences correctly
Standard Outcome	3.3.4 Constructing and writing single and compound sentences based on graphical materials correctly.
Existing knowledge	Students have learned single and compound sentences.
Objectives	• To arrange the given words to be a single sentence based on the graphic material correctly. • To write instructions using single and compound sentences to produce cake recipes based on the scratch program.
Materials and tools	• Picture • Computer • Scratch Program

Steps:

Activation			**Duration***:*
Induction Set	Chicken Life cycle	1. Students peruse the given picture. 2. Students construct single sentences orally on the picture 3. Teacher assists the students in constructing the sentences.	4 mins **Decomposition**
Step 1	1.Oral conversation in the *scratch* program. 2.Agricultural based ingredients in the cake: • Eggs • Butter	1.Teacher shows the conversation using the *scratch* program. 2. Students are to read the script of conversation and identify agricultural-based ingredients in a cake. 3. Students have a question and answer session with teacher on the agricultural-based ingredients in a cake.	6 mins **Abstraction**
Learn By Doing			**Duration**

Step 2	- *Scratch* Program - Mahjung Paper	1. Students are to discuss in small groups on how to write instructions on steps of baking cake 2.Students scrutinize scratch program conversations and write steps to prepare cakes in the mahjung paper. 3. A representative of each group present the results of their discussions to the class.	(10 mins) **Inquiry Based Learning** **Algorithm**
Integration			**Duration**
Step 3	- Mahjung Paper	1. Teacher compares the presentation of each group. 2. Teacher explains if there are groups that make mistakes in arranging steps in baking a cake. 3. Teacher associates students' assignments with cake baking activity in real life.	(5 mins) **Evaluation**
Summarize			**Duration**
Closure		1. Teacher puts emphasis on pure values found in group activities (step by step and size accuracy for baking cakes). 2. Teacher praises the group who are successful in arranging the steps in cake baking and selecting proper material.	(5mins)

Additionally, students are also asked to write a cake recipe using an algorithmic technique, a series of instructions contained in the scratch program. The algorithm used consists of five instructions, namely:

a) Sift flour
b) Mix sugar, butter and eggs and stir well in a mixer
c) Mix the baking powder and dried fruit into the wheat flour
d) Mix the wheat flour and dried fruits batter with sugar, butter and eggs batter
e) Bake the dough for 45 minutes to 1 hour.

RESEARCH FINDINGS

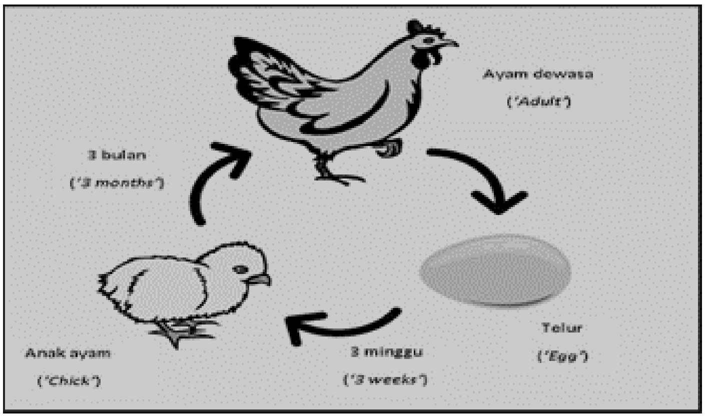
The findings of this study are based on the lesson implemented in a 4th year class based on four steps, namely the induction set; step 1; step 2; step 3; and enrichment activities.

Results and Induction set

The students are shown a picture of a life cycle of chickens as shown in Figure 4. Based on the picture of this life cycle, the students have constructed grammatical single sentences for example:

a) The chicken in the picture is a hen.
b) The eggs have hatched.
c) A chick hatches from a chicken egg

Figure 4: Chicken Life cycle

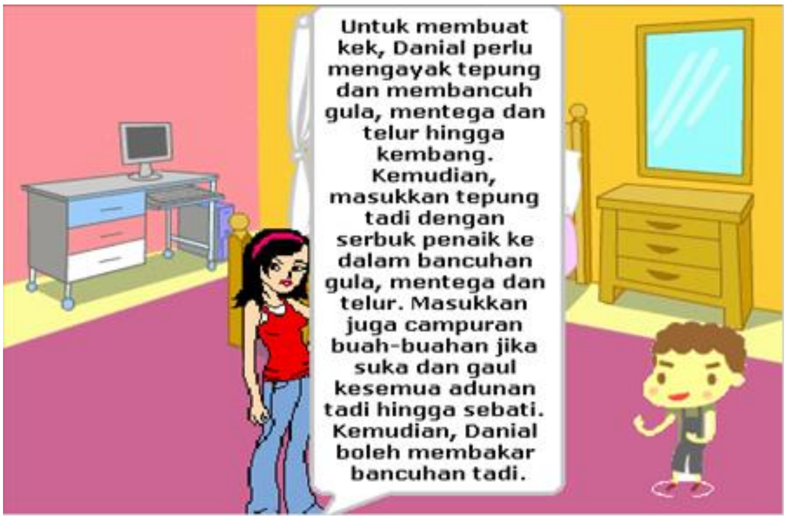


Source: Researcher

Findings of Step 1

Teacher showed a scratch program conversation as shown in Figure 5. Students were able to read conversations and identify agricultural-based cake-baking ingredients. Next, the question and answer session was conducted on the agricultural-based ingredients in the cake.

Figure 5: A scratch program conversation


Untuk membuat kek, Danial perlu mengayak tepung dan membancuh gula, mentega dan telur hingga kembang. Kemudian, masukkan tepung tadi dengan serbuk penaik ke dalam bancuhan gula, mentega dan telur. Masukkan juga campuran buah-buahan jika suka dan gaul kesemua adunan tadi hingga sebati. Kemudian, Danial boleh membakar bancuhan tadi.

Source: Researcher

Findings of Step 2

The scratch program was shown and students successfully listed ingredients based on agricultural products for example, butter; eggs; and granulated sugar, referring to a dialogue in the scratch program. After the dialogue in the scratch program was read, the students listened carefully to the steps of baking the cake as shown in Figure 6.

Figure 6: Steps in baking a cake

Source: Researcher

Findings of Step 3

Students formed four small groups as shown in Figure 7 and wrote steps to prepare cakes. Group representatives present their own group findings and four groups were able to write the correct steps in a cake preparation.

Figure 7: Small group discussion and presentation

Source: Researcher

Findings on Enrichment Activities

Students were asked to write steps to prepare the cake using a single sentence based on the scratch program. Students were able to write these steps using a grammatical single sentence as shown in Figure 8. The examples of single sentences formed are as follows:

a) First, the flour must be sifted.
b) Stir sugar, butter and egg well.
c) Mix flour and baking powder into the dough.
d) Mix dried fruit and stir well
e) Stir the dough well.

f) Bake the dough in an oven

Figure 8: Steps in enrichments activities

Source: Researcher

DISCUSSION

Teaching and learning with scratch programs are particularly attractive to students in primary schools. Evidently, scratch programs are fun and easy to understand by students. Consequently, students are more focused in their learning. The use of the scratch program has led the students to write an algorithm-oriented instruction on baking steps with grammatical single sentences. Students are also not bound by the use of dull textbooks during teaching and learning process. As a conclusion to this research, it can be pointed out that computational thinking is a vital skill for 21st century students. Although a lot of research is being conducted into teaching both CT in schools, lots of third level students will never have been exposed to these concepts. It is important that all students have good problem solving skills and CT can greatly benefit this. Many different methods have been proposed and it seems like a non-compulsory CT course for students is a particularly effective and useful method.

CONCLUSION

This research have shown that teaching CT or integrating CT concepts could improve student's analytical skills; provide a better understanding that programming is about solving the problem not just the code; improve attitudes and confidence towards programming; used as an early indicator and predictor of academic success; and correlate strongly with general academic success

However, CT and research into it are still in the early stages, therefore long-term effects as well as additional benefits still need to be researched. The above findings are encouraging and show that CT is a beneficial skill but more research is required before the extent of the impact of teaching CT can have on students is known.

ACKNOWLEDGEMENT

The authors would like to acknowledge the involvements of teachers and 32, 4th year students at a Seremban district primary school where this research have been carried out.

REFERENCES

A.K.. (2012). November. Computer science at school/CS teacher education: Koli working group report on CS at school. In Proceedings of the 12th Koli Calling International Conference on Computing Education Research (pp. 29-38). ACM.

Barr, D., Harrison, J., Conery, L. (2011). Computational Thinking: A Digital Age. Learning & Leading with Technology, March/April, 20-23.

C.A.R. Hoare and N. Wirth. (1972). An Axiomatic Definition of the Programming Language PASCAL.Acta Informatica 2, 335--355 (t973)

Csizmadia, A and et al. (2015). Computational Thinking – A Guide for Teacher. Computing At School. UK.

J. Cuny, J. Snyder, and J. Wing, "Demystifying computational thinking for non-computer scientists," Unpublished manuscript in progress, referenced in http://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf

Natasha Nesiba, Enrico Pontelli. 2015. DISSECT: Exploring the Relationship Between Computational Thinking and English Literature in K-12 Curricula. http://fie-conference.org/sites/fieconference.org/files/1570093959.pdf pp.1-8

N. Wirth. (1973), Algorithms and Data Structures. Englewood Cliffs, N.J. Prentice-Hall

Qualls, J.A. and Sherrell, L.B., 2010. Why computational thinking should be integrated into the curriculum. Journal of Computing Sciences in Colleges, 25(5), pp.66-71.

Schulte, C., Dagiene, V., Eckerdal, A., Hornung, M., Jevsikova, T., Peters, A.K., Sentance, S., Thota. (2012). N.: Computer science at school / CS teacher education Koli working-group report on CS at school. In: Proceedings of the 12th Koli Calling International Conference on Computing Education Research.

ABOUT THE AUTHORS

ASSOCIATE PROFESSOR DR FARIDAH BINTI NAZIR
International Islamic University Malaysia,
Pagoh Education Hub.
fidafn@iium.edu.my

DR ZANARIAH JANO
Technical University of Malaysia,
Malacca.
zanariah.jano@utem.edu.my

NORLIZA OMAR
Technical University of Malaysia,
Malacca.
norliza.omar@utem.edu.my