The Electronic Journal of Mathematics and Technology, Volume 17, Number 1, ISSN 1933-2823

Computation as a Big Idea in Mathematics

Weng Kin HO
wengkin.ho@nie.edu.sg
Mathematics and Mathematics Education Academic Group
National Institute of Education
Nanyang Technological University
Singapore

Abstract

Charles (2005) defines a Big Idea in Mathematics to be "a statement of an idea that is central to the learning of mathematics, one that links numerous mathematical understandings into a coherent whole" (p. 10). Therein, Charles listed a total of 21 Big Ideas – acknowledging the impossibility that all mathematicians and mathematics educators can agree on these – with the hope that they can be a starting point to initiate conversations. In this paper, we propose that 'Computation' can be added to this list of big ideas in mathematics and give compelling reasons to support our proposal. Additionally, we describe some implications of such an inclusion with particular emphasis on teaching and learning of mathematics at schools.

1 Introduction

Mathematics educators in the preceding century recognized that mathematics teachers require a broad and deep knowledge base for teaching mathematics that includes knowing the subject content, pedagogy, students and student learning ([23, 25]). Content knowledge of mathematics in itself includes the nature of mathematics and the mental organization of teacher knowledge ([15]). Similarly, [3] reported on the influences that the nature, depth and organization of teacher knowledge have on teachers' presentation of ideas, capacity to facilitate students' connection of mathematical ideas and flexibility in responding to students' queries.

A teacher's integration of pedagogy and understanding of content forms the foundation for one's Pedagogical Content Knowledge defined by [25] as

> "the ways of representing and formulating the subject that make it comprehensible to others ... an understanding of what makes the learning of specific topics easy or difficult; the conceptions or pre-conceptions that student of different ages and backgrounds bring with them to the learning of those most frequently taught topics and lessons."

Thus, teachers' perception of mathematics needs to be a coherent and connected whole ([4, 26]), and [11] proposed that this be achieved through "the grounding of a teacher's mathematics content knowledge and their teaching practices around a set of Big Mathematical Ideas (*Big Ideas*)" (p. 9). In that same paper, a *Big Idea* in Mathematics is defined to be "a statement of an idea that is central to the learning of mathematics, one links numerous mathematical understandings into a coherent whole" (p. 10).

The phrase 'a coherent whole' stresses the building of strong connections as Hierbet and Carpenter pointed out that "the degree of understanding is determined by the number and strength of connections" ([17, p.67]). By viewing Big Ideas in Mathematics as having connections with other ideas and the understanding of which nurture a deep understanding of mathematics, teachers and students alike no longer perceive mathematics "as a set of disconnected concepts, skills, and facts" ([11]), but rather as a connected whole.

A total of 21 Big Ideas in Mathematics are listed in [11, pp. 12-21] which are intended for elementary and middle school mathematics. Understandably, formulating such a list is not highly non-trivial; the interested reader is encouraged to find out the evolutionary history of these 21 Big Ideas in Mathematics (p. 11).

Charles ([11]) asserts "that it is impossible to get one set of Big Ideas and Understandings that *all* mathematicians and mathematics educators can agree on" for his main purpose for listing down these 21 Big Ideas in Mathematics was then to initiate a conversation so that it can become mainstream conversation "about mathematics standards, curriculum, teaching, learning, and assessment" (p. 9).

Recent years saw increasing research efforts to understand Computational Thinking, and possible integration of Computational Thinking and Mathematics ([10]). Fundamental to this is the notion of 'Computation'. In this paper, we propose 'Computation' to be added to this existing list and give compelling reasons to support our proposal. We elaborate the (4 W's): *What*, *When*, *Where*, *Why*; and 1 H: *How* of the Big Idea of Computation. Additionally, some implications of such an inclusion on teaching and learning of mathematics at schools will be discussed. For the teacher audience, many examples are given to illustrate the practical aspects of our discussion.

2 Computation as a Big Idea in Mathematics

To argue that Computation is a Big Idea in Mathematics, two important qualities of a Big Idea need to be revisited. First, the name of the Big Idea does not constitute the idea itself. Charles ([11]) explains that "rather the Big Ideas are the statements that follow the name". In order to come to grips with the mathematical meaning of Computation as a Big Idea, we must give meaning and understanding to the word 'Computation'.

Secondly, a Big Idea must be one "central to the learning of mathematics" ([11, p.10]). For example, a mathematics student often sees regularity, repetition, structure and predictability through objects like number patterns, number sequences, shapes, algebraic rules, etc., where understanding is grounded on knowing that all the above observations are instances of *Patterns* (Big Idea #9) – where "relationships can be described and generalizations made for mathematical situations that have numbers or objects that repeat in predictable ways" [11, p. 17]. Hence our proposal for 'Computation' to be a Big Idea in Mathematics must be supported by many examples that computation is central to the learning of mathematics – a claim that the reader, we hope, would find intuitive even before reading this paper.

In this section, we first present the statement of the idea for the Big Idea of Computation and then elaborate on the central aspects of Computation as an idea pertaining to learning mathematics. This is done by describing the 4 W's and 1 H of Computation as a Big Idea in relation to mathematical learning.

- What: Computations can occur in various forms as the student engages in learning mathematics. Both teachers and students need to heighten their awareness if they are able to recognize something they are doing is actually a form of computation.
- When and Where: Firstly, learners of mathematics not only need to recognize a computation has been executed when it happens but also know when to execute one. Similarly, teachers need to know when a computation will activate effective learning of mathematical concepts. Secondly, computations do not occur arbitrarily anywhere. All computations are executed at some precise points somewhere in the process of mathematical learning. Teachers need to know where to place an opportune computation to ensure just-in-time mathematics learning.
- Why: Students often 'do' mathematics by executing calculations but often without a deep understanding of why these calculations were performed in the first place. Teachers must also know the reasons for asking students to perform a certain computation instead of calculating for its own sake.
- How: There are many different computations that achieve the same desired outcome. How do learners choose the 'right' computational method amongst several others? How can teachers exploit the affordance of computational methods in engaging students with higher-order thinking?

Charles [11] already warned that if anyone decides to modify the set of Big Ideas in Mathematics or build their own then one must be cognizant of the process he used to develop the list and some of the issues he confronted in that development. One thorny issue is about how big (or small) a Big Idea should be. Charles advised that "Big Ideas need to be big enough" that it is relatively easy to articulate several related ideas – what he called mathematical understandings ([11]). Thus, to justify that 'Computation' is a Big Idea in Mathematics we shall explicate those mathematical understandings related to 'Computation'.

The second important point is that Big Ideas need to be useful to teachers, curriculum developers, test developers, and policy-makers who are responsible for shaping the national or state standards. By describing the 4 W's and 1 H of 'Computation', we want to ensure that this Big Idea of Computation does not get too big so that its usefulness is unclear to the audience.

The last advice given in [11] regarding the inclusion of an idea as Big Idea in Mathematics is the size of the resulting list of Big Ideas – it must not be too many. Therefore, we argue that a few existing Big Ideas may be subsumed under it, thereby ensuring that a teacher's content knowledge, teaching practices and understanding of the curriculum may rest on a smaller collection of Big Ideas. We shall continually bear in mind Charles' advice as we continue our discussion.

2.1 Defining the Big Idea of Computation

The Big Idea of Computation can be defined as follows:

Definition 1 (Computation) *A* computation *is a calculation (arithmetic or not) performed by an agent (human or machine) following a well-defined model, i.e., one which describes how an output of a mathematical function is produced given an input. Such a calculation is finite and deterministic in nature, i.e., given an input, it produces a unique output by applying a finite set of rules and in finite time.*

Already in [11, p.16], some algorithms and computations for operations with rational numbers are mentioned under the Big Idea #7, Basic Facts and Algorithms. The scope of these algorithms is too narrow. In fact the mathematical understanding that comes with the Big Idea of Computation is far wider and may thus subsume Big Idea #7. Let us look at a few examples and non-examples of computation below to articulate the mathematical understanding of Computation as a Big Idea in Mathematics.

Example 2 (Computation: algebraic expansion) *Expand* $(2m+n)(m+n)$. *Correct expansion requires the distributive law (DL):*

$$(2m+n)(m+n) = (2m)\cdot m + (2m)\cdot n + n\cdot m + n\cdot n.$$

By the associative law (AL) of multiplication, $(2m)\cdot m = 2m^2$, $(2m)\cdot(-n) = 2mn$*; and by the commutative law (CL) of multiplication,* $n\cdot m = m\cdot n = mn$*, and so the above is expressed as:*

$$(2m)\cdot m + (2m)\cdot n + n\cdot m + n\cdot n = 2m^2 + 2mn + mn + n^2 = 2m^2 + 3mn + n^2,$$

where the last equality is obtained by grouping of like-terms (GL). In the process of expansion, one executes instances of some laws of arithmetic (DL, AL, CL and GL). The calculation starts with an algebraic expression, ends with another by applying a finite number of rules and terminates in finite time. A certain writing convention is adopted:

C1. The multiplication symbol $\times$ *is omitted between numbers and letters.*

C2. Numbers are written first, followed by letters in alphabetical order.

C3. Brackets are used to show multiplication of an expression that has more than one term.

C4. Powers are used to show (repeated) multiplication of a term with itself.

The arithmetic laws, together with the writing convention, constitute the 'algorithm' and so algebraic expansion may be regarded as a (arithmetic) computation.

Example 3 (Computation: rewriting systems) *Consider a rewriting system encountered in logic defined by the following finite set of rewriting rules:*

1. *(Double negation)* $\neg\neg A \to A$;
2. *(De Morgan's laws)* $\neg(A\wedge B) \to \neg A \vee \neg B$; *and* $\neg(A\vee B) \to \neg A \wedge \neg B$;
3. *(Distributive laws)* $(A\wedge B)\vee C \to (A\vee C)\wedge(B\vee C)$; *and* $A\vee(B\wedge C) \to (A\vee B)\wedge(A\vee C)$,

where $\rightarrow$ indicates that an expression matching the left hand side of the rule can be rewritten to one on the right hand side, and the symbols each denote a sub-expression. The rewriting terminates when none of the rules can be applied to any sub-expressions. For instance, the formula $\neg(p \vee \neg(\neg q \wedge r))$ may be rewritten in several ways to its terminating form: e.g.

$$\begin{aligned}
&\neg(p \vee \neg(\neg q \wedge r)) \\
\rightarrow&\neg(p \vee (\neg\neg q \vee \neg r)) \\
\rightarrow&\neg(p \vee (q \vee \neg r)) \\
\rightarrow&\neg p \wedge \neg(q \vee \neg r) \\
\rightarrow&\neg p \wedge (\neg q \wedge \neg\neg r) \\
\rightarrow&\neg p \wedge (\neg q \wedge r),
\end{aligned}$$

or

$$\begin{aligned}
&\neg(p \vee \neg(\neg q \wedge r)) \\
\rightarrow&\neg p \wedge \neg\neg(\neg q \wedge r)) \\
\rightarrow&\neg p \wedge (\neg q \wedge r).
\end{aligned}$$

Although there is no unique path of rewriting as different sub-expressions may be chosen to be rewritten first, the terminating expression, called the Conjunctive Normal Form (CNF)[1]*, is unique. The procedure relies on a finite set of rules, terminates at a unique output for each given input, and ends in finite time; whence an example of a (non-arithmetic) computation.*

Example 4 (Computation: recursive equations) *Recursion is common computation in mathematics. For example, the factorial function $f : \mathbb{N} \longrightarrow \mathbb{N}$ may be defined recursively by $f(0) = 1,\ f(n) = n \cdot f(n-1)$. Number sequences may also be coded recursively, e.g., the Fibonacci sequence $\{f_n\}_{n=1}^{\infty}$: $f_0 = f_1 = 1,\ f_{n+2} = f_n + f_{n+1}$. Numerical solutions of equations often involve recurrence equations; e.g. the Newton's Method for solving an equation $y = f(x)$ for a differentiable function f is given by*

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)},$$

where x_0 is a seed.

Example 5 (Computation: computer simulations) *Computers can be used to simulate random events via pseudo-random number generators. Actually, pseudo-random number generators are deterministic real-time automata, i.e., they take the current clock time as an input in the calculation. Let us use* MATLAB *to illustrate our point here. In* MATLAB*, the function* `rand()` *returns a pseudo-random value drawn from a uniform distribution in the interval [0,1]. The sequence of numbers produced by the pseudo-random number generator is determined by the internal state of the generator. Setting the generator to the same fixed state allows computations to be repeated, and so everything is fixed. In other words, each time* MATLAB *is started the state is reset and therefore the same sequence of*

[1] Any logical formula and its conjunctive normal form are logically equivalent, and so the Big Idea of Computation connects with the Big Idea of Equivalence (Big Idea #3) through rewriting systems.

numbers in each session will be generated. This drives home the point that pseudo-random number generators are deterministic mathematical functions. To simulate randomness, the function needs to be initialized to a different state each time the program starts, and hence typically this is done via the statement

```
rand('state',sum(100*clock))
```

Thus, a computer simulation of a random event is rightly considered as a computation because it is deterministic and finite in nature, i.e., computer simulations run on a finite set of rules and terminate in finite time.

Example 6 (Non-computation: non-termination) *Consider the following flowchart as an attempt to calculate the value of* $\sqrt{2}$.

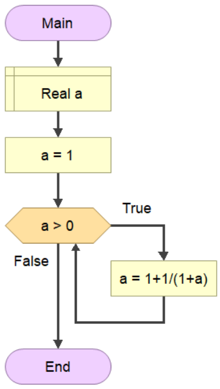

Figure 1: A flowchart that does not terminate.

Note that every step of passing information around in the flowchart involves only a finite set of operations. However, the flowchart's attempt to calculate $\sqrt{2}$ *is deemed to fail because the while-loop involves a stopping criterion of* $a \leq 0$ *(i.e., the negation of* $a > 0$ *as presented in the orange hexagon). Since the recursive assignment* $\left(a \mapsto 1 + \frac{1}{1+a}\right)$ *always yields a positive output, the flowchart enters into an infinite while-loop and hence never terminates. In Computation Theory, this would have been classified as a computation albeit a non-terminating one. Such a non-terminating computation yields no information and for us, this is useless. Therefore, we rule them out from our definition of 'Computation'. Of course, we can always change the looping criterion so that an estimate of* $\sqrt{2}$ *can be produced in finite time. Then such an altered flowchart will be regarded as a computation.*

As a passing remark, flowcharts often provide a convenient passage for students to start with algorithmic reasoning and later to develop computational thinking for problem solving. The advantage of using flowchart is that it is visual in nature and requires no programming pre-requisite ([19]). Computation as a Big Idea can enter a young child's mathematics education via early exposure to flowcharts.

Example 7 (Non-computation: natural random event or experiment) *Consider this experiment: roll a fair 6-sided die four times and record the result as a quadruple* (x_1, x_2, x_3, x_4)*, where* $x_i \in \{1, 2, 3, 4, 5, 6\}$ *for each* $i = 1, 2, 3, 4$*. Although the process terminates in finite time, one is not* *guaranteed that every run of the same process produces the same output. In Computation Theory, non-determinism is usually considered as a computational effect. Since our definition of 'Computation' is restricted to be deterministic by the requirement of a mathematical function in any well-defined model, this experiment is not regarded as a computation in our discourse.*

Remark 8 *Our interpretation of 'Computation' does not exclude the use of pseudo-random processes that computers exploit in simulating random events. What we do rule out is the use of natural physical systems that involve non-determinism or experimental uncertainty (e.g., flipping of coins, rolling of die, swinging a pendulum). The reason for that is that we do not subscribe to* pancomputationalism*, where it is postulated that evolution of the universe (together with all of its physical processes) is itself a computation.*

2.2 Computation is central to mathematical learning

Computation as a Big Idea in Mathematics must play a central role of supporting effective, engaging and meaningful learning experience in the mathematics classroom. From the preceding section, we have seen some of the 'what's of Computation which are mostly relevant to pre-tertiary and tertiary mathematics. Here we see how 'Computation' fulfills the role of a Big Idea in Mathematics at schools. We shall draw anecdotal examples from across various topics in primary and secondary school mathematics (in the context of Singapore) to illustrate the 'what', 'when', 'where', 'why' (4 W's) and 'how' (1 H) of Computation as a Big Idea.

What. Because school mathematics involves a lot of routinized calculations that are taught as methods that to be learnt by heart and executed at will, a large part of the classroom time has been dedicated to drill-and-practice. Every topic would have specific algorithms suited for certain purposes. Just to illustrate how early students in Singapore encounter algorithms, we point the reader to Table 1, where we have a partial list of algorithmic computations in the Number Strand (Sub-strands: Whole Numbers and Fractions) that primary school students in Singapore are required to perform at will. From this, it could be imagined the huge mass of computational algorithms that mathematics students would acquire throughout their first ten years of compulsory education.

It is certainly not our purpose to make a systematic list of all the computational skills that a student learns by the time one arrives at the secondary school. What we are concerned is whether students are aware which of the many activities they engage, while learning mathematics, are actually computational. Are all such computations of the same nature? The example below serves to explain the importance of 'what' in computation.

Example 9 (Solving quadratic equations) *Consider solving $x^2 - 5x + 6 = 0$ in the Secondary 3 (students aged 15) Singapore Mathematics syllabus. One way is to factorise the (monic) quadratic expression $x^2 - 5x + 6$ into the product of two linear factors by* trial and error. *To do so, the student first assumes that these factors involve only integers, i.e., the roots of the equation are rational. Next, one considers all possible integral factorizations of* 6*:*

$$6 = 1 \times 6 = 2 \times 3 = (-1) \times (-6) = (-2) \times (-3).$$

Primary level (7–12 years old)		
Sub-strand	Topic	Description
Whole numbers	Addition and Subtraction Multiplication and Division	addition and subtraction algorithms (up to 4 digits) multiplication algorithm - up to 4 digits by 1 digit - up to 3 digits by 2 digits division algorithm - up to 4 digits by 1 digit
Fractions	Mixed numbers and improper fractions Addition and subtraction	achieve mastery of conversion between mixed numbers and improper fractions adding and subtracting fractions with denominators of given fractions not exceeding 12 and not more than two different denominators

Table 1: A partial list of computations for primary school students in Singapore

Each of these possibilities corresponds to its expansion given by:

$$\begin{aligned}
(x+1)(x+6) &= x^2+7x+6 \textit{ (wrong!)}\\
(x+2)(x+3) &= x^2+5x+6 \textit{ (wrong!)}\\
(x-1)(x-6) &= x^2-7x+6 \textit{ (wrong!)}\\
(x-2)(x-3) &= x^2-5x+6 \textit{ (correct!)}
\end{aligned}$$

If rationality of the roots is not assumed, then this trial-and-error method is non-deterministic because there are infinitely many ways to factorize 6 *over* $\mathbb{R}$*.*

With an overt assumption of rational roots in place, the perceptive student would realize the difficulty of decomposing the constant term into two integral factors, especially when it is large. By our definition of 'Computation', although the trial-and-error method involves simple arithmetic calculations it is not a computation owing to its non-determinism and possible non-termination. Computation as a Big Idea provides us with an alternative paradigm in understanding why the trial-and-error method of factorizing quadratic expressions can sometimes be so hard and frustrating for students (see also [24]).

Another method for solving quadratic equations $ax^2+bx+c=0$ $(a \neq 0)$ *is available to Secondary 3 students in Singapore, namely, by using the quadratic formula:*

$$x = \frac{-b \pm \sqrt{b^2-4ac}}{2a}.$$

This method allows the student to obtain the roots quickly as no guessing is required – arithmetic calculations performed are deterministic and finite in nature. The algorithm ends in just a few steps with no assumption on the rationality of roots. E.g. $x^2-5x+6=0$ *can be solved as follows:*

$$x = \frac{-(-5) \pm \sqrt{(-5)^2-4(1)(6)}}{2(1)} = \frac{5 \pm \sqrt{25-24}}{2} = \frac{5 \pm 1}{2} = \frac{5-1}{2} \textit{ or } \frac{5+1}{2} = 2 \textit{ or } 3.$$

As students learn to identify (1) that certain activities are actually computations (and some which are not), and (2) amongst identified computations that achieve the same result those which are more efficient, they begin to make intelligent choices of the computational methods. For teachers, the Big Idea of Computation helps them shape their lesson development that will eventually guide students in making effective computational choices ([20]).

When. Knowing that something is a computation does not mean one knows when to perform it. To know when to execute a computation requires one's judgment of timeliness. Related to 'when' is also the matter of 'where', i.e., the situation where a computation should be performed.

Example 10 *Consider the following problem that can be given to a primary school pupil.*

> *Find three different whole numbers* $a,\ b$ *and* c *that will make* $\frac{1}{a}+\frac{1}{b}+\frac{1}{c}=1$.

When presented with this non-routine problem, one good heuristic would be to start experimenting with some small numbers. This important first step of applying computation on a few 'test' inputs will give the student a quick sense of what the problem is about. Crucially, preliminary computations aids sense-making and guides one to the solution.

Two obvious deductions can be made right away! Firstly, none of these numbers a, b *or* c *can take the value of* 1*; otherwise, the reciprocal of one of these is already* 1 *and thus* $\frac{1}{a}+\frac{1}{b}+\frac{1}{c}>1$ *because the other two reciprocals are positive. Secondly, if* $a,\ b,\ c>3$ *then* $\frac{1}{a},\ \frac{1}{b},\ \frac{1}{c}<\frac{1}{3}$*, and so* $\frac{1}{a}+\frac{1}{b}+\frac{1}{c}<\frac{1}{3}+\frac{1}{3}+\frac{1}{3}=1$*. This is why at least one of the* a, b *or* c *must be* 2*. So one might as well let* $a=2$*, and continue the search for* b *and* c*. Thus, we have* $\frac{1}{2}+\frac{1}{b}+\frac{1}{c}=1$ *which boils down to* $\frac{1}{b}+\frac{1}{c}=\frac{1}{2}$*. Again for the same reasons, plugging numbers* b *and* c *that are greater than* 4 *would never work. This forces us to guess that one of these, say* b*, must either be* 3 *or* 4*. Do note that if the corresponding value of* c *does not satisfy the criteria stated above (i.e.,* c *is a whole number distinct from* b *and* a*) then no solutions can exist, which will render the problem unsolvable!*

Writing c *as the subject of the formula (which is a timely and important computational technique)* $c=\frac{1}{\frac{1}{2}-\frac{1}{b}}$ *helps us test the two cases:*

$$(1)\ b=3 \implies c=\frac{1}{\frac{1}{2}-\frac{1}{3}}=6 \text{ or } (2)\ b=4 \implies c=\frac{1}{\frac{1}{2}-\frac{1}{4}}=4.$$

The search for the solution ends at the first option, i.e., $a=2, b=3$ *and* $c=6$*, noting that the second is inadmissible since* $b\neq c$*.*

This example may seem contrived; indeed one need not even try too long before hitting on the solution. However, the point here is that by combining computation and mathematical reasoning in a clever way one can determine if the equation has any desired solutions in a *finite* number of steps within *finite* time. To fully grasp the non-trivial reasoning that has been used in this example, the reader is encouraged to solve an extension of this problem, i.e., to affirm or refute the following statement:

You can never find four different whole numbers $a,\ b,\ c$ and d that satisfy the equation:

$$\frac{1}{a}+\frac{1}{b}+\frac{1}{c}+\frac{1}{d}=1.$$

In summary, computation often comes in handy when a student engages in problem solving, especially in the initial stages of trying to understand the problem, and to devise a plan ([28]).

Why. An overemphasis on drill-and-practice often lead students to believe that mathematics is all but repetitive, routinized and boring application of formulae and methods ([6]). Calculations when performed without meaning are then easily forgotten once the students graduated from their schools, leaving behind them most of the mathematics they learnt for good ([16]).

In the next example, we look at some rewriting systems encountered in Additional Mathematics, a mathematics course taken by most secondary school students in Singapore.

Example 11 *When students learn about trigonometric identities, they are usually introduced to some of these:*

T1. Reciprocal/quotient identities:

- $\csc\theta = \frac{1}{\sin\theta}$
- $\sec\theta = \frac{1}{\cos\theta}$
- $\cot\theta = \frac{1}{\tan\theta}$
- $\tan\theta = \frac{\sin\theta}{\cos\theta}$

T2. Pythagorean identities:

- $\sin^2\theta + \cos^2\theta = 1$
- $1+\tan^2\theta = \sec^2\theta$
- $1+\cot^2\theta = \csc^2\theta$

T3. Double angle formulae:

- $\sin 2\theta = 2\sin\theta\cos\theta$
- $\cos 2\theta \begin{cases} = 2\cos^2\theta - 1 \\ = 1-2\sin^2\theta \\ = \cos^2\theta - \sin^2\theta \end{cases}$
- $\tan 2\theta = \dfrac{2\tan\theta}{1-\tan^2\theta}$

Exercises that require students to prove trigonometric identities using some of the aforementioned identities are commonplace in students' homework, quizzes, tests and written examinations. For example, students are expected to know how prove the following identity:

$$1-\cos 2x = \tan x \sin 2x$$

using the following sequence of equations:

$$\begin{aligned} RHS &= \tan x \sin 2x \\ &= \frac{\sin x}{\cos x} \cdot 2 \sin x \cos x \\ &= 2\sin^2 x \\ &= 1 - \cos 2x \\ &= LHS. \end{aligned}$$

It is not difficult to see that the basic set of trigonometric identities which students need to know by heart represents a core set of rewriting rules for trigonometric expressions. Viewed in this way, proving of trigonometric identities is merely executing a finite sequence of rewriting rules for sub-expressions of the original trigonometric expression(s).

Another topic in Additional Mathematics that makes use of the idea of a rewriting system is that of obtaining the indefinite integral for a given function, i.e., its antiderivative. The following are some indefinite integrals students need to memorize: Let a and b be constants, $a \neq 0$.

I1. $\int \sin(ax+b)\,\mathrm{d}x = -\frac{\cos(ax+b)}{a} + C$

I2. $\int \cos(ax+b)\,\mathrm{d}x = \frac{\sin(ax+b)}{a} + C$

I3. $\int \sec^2(ax+b)\,\mathrm{d}x = \frac{\tan(ax+b)}{a} + C$*, where C is an arbitrary constant.*

Students understand very little of why trigonometric identities or indefinite integrals are important, and even less so of the connection between these topics. Computation as a Big Idea in Mathematics helps to connect these two seemingly distant topics. The idea here is to join these two rewriting systems from a computational point of view.

Consider the following problem of finding an antiderivative of a trigonometric function:

> *Find*
> $$\int \cos^4(x) - \sin^4(x)\,\mathrm{d}x.$$

In order to find an antiderivative of $\cos^4(x) - \sin^4(x)$, one needs to execute a pattern matching with the possible forms of the integrand in the list (I1)-(I3). Only those forms of the function that appears in (I1)-(I3) are admissible for rewriting to take place, and so clearly neither $\cos^4(x) - \sin^4(x)$ nor its sub-expressions are not in this list. To facilitate rewriting using the rules in (I1)-(I3), one must be able to rewrite $\cos^4(x) - \sin^4(x)$ into a function that appears in this list.

In order to achieve this, the student must be aware that there exists another rewriting rule elsewhere that would do the job. Difference of two squares leads to this factorization:

$$\cos^4(x) - \sin^4(x) = \left(\cos^2(x) + \sin^2(x)\right)\left(\cos^2(x) - \sin^2(x)\right).$$

Then the salient rules to be used are the Pythagorean identity (T1) and the Double Angle Formula (T3). In particular, the student needs to choose the first identity regarding $\cos^2(x) + \sin^2(x)$ and the *third identity regarding* $\cos 2\theta$*, namely,* $\cos^2\theta + \sin^2\theta = 1$ *and* $\cos 2\theta = cos^2\theta - \sin^2\theta$*. Replacing* θ *with* x*, one has*

$$\cos^2(x) + \sin^2(x) = 1 \text{ and } \cos^2(x) - \sin^2(x) = \cos(2x).$$

This would result in rewriting the indefinite integral as:

$$\int \cos^4(x) - \sin^4(x)\,\mathrm{d}x = \int 1\cdot\cos(2x)\,\mathrm{d}x = \int \cos(2x)\,\mathrm{d}x.$$

Since $\cos(2x)$ *matches the pattern in (I2), the student executes the rewriting rule (I2):*

$$\int \cos^2(x) - \sin^2(x)\,\mathrm{d}x = \int \cos(2x)\,\mathrm{d}x = \frac{\sin(2x)}{2} + C.$$

Viewing the processes of proving trigonometric identities and finding antiderivatives are related ones through the computational lens of rewriting systems, it is easy to anticipate that most students have difficulty tackling this kind of problem. The teacher who is aware of such learning difficulties may then draw to the students' attention the presence of two rewriting systems that they can apply at their disposal. By linking these two seemingly unrelated topics through the Big Idea of Computation, students begin to appreciate mathematics that they are learning as a coherent body of knowledge and thus attain a deeper level of understanding.

How. This last aspect of 'how' of Computation deals with two questions: (1) How to compute something? (2) Amongst the several 'how's, which one should be chosen to best achieve the desired computation? The first question is too general for us to give a sensible answer. Firstly whether a given problem can be solved computationally is a very general (and hard) theoretical problem, which we may have neither the room nor expertise for development here. After all, it is not within the usual learning experience of a mathematics student in an authentic classroom to create a new method of computation. It is the second question we want to pursue. Suppose a student knows more than one method to carry out a certain computation. Does he know which to choose? Though this concern has been raised much earlier, e.g., [27], very few classroom implementations are designed to teach students how to make wise choice of computational options. In our earlier discussion of the 'what' of Computation, we have emphasized that the student must grow in one's awareness of computation and its nature. The following example arises in the choice of computational method for evaluating the combination number – also known as the binomial coefficient.

Example 12 *Students encounter the combination number* $\binom{n}{r}$ *in combinatorics as well as the binomial theorem. This number is defined analytically by*

$$\binom{n}{r} = \frac{n!}{r!(n-r)!}, \qquad (0 \le r \le n),$$

where the symbol $m!$ *denotes the factorial of* m*, defined by* $m \times (m-1) \times \cdots \times 1$*. In combinatorics, one can reason that the combination number* $\binom{n}{r}$ *equals the number of ways one can select* r *objects out of* n *distinct objects.*

From the definition, one immediately arrives at the fact that

$$\binom{n}{r} = \frac{n \times (n-1) \times \cdots \times (n-r+1)}{r \times (r-1) \times \cdots \times 1}. \tag{1}$$

For instance, one can compute $\binom{6}{4}$ *as follows:*

$$\binom{6}{4} = \frac{6 \times 5 \times 4 \times 3}{4 \times 3 \times 2 \times 1} = \frac{6 \times 5}{2 \times 1} = 15, \text{ and } \binom{6}{2} = \frac{6 \times 5}{2 \times 1} = 15.$$

That $\binom{6}{4} = \binom{6}{2}$ *is just an instance of a more general identity concerning combination numbers. Replacing* r *by* $n - r$ *in the Equation (1) yields*

$$\begin{aligned}\binom{n}{n-r} &= \frac{n \times (n-1) \times \cdots \times (n-(n-r)+1)}{(n-r) \times (n-r-1) \times \cdots \times 1} \\ &= \frac{n \times (n-1) \times \cdots \times (r+1)}{(n-r)!} \\ &= \frac{n \times (n-1) \times \cdots \times (r+1) \times r!}{(n-r)!r!} \\ &= \frac{n!}{(n-r)!r!} \\ &= \binom{n}{r}.\end{aligned}$$

A simple combinatorial proof of this relies on a certain bijection principle, i.e., each selection of r *objects out of* n *distinct ones is in one-to-one correspondence to a selection of* $n - r$ *objects out of* n *distinct ones through set complement. So the combinatorial identity*

$$\binom{n}{r} = \binom{n}{n-r} \tag{2}$$

is taught to students earlier in their acquaintance with these combination numbers. This identity provides an appropriate context for the teacher to talk about computational choices. For instance, the teacher may ask the students to work out $\binom{13}{11}$ *mentally. Which of the following computational options should the student choose?*

$$(1)\ \binom{13}{11} = \frac{13 \times 12 \times 11 \times \cdots \times 3}{11 \times 10 \times \cdots \times 1} \text{ or } (2)\ \binom{13}{11} = \binom{13}{2} = \frac{13 \times 12}{2 \times 1}?$$

A student who observes that there are fewer factors to multiply in both the numerator and denominator in (2) would decide to use this as the more efficient computation of $\binom{13}{11}$*. An immediate benefit of this choice results in the efficiency of binomial expansion. E.g. in the expansion of* $(1+x)^5$*, instead of computing* all *the binomial coefficients in*

$$1 + \binom{5}{1}x + \binom{5}{2}x^2 + \binom{5}{3}x^3 + \binom{5}{4}x^4 + x^5,$$

it suffices to calculate only those coefficients $\binom{5}{r}$*, where* $1 \le r \le 2$*, whence*

$$1 + \binom{5}{1}x + \binom{5}{2}x^2 + \binom{5}{2}x^3 + \binom{5}{1}x^4 + x^5.$$

3 Implications to teaching and learning mathematics

From our preceding exposition, we have described Computation as a Big Idea in Mathematics, and used the 4 W's and 1 H to illustrate further the understanding of this Big Idea in context of mathematics education in schools. In this section, we elaborate how the recognition of Computation as a Big Idea may be translated to lesson tasks and activities that make use of it in engaging students in meaningful lesson tasks and activities. Instead of dealing with topical applications, we take a more objective-oriented approach, i.e., what are the lesson's aims?

3.1 Lesson consolidation, guided-discovery and formation of conjectures

Lesson consolidation. One important component of a lesson is consolidation, where the teacher helps the students summarize and overview what they have learnt using the lesson development ([5]). One of the ways is for students to make notes and chart mind-maps to gather all the related definitions, concepts and algorithms at one place. Because each topic involves computational procedures that undergird essential concepts, the Big Idea of Computation lends itself naturally to help students connect all these seemingly isolated nuggets of content knowledge into one coherent picture. Here is an example.

Example 13 *Electronic spreadsheets are very efficient platform for students and teachers to use with regards to connecting Computation with the Big Ideas in Mathematics ([18]). Having a lower technical overheads as compared to programming languages, spreadsheets have many useful built-in mathematical functions and features that students manufacture useful products as they construct their mathematical knowledge.*

For Additional Mathematics, Secondary 3 students would have acquired the following mathematical content knowledge regarding quadratic expressions and functions:

- *Factorization of quadratic expressions; Quadratic formula*
- *Discriminant and nature of roots; Sum and product of roots*
- *Graphing of quadratic functions; Quadratic inequalities and their solutions.*

To help students consolidate their learning in this topic, teachers may guide them to build their own quadratic calculator (see Figure 2).

A quadratic calculator allows the user to enter the coefficients a, b and c of a quadratic function $f(x) = ax^2 + bx + c$ (in the yellow-colored cells), as well as the interval $[A, B]$ which represents the domain of f. The calculator designed as a spreadsheet has several features. Firstly, it graphs the function $y = f(x)$ in the right panel. Secondly, it completes the square for the function and presents both the coordinates of the turning point – stating its nature – and the equation of the line of symmetry $x = -\frac{b}{2a}$. The calculator would determine the nature of the roots of the equation $f(x) = 0$, and calculate their values if they exist.

The important point here is that (i) the teacher shows the students the final product of the quadratic calculator and demonstrates some of its features, and then (ii) the students were given the task to manufacture this quadratic calculator. Students are taught just-in-time spreadsheet skills (how to use the fill-handle to invoke recursion, how to display graphs, etc.) to be able to construct this spreadsheet.

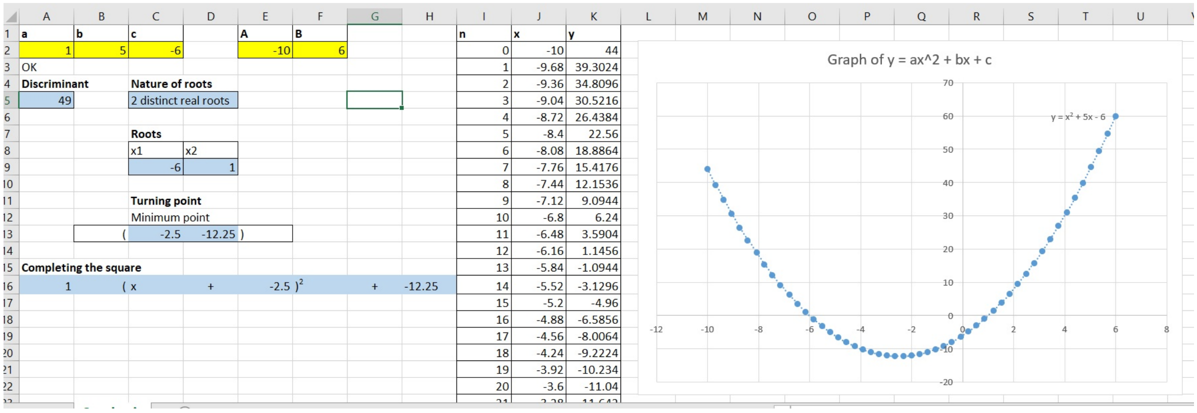

Figure 2: Student's product: quadratic calculator

In their construction, students employ all they have learnt about quadratic expressions and functions, as well as other topics like number sequences (for creating the table of values). Once the students have completed the construction of the quadratic calculator, they would have integrated computational thinking and mathematical thinking to produce a physical product – an concrete evidence of their understanding of the topic. The quadratic calculator can then be used by the students should they revisit and revise the topic again.

Guided-discovery. Teachers can make use of the Big Idea of Computation to create guided-discovery journey ([8]) for students. One can make use of pseudo-codes or codes of some programming language as a tool for guiding students to discover some mathematical results.

Example 14 *In the topic of Coordinate Geometry in Additional Mathematics, a Secondary 3 student will learn a new method called the* shoelace formula *to computing the area of a triangle with three non-collinear vertices having coordinates* $A(x_1, y_1)$, $B(x_2, y_2)$ *and* $C(x_3, y_3)$, *i.e.,* $\triangle ABC$ *has area:*

$$\frac{1}{2}\left|(x_1y_2 + x_2y_3 + x_3y_1) - (x_2y_1 + x_3y_2 + x_1y_3)\right|. \tag{3}$$

In this guided-discovery activity, the students are asked to study a set of pseudo-codes for calculating the area of a triangle ABC *with given coordinates (given below):*

```
FUNCTION AreaOfTriangle((x1,y1),(x2,y2),(x3,y3)):
    a1 = ((y1 + y2)*(x2 - x1))/2  #Area of a trapezium
    a2 = ((y2 + y3)*(x3 - x2))/2
    a3 = ((y1 + y3)*(x3 - x1))/2

    IF  a1 + a2 < a3 then
        area = a3 - (a1 + a2)
    ELSE
        area = (a1 + a2) - a3

    OUTPUT area
```

As they study the program, the teachers will carry out a conversation with the students that lead them to understand the meaning of the syntax of the program. Crucially, such a conversation is centred around two pictures (see Figure 14 below).

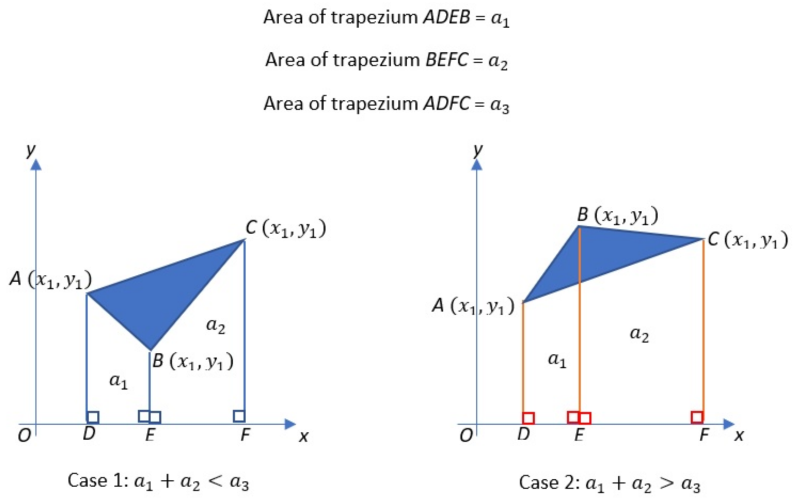

Figure 3: Deriving the formula for the area of the triangle ABC: two different cases

The teacher may guide the students to answer the following questions:

1. *What do you think the symbol # mean?*
   (Answer: It stands for a comment line, where the programmer wishes to write some remarks or reminders not intended to be compiled by the program.

2. *The comment line indicates that* `a1` *represents the area of a certain trapezium. Which one do you think it refers to in the left (or right) diagram in Figure 14? A similar question may be asked of the syntax* `a2` *and* `a3`*.*
   Answer: `a1` *(resp.,* `a2`*,* `a3`*) stands for the area of trapezium ADEB (resp., BEFC, ADFC).*

3. *By examining both the diagrams in Figure 14, explain the geometrical meaning of the inequality found in the conditional statement* `a1 + a2 < a3`*.*
   Answer: When the point B is below the line segment AC, the total area of trapezia ADEB and BEFC is less than the area of trapezium ADFC.

4. *In each of the cases, i.e., Case 1, when $a_1 + a_2 < a_3$, and Case 2, when $a_1 + a_2 \geq a_3$, calculate the area of the triangle ABC in terms of the areas* `a1`*,* `a2` *and* `a3`*.*
   Answer: Case 1. Area of ABC = $a_3 - (a_1 + a_2)$; Case 2. Area of ABC = $(a_1 + a_2) - a_3$..

5. *After the lesson, the students are given the homework of deriving the shoelace formula for finding the area of the triangle ABC, i.e., Equation (3). A possible solution for this homework is given below:*

$$
\begin{aligned}
&|a_3 - (a_1 + a_2)| \\
=&\left|\frac{(y_1+y_3)(x_3-x_1)}{2} - \frac{(y_1+y_2)(x_2-x_1)}{2} - \frac{(y_2+y_3)(x_3-x_2)}{2}\right| \\
=&\frac{1}{2}\left|y_1x_3 - y_1x_1 + y_3x_3 - y_3x_1 - y_1x_2 + y_1x_1 - y_2x_2 + y_2x_1 - y_2x_3 + y_2x_2 - y_3x_3 + y_3x_2\right| \\
=&\frac{1}{2}\left|(x_1y_2 + x_2y_3 + x_3y_1) - (x_2y_1 + x_3y_2 + x_1y_3)\right|.
\end{aligned}
$$

With this formula the students have derived, the teacher may then tasks them to create an alternative set of pseudo-codes for the same program specification, i.e., finding the area of the triangle given the coordinates of all its vertices.

Forming conjectures. Computation sets up a convenient environment for pattern recognition and hence a rich opportunity for students to learn to form conjectures ([9]). In the example below, students make use of their observation of computations as regular patterns, and in so doing form simple conjectures.

Example 15 *Students are shown three blocks of 4 numbers (each):*

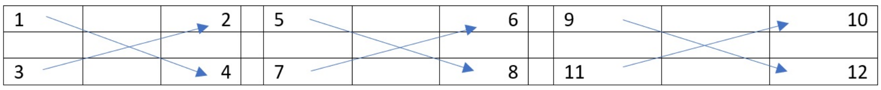

Figure 4: Forming conjectures using computations

The teacher then instructs the students to look at the first block of four numbers (Figure 4, left) multiply the bottom-left number by the top-right number, and likewise for the top-left and bottom-right numbers; and then to take the difference to obtain a positive number. For example, in the first set of four numbers on the left, we have $3 \times 2 - 1 \times 4 = 6 - 4 = 2$. *The teacher then asks the students to repeat the same computational procedure for the other two blocks of numbers, and to make a new conjecture capturing whatever pattern they have observed. One such conjecture they may form is* $bc - ad = 2$ *for any block of numbers (see Table 2):*

a		b
c		d

Table 2: A block of numbers in this sequence

The teacher then asks the students to give reasons to explain their observation or even to come up with a proof for this conjecture.

3.2 Problem solving and mathematical modelling

Problem solving. The heuristic of "Act It Out" can often be implemented through computation ([28]). So the Big Idea of Computation can easily set in whenever students engage in understanding an unseen problem, making sense of it and attempts to devise a plan.

Example 16 *The teacher invites the students to play a game, which involved a certain arrangement of boxes as shown in Figure 5. The teacher randomly distributes the given numbers* 1*,* 2*,* 3 *and* 4 *into the top layer of boxes: one number to each box.*

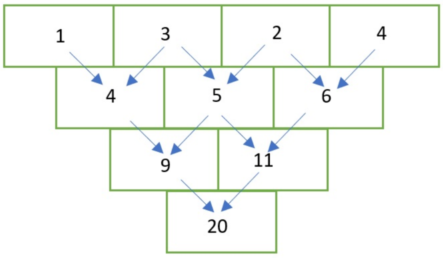

Figure 5: A number game

Two adjacent numbers will be added together to produce the number below them in the next layer. This computation will be carried out for all the cells in each layer, till the final number is computed at the bottom layer. For instance, in the above distribution $\{1,3,2,4\}$ *the final number obtained is* 20 *(Figure 5). The teacher then poses the problem to look for the distribution(s) of numbers in the top layer that will maximize the final number appearing in the bottom layer. Different variants and extensions of the problem may also be considered; for instance, increasing the number of cells in the top layer and the number of layers, allowing repetitions for some of the given numbers, and so on.*

Mathematical modelling. In any mathematical modelling, computations bring life to the model and allow the user to see how the model simulates and approximates the actual phenomenon modelled after ([2]).

Example 17 *Forest or bush fires are a common disaster experienced in many countries during dry seasons. A mathematical model inspired by cell-automata may be used to simulate the spread of forest fires; this approach originated from [21]. Figure 6 shows a* 10×10 *grid with each cell representing a tree in the forest. An initial number* 2 *was placed somewhere in the middle for Time Frame 1, representing the source of the first fire.*

Students are asked to (1) split the number 2 *into two non-negative integers, e.g.,* $2 = 0+2$ *or* $1+1$*, (2) select two adjacent cells (a cell is* adjacent *to another if they share a common edge or vertex) and (3) fill the cells with the numbers in any preferred way. This step is intended to model the phenomenon of fire spreading from the source to neighboring trees. Figure 7 shows some ways to choose the cells and fill in the numbers to model possible spread patterns.*

In Time Frame 2, the numbers in each existing cell is increased by 1 to model the increase in the intensity of fire. Then the students again split the numbers found in each cell and distribute to randomly selected adjacent cells. At instances where a cell is receiving more than one number from

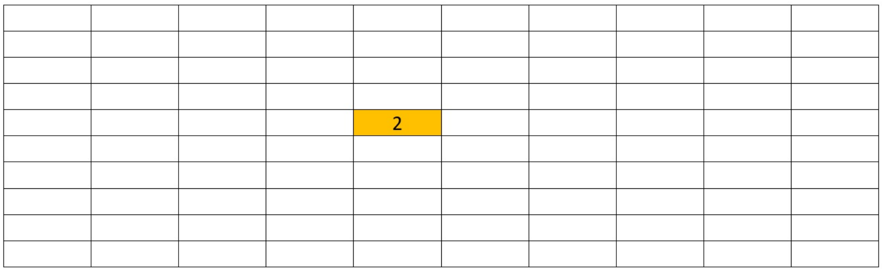

Figure 6: Initial state of the setup in Time Frame 1

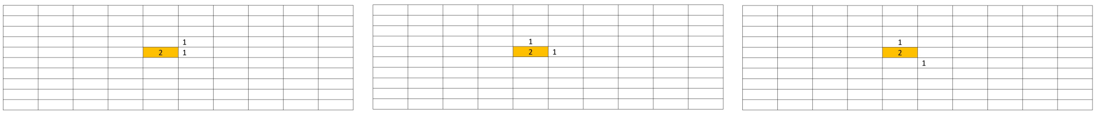

Figure 7: Possible patterns of fire spreading in Time Frame 1

its adjacent cells, the sum of the incoming numbers is obtained and added to the number currently occupying that cell. Figure 8 (left) shows such a spread pattern in Time Frame 2 which emerges from the spread pattern depicted in Figure 7 (left). Then, the students apply these computational rules in the subsequent time frames. Figure 8 (right) depicts a possible spread pattern in Time Frame 3.

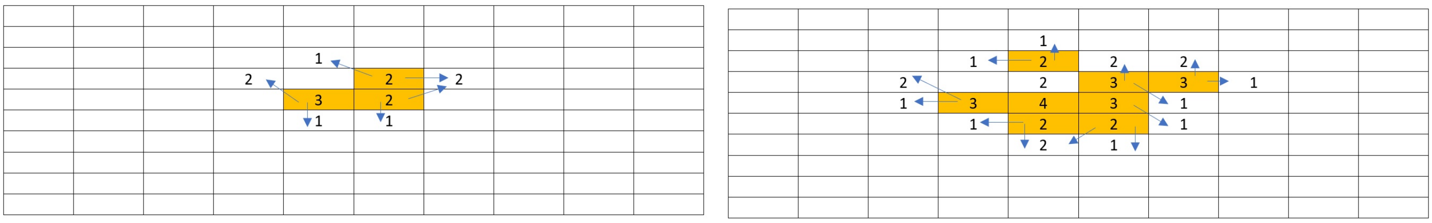

Figure 8: Possible spread pattern in Time Frames 2 and 3

Based this mathematical model, the teacher may lead the students to re-examine some of the assumptions made in this model; e.g. the spreading always involves two neighboring cells, the numbers in each cell keep increasing indefinitely as time progressed, etc. The students can also consider ways in which spreading may be slowed down or prohibited, and include these interventions into the existing model.

4 Conclusion

4.1 Summary

In this paper, we have proposed ‘Computation’ be included as one of the Big Ideas in Mathematics. By defining computation as a deterministic input-output process of finite nature, i.e., determined by a finite set of operational rules and terminating in finite time, we articulate the essence of this new Big Idea. This definition is clarified through several examples and non-examples, and is followed by a qualitative description of the Big Idea of Computation using a 4 W's and 1H approach – establishing the centrality of this Big Idea in the learning of mathematics at schools. We believe that Computation as a Big Idea in Mathematics can have significant implications to the teaching and learning of mathematics, and we defend this belief by providing an assortment of applications in several pedagogical scenarios, ranging from lesson consolidation, through mathematical problem solving, to mathematical modelling. The ubiquitous nature of Computation allows it to be used as a connecting idea between various topics.

4.2 Future works

Lesson study on classroom implementations that invoke the Big Idea of Computation. While we have demonstrated that Computation as a Big Idea in Mathematics has its potential applications in enhancing teaching and learning of school mathematics, there are currently very few case studies of such classroom implementations, let alone a critical evaluation of the efficacy of this approach. In particular, we have talked about the 4 W's and 1 H of Computation but left out 'Who'. The question of who really benefits from the using of the Big Idea of Computation in teaching and learning is indeed an open problem. Future works should be geared towards actual classroom implementation based on Computation as a Big Idea in Mathematics.

Non-computational aspects. Not everything is computable in mathematics; for instance, there are non-computable number sequences known as lawless sequences ([22]), and non-computable real numbers. Moreover, many of the existential proofs in mathematics are not constructive in nature – for example, Hilbert's Nullstellensatz – which can only be conceived from a *non-computational* point of view. Mathematics teachers should take the opportunity to let students appreciate the beauty of mathematics in its totality, i.e., both the computational and non-computational views. The relationship between these two views deserve further investigation.

Computations with computers. Though the main agent for implementing a computation in a mathematics classroom is still human, it is important to emphasize that the ultimate goal is to get the students to harness the speed and accuracy of computers. There are many new areas in which computer technology which are rapidly changing our approach of teaching and learning mathematics, e.g., the use of Artificial Intelligence ([29]), Augmented Reality ([1]), Automated Reasoning ([7]), Dynamic Geometry, and Computer Algebra Systems, and so on. Founded on Computation ass a Big Idea in Mathematics, the 21st century mathematics classroom must be ready to embrace and exploit computer technology.

Impact on future mathematics curriculum. This paper explores the role of Computation as a Big Idea in Mathematics in the mathematics classroom of today's world. Can Computation as a Big Idea in Mathematics be employed to develop a fundamentally novel mathematics curriculum? Recently some efforts have been invested in this direction by Conrad Wolfram ([13]). One 'low-hanging fruit' is to introduce numerical analysis or numerical methods in schools ([14]). Certainly, more intentional academic investigations need to be put into this important area of research.

References

[1] Ahmad, N., and Junaini, S. (2020). Augmented Reality for Learning Mathematics: A Systematic Literature Review. *International Journal of Emerging Technologies in Learning (iJET), 15*(16), 106-122. Kassel, Germany.

[2] Ang, K. C. (2020). Computational thinking as habits of mind for mathematical modelling. In W.-C. Yang, & D. Meade (Eds.), *Electronic Proceedings of the 25th Asian Technology Conference in Mathematics* (pp. 126-137). Mathematics and Technology, LLC.

[3] Ball, D. L. (1988). *Research on Teaching Mathematics: Making Subject Matter Knowledge Part of the Equation*. National Center for Research on Teacher Education, East Lansing, MI.

[4] Boaler, J., and Humphreys, C. (2005). *Connecting Mathematical Ideas: Middle school video cases to support teaching and learning.* Heinemann.

[5] Bobis, J., Russo, J., Downton, A., Feng, M., Livy, S., McCormick, M., and Sullivan, P. (2021). Instructional Moves that Increase Chances of Engaging All Students in Learning Mathematics. *Mathematics, 9*, (582), 1-19. Available at https://doi.org/10.3390/math9060582.

[6] Boesen, J., Helenius, O., Lithner, J., Bergqvist, E., Bergqvist, T., and Palm, T. (2014). Developing mathematical competence: From the intended to the enacted curriculum, *Journal of Mathematical Behavior, 33*, 72-87

[7] Botana, F., Recio, T., Pilar Vélez, M. (2017). The role of automated reasoning of geometry statements in mathematics instruction. *CERME 10*, Dublin.

[8] Bruner, J. (1961). *The act of discovery*. Harvard Educational Review.

[9] Canadas, M.C., Deulofeu, J., Figueiras, L., Reid, D., and Yevdokimov, O. (2007). The conjecturing process: perspectives in theory and implication in practice. *Journal of Teaching and Learning, 5*(1), 55-72.

[10] Chan, S.-W., Looi, C. K., Ho, W. K., and Kim, M. S. (2022). Tools and Approaches for Integrating Computational Thinking and Mathematics: A Scoping Review of Current Empirical Studies. *Journal of Educational Computing Research*, 1-45.

[11] Charles, R. I. (2005). Big ideas and understandings as foundations for elementary and middle school mathematics. *Journal of Mathematics Education Leadership, 7*(3), 9 – 14.

[12] Choy, B. H. (2019). Teaching towards Big Ideas: Challenges and Opportunities. In T. L. Toh and J. B. W. Yeo (Eds.), *Big Ideas In Mathematics, Yearbook 2019, Association of Mathematics Educators* (pp. 95 – 112). World Scientific.

[13] Conrad. W. (2020). *Math(s) Fix, The: An Education Blueprint for the AI Age.* Wolfram Media.

[14] Corless, R. (2004). *Computer-mediated Thinking*. Technical Report. ORCCA and Department of Applied Mathematics, University of Western Ontario, Canada. Available at https://publish.uwo.ca/~rcorless/papers/educ/CMTpaper.pdf.

[15] Fennema, E., and Franke, M. (1992). Teachers' knowledge and its impact in: D.A. Grouws (Ed) *Handbook of Research on Mathematics Teaching and Learning* (New York: Macmillan Publishing).

[16] Hiebert, J. (2003). What research says about the NCTM Standards. In J. Kilpatrick, G. Martin, & D. Schifter (Eds.), *A research companion to principles and standards for school mathematics*, National Council of Teachers of Mathematics, Reston, VA, (pp. 5-26).

[17] Hiebert, J., and Carpenter, T. P. (1992). Learning and teaching with understanding. In D. A. Grouws (Ed.), *Handbook of research on mathematics teaching and learning* (pp. 65-97). New York: MacMillan Publishing Company.

[18] Ho, W. K., Ghosh, J., Kaur, B., and Looi, C. K. (2022). Spreadsheets: A Technological Nexus Between Big Ideas in Mathematics and Computational Thinking in Mathematics Classrooms. *Submitted to the 27th Asian Technology Conference in Mathematics (currently under review)*, 1-15.

[19] Ho, W. K., Looi, C. K., Huang, W., Seow, P., Chan, S. W., and Wu, L. (2021). Flowgorithm: A possible passage from algorithmic reasoning to creatively founded mathematical reasoning. In Yang, W.-C., Meade, D., & Majewski, M. (Eds.), *Electronic Proceedings of the Twenty-Sixth Asian Technology Conference in Mathematics* (pp. 97-111). South Carolina, United States: Mathematics and Technology, LLC.

[20] Hurst, C. (2016). Towards a framework for making effective computational choices: A 'very big idea' of mathematics. *Australian Primary Mathematics Classroom, 21*(4), 16 - 21.

[21] Jellouli, O., Bernoussi, A., Mâatouk, M., and Amharref, M. (2016). Forest fire modelling using cellular automata: application to the watershed Oued Laou (Morocco), *Mathematical and Computer Modelling of Dynamical Systems, 22* (5), 493-507.

[22] Kreisel, G. (1969). Lawless sequences of natural numbers, *Compositio Mathematica*, tome 20, 222-248.

[23] National Council of Teachers of Mathematics (NCTM). (1991). *Professional standards for teaching mathematics.* Reston, VA: NCTM.

[24] O'Connor, B. R., and Norton, S. (2016). *Investigating Students' Mathematical Difficulties with Quadratic Equations.* Mathematics Education Research Group of Australasia (Adelaide: MERGA) pp. 552–559.

[25] Shulman, L. (1986). Those who understand: Knowledge growth in teaching. *Educational Researcher, 15*(2), 414.

[26] Sullivan, P. (2011). Identifying and describing the knowledge needed by teachers of mathematics. *Journal of Mathematics Teacher Education, 14*, 171 - 173.

[27] Swan, P. (2004). Computational choices made by children: What, why and how. *Australian Primary Mathematics Classroom, 9*(3), 27-30.

[28] Toh, T. L., Quek, K. S., Leong, Y. H., Jaguthsing D., and Tay, E. G. (2011). *Making Mathematics Practical: An Approach to Problem Solving*. World Scientific.

[29] Van Vaerenbergh, S., and Pérez-Suay, A. (2022). A classification of artificial intelligence systems for mathematics education. In Richard P. R., Vélez, M. P., Van Vaerenbergh, S. (Eds.), *Mathematics Education in the Age of Artificial Intelligence: How Artificial Intelligence can Serve Mathematical Human Learning (pp. 89–106)*. Springer International Publishing.