Article

https://doi.org/10.34074/scop.4010008

EMBEDDING CRITICAL THINKING IN INFORMATION TECHNOLOGY NUMBER SYSTEMS STUDIES: REFLECTIONS ON TEACHING BINARY AND DECIMAL CONVERSIONS WITHIN A FOUNDATIONAL DEVELOPING COMPUTER APPLICATIONS COURSE

John Mumford

INTRODUCTION

Critical Thinking (CT) is integral to the teaching of information technology (IT) at the Southern Institute of Technology in undergraduate- and particularly at postgraduate-level programmes. The skills and dispositions involved in CT provide a lifelong toolkit for academic studies and for adaptation to the changing needs of technology in the workplace. The National Education and Learning Priorities and the Tertiary Education Strategy advocates CT at early childhood, primary, secondary and tertiary levels of study (Tertiary Education Council, 2020). Definitions and perspectives about CT vary, but its importance for students and teachers is likely to be enduring.

The Programme for International Student Achievement, which incorporates CT, aims to assesses the ability to skilfully apply reading, mathematics and science knowledge to address real-world problems (OECD, 2021). Dan Finkel, a speaker, math game designer, master teacher, writer and developer, who is dedicated to the transformation of mathematics education, confirms that, traditionally, within the mathematical domain, "There's no room to doubt or imagine or refuse – so there's no real thinking here" (TEDx, 2016). We need people who can think and problem-solve to deal with the rapidly changing world and continuously adapt themselves to new technologies. Now, more than ever, we need more critical thinkers. From school level to tertiary level, our numerical and data-rich world requires the application of CT and IT to empower individuals to manage their lives and careers. CT, like playing music is, to some extent, a set of skills that can be learned with guidance and practice. CT can also be applied to concepts, their rationale and, ultimately, to their underlying assumptions.

CRITICAL THINKING IN INFORMATION TECHNOLOGY

In the context of IT education, real-life challenges include computational thinking which, as a core skill, is more than just following or coding an algorithm. Doleck, Bazelais, Lemay, Saxena and Basnet (2017) identify five computational thinking competencies: algorithmic thinking, cooperativity, creativity, critical thinking and problem solving. Although it would be possible, but not desirable, to teach algorithmic thinking in isolation from CT, why not embed CT within number systems studies and create both present and future benefits for our students? Why should we follow mathematical procedures uncritically and often passively? Admittedly, we need proven algorithms that can be readily applied to given tasks to consistently produce the correct answers. However, let's think about how some notable scientists and thinkers with deep knowledge of their domains frequently question such approaches. Such questions can lead to new ways of understanding procedures and may open the way to less explored areas of science.

The concepts associated with mathematical procedures are the vital link to asking, why does it work? Or perhaps, why should it work? What are the logical bases for such concepts? We may even ask, why does it matter if we understand the concept underlying a procedure, if we can get the right answer? Admittedly, "Theory without practice is empty; practice without theory is blind" (Ako Aotearoa, 2021).

This article outlines the teaching strategies forming the basis for a series of lessons on number systems in IT, students' responses, teacher reflections and ways of promoting and modelling CT for a pre-degree course in computer applications.

CRITICAL THINKING IN THE CLASSROOM

The session started on a Monday morning on campus with ten New Zealand Certificate in Information Technology Essentials (Level 4) students. Noting that there were several mature students with prior computing experience among the school leavers in the class, I could see an opportunity to draw on their work and life experiences to inform a fresh approach to teaching number systems conversions. Some students would be revisiting semi-familiar material and others would be returning to education after some years of employment, and perhaps had never dealt with this topic. This seemed to be an ideal opportunity to present the material with a questioning approach at the outset to foreground CT, rather than defer it to more advanced courses. Students familiar with number systems conversion procedures would revisit why and what assumptions we start with, rather than just how we do the conversions. Mature students returning to study bring with them the benefits of life experience and frequently ask, Why do I need to learn number systems conversions? How could it be useful in an IT applications course?

The lesson commenced with a short sequence of 0's and 1's written on the whiteboard – for example, 101, followed by a prompt to the class to comment on what it means to them. This revealed the assumptions that were made (or had to be made) to create some agreed meaning. We then reviewed the values and positions of the decimal columns which frame our shared understanding of what we assumed had been written down. Thus, "learning takes place when new information is built into and added onto an individual's current structure of knowledge, understanding and skills" (Pritchard, 2009, p. 17). Naturally, this led on to further sessions involving the range of number bases – for example, the octal and hexadecimal systems used in computing and the implications that may flow from these.

The next class began illustrating the principles of the denary number system and how they might also apply to other systems such the binary system. We considered the value or 'weight' of the columns and the range of digits that could meaningfully be used within each system – for example, 0 to 9 for denary, 0 to 1 for binary. This then naturally led to noting that the number of columns needed was driven by the range of numbers that we needed to represent. Of course, bits and bytes were already familiar terms for the students, and thus groupings of 8 bits flowed from such thinking. We looked at several examples of numbers within these systems and observed that there seemed to be a relationship between the maximum number in each set of columns – for example, 99_{10} is the same as 100_{10}, the next column to the left, minus one. We tried this with binary, to see if it held true. For example, 3_{10} is the same as 4_{10} -1 or $011_2 = 100_2 - 001_2$ visually (without carrying or borrowing).

By this time, the class was beginning to get curious about how we could know which number system we were using at the outset. It was encouraging to see small glimpses of CT being shown here – what must be assumed to proceed with such number systems conversions? Do people always state explicitly what number base they are using? But thankfully the class felt that there was inconsistency here, and that computers were logical and completely consistent. We agreed that making a practice of explicitly writing the number base as a subscript (thus

101_2) would be better, especially when the base could be 8 or even 16. We began to see just how valuable CT is, even in the middle of 'number crunching.' In the context of experiential learning theory, teachers acting in the subject expert role "often teach by example, modelling and encouraging critical thinking" (Kolb & Kolb, 2017, p. 18). Several conversion examples were worked through with the students noting the importance of writing the number base explicitly and of taking care about verbalising incorrect statements – for example, 101_2 was not "one hundred and one" in binary, but "one zero, one" in base two (binary).

After experiencing other non-denary number systems in their hardware and web design classes, the students were beginning to see the benefits of grasping the principles of these differing (but related) number systems, and enjoying the buzz of becoming proficient in converting between them as required. It seemed that these curious number systems found uses in hardware addresses, network addresses, red green blue (RGB) hexadecimal number codes in web design, and even in programming.

We finished the week with a view to further develop converting between number systems and to discuss various algorithms or involve CT to question standard routines and to experiment with informal 'eyeballing' methods. The students thought that that sounded a little unstructured and not particularly 'algorithmic.' However, as they began to try out other strategies to convert between number bases with varying success, it started to become clear that established approaches to conversion had good bases. Despite knowing what would work, inviting them to experiment by trial and error helped the students to grasp the computer rationale for standard approaches, while acknowledging that a numerate human could work backwards from the number given to get the correct answer, without necessarily following a strict algorithm. This tapped into their creativity and problem-solving dispositions, while they co-operated on set exercises to arrive at the correct answers.

Slowly, but surely, the explicit embedding of CT was stimulating the students to approach an apparently routine topic with a more open, and sometimes playful, disposition. It was encouraging to see the beginnings of CT stirring within students, even in a foundational computing course. Also, it was reassuring to sense that the sessions catered for various learning styles (Honey & Mumford, 1986) including those who prefer to learn by doing (activists), those who stand back and observe (reflectors), those who like to see how things fit into frameworks and concepts (theorists) and students who are happy if a method works (pragmatists).

OBSTACLES TO LEARNING

Several obstacles were encountered as we worked through conversion examples. Using the standard 'divide and remainder' algorithm, the concept and practice of integer division caused some confusion. For example, using integers for simplicity, dividing say 10 by 2 for binary conversion repeatedly until we cannot (meaningfully) divide any further, generated some interesting discussion. The result sequence produced remainders of 0 and 1 at various steps. But why can't we take 5/2 = 2.5 and use this to arrive at the correct answer? Do we have to use division? What about starting with 10 and performing repeated subtraction until that process naturally finishes? These are all valid points and after some experimentation, the repeated subtraction method was most preferred. The students began to realise that we had made (or needed to make) simplifying assumptions to proceed with the logic of number systems conversion. Yet again, CT was surfacing with logic and assumptions.

The second obstacle was knowing which digit we start with to populate the columns to get a correct answer. Trying to fill the columns up starting at the unit (ones) column seemed easier – but then did not produce the right answer. Interestingly, even after the students became quite proficient with the conversion process, doubts were still present about where this could lead. Perhaps 0 and 1 could be used to indicate not only number values but electrical states, since digital devices used on/off-type processing. By being open to other perspectives, we were unwittingly making small steps in CT skill development.

CONCLUSION

Institutes of technology focus on vocational education and training. Application and qualification in IT courses aim to develop technical and soft skills for further study or employment. The embedding of CT in teaching and learning at the Southern Institute of Technology encourages students to challenge themselves and their often-hidden assumptions about the topics they study, the concepts that are needed and the dispositions that accompany them. This article has outlined some of the teaching and learning approaches I use with CT foregrounded, and my reflections on various perspectives of number systems conversions. All the students achieved their learning outcomes, with the added value of CT to take with them on their academic or employment journeys. Not only did the students graduate with their New Zealand Certificate in Information Technology Essentials (Level 4), but this course prepared them for undergraduate study and beyond by embedding CT in a relatively simple context of number systems conversions.

John Mumford is an Information Technology lecturer at the Southern Institute of Technology. His research interests include teaching innovation, mathematics education, adult literacy and numeracy and postgraduate Information Technology education. John has a Master of Adult Literacy and Numeracy and is committed to empowering learners to develop their critical thinking capabilities.

https://orcid.org/0000-0003-3000-7820

REFERENCES

Ako Aotearoa. (2021). *Teaching adults poster – Ako Aotearoa.* https://ako.ac.nz/assets/Knowledge-centre/RHPF-0913-Teaching-adults-posters/Poster-A-Theory-without-practice-is-empty-practice-without-theory-is-blind.pdf

Dolek, T., Bazelais, P., Lemay, D. J., Saxena, A., & Basnet, R. B. (2017). Algorithmic thinking, cooperativity, creativity, critical thinking, and problem solving: Exploring the relationship between computational thinking skills and academic performance. *Journal of Computers in Education, 4*(4), 355–369. https://doi.org/10.1007/s40692-017-0090-9

Honey, P., & Mumford, A. (1986). *Manual of learning styles* (2nd ed.). P. Honey.

Kolb, A. Y, & Kolb, D. A. (2017). Experiential learning theory as a guide for experiential educators in higher education. *Learning & Teaching in Higher Education, 1*(1), 7–44. https://nsuworks.nova.edu/elthe/vol1/iss1/7

OECD (2021). *What is PISA?* https://www.oecd.org/pisa/

Pritchard, A. (2009). *Ways of learning: Learning theories and learning styles in the classroom* (2nd ed). Routledge. https://www.researchgate.net/file.PostFileLoader.html?id=56c2ba4d7dfbf9d0548b45a3&assetKey=AS%3A329633892454402%401455602252915

TEDx. (2016, Feb 18). *Five principles of extraordinary math teaching | Dan Finkel* [Video]. YouTube. https://www.youtube.com/watch?v=ytVneQUA5-c

Tertiary Education Council (2020). *2020 statement of National Education and Learning Priorities (NELP) and the Tertiary Education Strategy (TES).* https://www.education.govt.nz/assets/Documents/NELP-TES-documents/FULL-NELP-2020.pdf