Osman Yaşar

# Viewpoint
# A New Perspective on Computational Thinking

*Addressing its cognitive essence, universal value, and curricular practices.*

THE IDEA OF adding computational thinking (CT) to a child's analytical ability goes back almost four decades,[20] yet its recent promotion[29] as an "attitude and skill set" for everyone has helped popularize it all over the world. While periodic reviews on the status of CT education[6,11] indicate wide agreement on what comprises CT, there is a struggle in the field by teachers and educators on how to integrate CT practices and skills into K–12 education. Many researchers and educators who initially supported the idea of teaching CT skills to everyone are now wary of its promise. Some of the remaining trouble spots include definition, methods of measurement, cognitive aspects, and universal value of CT.[6] This Viewpoint presents an alternative perspective on computational thinking, positioning CT as a link to cognitive competencies involved not only in science and engineering but also in everyday life.

A major source of current troubles with CT comes from linking it to electronic computing devices and equating it with thinking by computer scientists. Accordingly, many of currently recognized CT skills are associated with problem solving and use of electronic devices with a goal of preparing tomorrow's programmers.[11,29] A decade of discourse and experimentation has yet to produce ways to separate CT from programming and the use of electronic devices. And, the lack of such separation continues to preclude us from capturing the cognitive essence of CT.

Teaching experts' habits of mind to novices is inherently problematic because of prerequisite content knowledge and practice skills needed to engage in the same thinking processes, not to mention the cost of providing them a similar environment to conduct inquiry and design. This problem is not unique to CT education; it also applies to scientific thinking (ST) and engineering thinking (ET) education.[7,26] A remedy that applies to all of them is to link experts' habits of mind to fundamental cognitive processes so we can narrow the skillsets down to more basic competencies that can be taught to novices.[31]

Linking CT to cognition is not a new idea. In fact, it is what led to the design of electronic computing 80 years ago when Alan Turing[27] suggested that if thoughts (that is, information) can be broken up into simple constructs and algorithmic steps, then machines can add, subtract or rearrange them as our brains do. Electronic machines have since taken many complex and voluminous computations off our brains, further supporting the view of brain as a biological computational device.[19] Unfortunately, understanding how biological computing generates cognition from electrical activities of neurons has been hindered by the fact that it involves a delicate, inaccessible, and complicated organ, the brain. The good news is that technology has recently broken some of these barriers. For example, neuroscientists now use imaging techniques to understand brain mechanisms that take part in receiving, storing, retrieving, and processing information. Cognitive psychologists use similar techniques to study where in the brain particular perceptual and cognitive processes occur. At the same time, cognitive and computer scientists form theories and



**Figure 1. A universal mechanism by which all heterogeneous things form and evolve.[30]**

Concept

Basic concepts, details, and facts
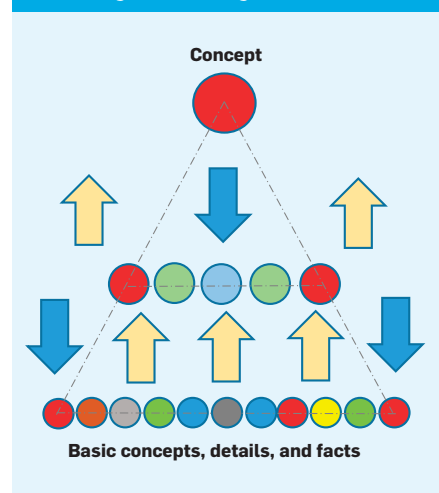
models of the mind to study how computation may be generating thinking.

Electronic computers have evolved to showcase many structural and functional similarities with the brain. So, we may have a chance to better understand how the brain works through easier access, use, and control of electronic devices. I suggest the similarities arise from quantifiable aspects of information constructs, as suggested by Alan Turing,[27] and the appearance of a universal mechanism (see Figure 1) by which quantifiable things form and evolve.[30] That is, like the granular matter, information constructs either unite associatively, as shown by the bottom-up arrows in Figure 1, to make bigger constructs or break down distributively, as shown by the top-down arrows, to smaller ones. Computing devices, be it electronic or biological, are likely to use similar ways to track and tally this invariant behavior of information. Another reason for similarities is the design, use, and control of electronic computing devices by biological computing agents.

Continuing the legacy of Turing to focus on device-independent processes (see Figure 2), we want to create more links between CT and cognition by identifying common patterns of information processing that are known to facilitate thinking. This may give us a framework to suggest a universal definition for CT—*thinking generated and facilitated by computation, regardless of the device that does the computation*—along with an electronic computing methodology to facilitate relevant cognitive processes. While the CS community is willing to modify its original definition along these lines,[1,28] current curricular CT practices still deal only with teaching of electronic CT skills.

A clear distinction should be made between electronic and biological CT to more effectively integrate desired CT skills to the relevant grade-level curricula. Having dealt with many issues of CT education for three decades at both college and K–12 levels,[30–34] I want to present an interdisciplinary perspective to address both cognitive and curricular aspects of CT by merging CS education research with concepts from epistemology, cognitive and neurosciences as briefly described here.

---

# Neuroscientists see little or no distinction now between the acts of information storage/retrieval and the act of thinking.

---

## Neuroscience's View of Information Storage, Retrieval, and Thinking

Contrary to the early compartmentalized and centralized design of electronic computers, the brain employs a distributed network of neurons to store, retrieve, and process information. Information gets stored into the memory in the form of a specific pattern of neurons placed on a pathway and fired together,[14] as shown in Figure 3. Therefore, the number and strength of neural pathways are key to improving storage and retrieval of information.

Humans are born with ~100 billion neurons that get connected to each other in various ways as we grow older. Other key factors that affect our mental growth include the functionality that each neuron or groups of neurons assume, the size they grow into, and the placement in different parts of the brain that they migrate towards. More important is the number of neural connections, which could go up to 100 trillion. As we learn things, new connections are being made while the existing ones are strengthened, weakened, or even eliminated if not revisited often enough.



**Figure 2. Information processing by electronic and biological computing devices include both device-independent and device-dependent processes.**

**Device-independent computational processes**

Electronic Computing

Biological Computing

---
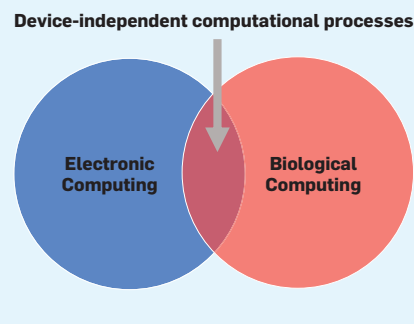
The latest developments in neuroscience have contributed significantly to our understanding of learning in relation to information retrieval.[4] Forgetting is now considered to be a good thing because it forces the learner to use effort to cognitively engage and recall or reconstruct newly acquired concepts through different neural pathways or links that exists and are retrievable. So, the more links to associated concepts, the higher the chances of recalling the newly acquired concept when needed later. Furthermore, cognitive retrieval practices attempted at different times, various settings, and contexts are good because every time the recall is attempted it establishes more links that will help the remembering and learning. Exposure to new concepts, then, through links to multiple views from different fields of study is an effective retrieval strategy recommended by cognitive psychologists.

Basically, retrieval sounds like an act of creative reimagination and what is retrieved is not the original pattern but one with some holes or extra bits. Consequently, neuroscientists see little or no distinction now between the acts of information storage/retrieval and the act of thinking. Such a consolidated view of storage, retrieval, and thinking is very much in tune with our model (Figure 1) of how information behaves naturally. Applying it to translate what neuroscientists say about storage and retrieval,[4] we posit that a memory or a newly learned concept can be a combination or outcome of previously formed memories and concepts, each of which might also involve another level of vast network of concepts and details mapped onto the brain's neural network in a hierarchical way. When new information arrives, it lights up all related cues, neurons and pathways in a distributive process that is similar to the top-down action, where new concept is broken up into related pieces. By the same token, retrieving a memory is a reassembly of its original pattern of neurons and pathways in an associative process that is similar to the bottom-up action.

Accordingly, the brain attempts to analyze *deductively* every new concept and information that it encounters in terms of previously registered models—*objects, faces, scenarios, and so on.*

And, as our knowledge grows further, the relationships among registered information eventually lead to interplay of various combinations and scenarios of existing models that eventually end up *inductively* clustering related details into conclusions, generalizations, and more inclusive models of information.[25] As a result, the details our brain registers and stores and the hierarchical connections it establishes between them, along with these generalizations and conclusions, build over time a pyramid-like structure (see Figure 1) that we have come to call *mind*.[19] Cognitive scientists often use a software analogy to distinguish it from the brain as noted here.
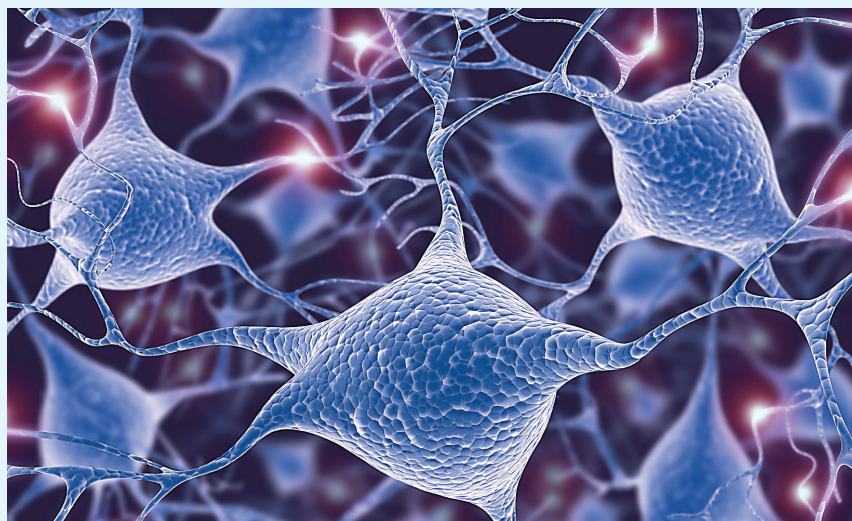
## Cognitive View of Information Processing

While the distributed structure of neurons and their connections (hardware) influence cognitive processing (software), the relationship between software (mind) and hardware (brain) is not a one-to-one relationship. According to the biological computing view of mind,[19] its processing of information consists of a hierarchy of many patterns and levels that may range from basic computations to more complex functions (sequence or structure of instructions) and models (mental representations) of perceived reality and imaginary scenarios.

While structural and functional similarities improve our understanding, I do not suggest the brain works exactly like electronic computers. Modeling the mind as a rational decision-making computational device has yet to fully capture mental representations and emotions.[10] In fact, we may never be able to model the human brain unless we understand what intelligence is and how it is possible for the human brain to make decisions on as much electricity as consumed by a dim light. Many believe it does so by simplification and avoidance of exhaustive computations and evaluations of hypothetical scenarios surrounding an issue.[13]

To explain the root causes of the brain's efficient operation, neuropsychologists and evolutionary biologists point to some structural (hardware) interference by an autopilot limbic system to bypass, simplify, or reduce more elaborate cognitive functions of



**Figure 3. Illustration of a distributed network of neurons firing and wiring together.**

an evolved neocortex. In fact, it appears we are caught up between two competing brains[24] whose operations can be well understood by the flow of information processing in Figure 1. Typically, there is a cyclical tendency between simplifying things (bottom-up) and digging things deeper (top-down). One of these processes is fast, effortless, automatic, inflexible, nonconscious, and less demanding of working memory, while the other is slow, effortful, controlled, conscious, flexible, and more demanding of working memory.[8]

Cognitive scientist Read Montague[19] points to some non-structural (software) tendencies to account for our brain's energy-efficient operation. He suggests that concern for efficiency, as part of our survival, leads to assigning value, priority, cost, and goals to our thoughts, decisions and action. To do this, the mind carries out computations, builds models, and conducts evaluative and hypothetical simulations of different scenarios. This may slow down and add imprecision to decision-making. However, because of bundling similar things together via a model, the overall process still ends up saving us from undertaking exhaustive and repetitive computations of various scenarios. According to Montague, the tendency to make trade-offs between simplicity and complexity and between details and generalizations is the root driver of our intelligence, and why we have pushed ourselves to be smarter over time.

Whether the causes are structural or non-structural, there is enough evidence about a duality in information storage, retrieval, processing, and reasoning that warrant further examination. In fact, both structural (hardware) and non-structural (software) drivers of our intelligence, reasoning, and thinking have a common mechanism that is consistent with Figure 1. For example, the act of modeling by our mind to assign value, cost, and goals to our thoughts before decision-making meshes well with the tendency for simplification (bottom-up flow of information in Figure 1). A clear advantage is that it makes it possible to work with approximate, abstract, or average representations, thereby bringing closure to an otherwise unending worry or inquiry about details. Accordingly, the human brain uses *modeling* not only for mental representation of external objects but also for wrapping up its own computations so it can compare their values and costs before deciding,[13] a cognitive mechanism that epistemologists came up with two centuries ago, as noted here.

## Epistemology of Knowledge Development

Epistemology is a branch of philosophy that studies *how we know what we know*. At its core are questions like '*what is true knowledge and its source?*' and '*how can we be sure of what we know?*' While scientists such as Galileo laid a strong foundation for building knowledge

through observations, experiments, and mathematics in the 16th century,[18] philosophers debated for two more centuries whether a scientist's subjective view of the world can be considered as true knowledge.

One of the debated views (empiricism) argued that the mind is a blank slate and that it acquires knowledge through perception and *inductive reasoning*, which involves putting perceptions, experiences and related pieces of information in a synthetic (associative) way to arrive at generalizations and conclusions as depicted by the bottom-up flow (arrows) of information in Figure 1. Knowledge acquired this way is not warranted because new experiences may later change its validity. The other view (rationalism) argued that knowledge is initially acquired through innate concepts which then serve as the source of additional knowledge derived from them in a rational (analytic) way using *deductive reasoning*. In deductive reasoning, a concept generally applies to all members and situations that fall under its representation, as depicted by the top-down flow (arrows) of information in Figure 1. Since innate concepts were considered true, knowledge derived from them was considered to be warranted, not needing further examination.

By arguing against both views, Immanuel Kant created a bridge to lay the foundations of epistemology and today's scientific methodology of inquiry.[15] He recognized what experience brings to mind as well as what mind itself brings to experience through structural representations. He considered that knowledge developed a posteriori through synthesis could become knowledge a priori later. And, a priori cognition of the scientist continues to evolve over the course of science's progress. Although the deductive and inductive cycle of scientific progress has historically been slow until recently,[18] the growing knowledge and the number of researchers tackling a problem have all now shortened the timescale of progress. Concepts and theories once considered true and valid are now quickly being changed or eliminated.

### Modeling: A Universal Process
Modeling and testing has been an important tool for scientific research for

> ## Concepts and theories once considered true and valid are now quickly being changed or eliminated.

hundreds of years. In principle, it works exactly as articulated by Kant, and as illustrated in Figure 1. Scientists ideally start with a model of reality based on current research, facts, and information. They test the model's predictions against experiment. If results do not match, they then break down the model deductively into its parts (sub models) to identify what needs to be tweaked. They retest the revised model through what-if scenarios by changing relevant parameters and characteristics of the sub models. By putting together inductively new findings and relationships among sub models, the initial model gets revised. This cycle of modeling, testing, what-if scenarios, synthesis, decision-making, and re-modeling is repeated while resources permit until there is confidence in the revised model's validity. Electronic computers have recently accelerated this cycle because not only do they speed up the model building and testing via simulations but they also help conduct studies that are impossible experimentally due to size, access, and cost.

Modeling and simulation (M&S) appears to be a device-independent process of information that links computing and cognition. Its associative and distributive processes even describe computable actions of other quantifiable things besides information. For example, formation of physical objects or particles from smaller ones resembles the act of modeling because both seem to involve packing *parts* together associatively to form a *whole*. Furthermore, such act of modeling is often driven by external forces or by a collective "trial and error" process controllable by conditions and rules of engagement—much like a simulation.[30]

Philosophers and psychologists have been studying the *parts-and-whole* dynamics since Plato[9] to explain the nature and human behavior. Recently, with help from technology, cosmologists and cognitive scientists have also been searching for a universal process that may be guiding the growth of all networked systems, ranging from the tiny brain cells to atoms, to the Internet, and even the galaxies. The view that such a process may be described computationally, as in Figure 1, is now gaining traction because formation and evolution of an abstract idea or a computational model of information appears to be no different than that of a system of physical particles.[30] If so, then not only can we learn from an ongoing millennial argument of such a universal topic, but we can also put computing at the center of a discourse well beyond CT to understand the nature itself.

### The Essence of Computational Thinking
Our brain's inclination to store, retrieve, and process information in an *associative/distributive* fashion may be a manifestation of a duality engrained in the fabric of matter and information. This inclination may just be an evolutionary response, shaped up over many years, to optimize the handling of sensory information whose quantifiable nature only resonates with distributive and associative operations. A similar evolution can be seen in electronic computing's structural change from a centralized hardware of the past to today's distributed network due to the growing need for faster processing and more storage to solve problems and improve our survival. As Montague suggests,[19] our changing need for simplicity and generalization (via associative processing) as well as complexity and details (via distributive processing) of information has driven us to think harder and become smarter. At the same time, while our brain structure and cognitive processing offer all of us a chance for full utilization of an optimized response to a changing environment, the efficiency, intactness, and effortfulness with which we all use it depends on the individual.

At the core of our CT framework in Figure 4 lies a dichotomy both in the quantifiable nature of sensory infor-

mation and in the way information storage, retrieval, and processing is done by the brain hardware. Since cognitive researchers have demonstrated how information processing could lead to cognitive inferences via inductive reasoning,[7,25-26] here we are not concerned about details of how information processing generates cognition but rather how duality in fundamental computation may lead to duality in higher-level reasoning. The invariant nature of information affects not only how similarly computing is done at the most fundamental level (that is, addition and subtraction at the core of our framework), but how this similarity would carry itself all the way to the high-level processing at the outer layers in Figure 4. Despite these similarities, however, high-level processing of information with different devices may still have device-dependent aspects, therefore requiring different skills to use each. Basically, the duality in information processing and *its cyclical and iterative use, as in Figure 1, is the very essence of computational thinking* that we all employ for learning, conceptual change, and problem solving. Anyone who wants to use electronic devices to further facilitate this process might need electronic CT skills on top of biological CT skills, as shown in the last layer of Figure 4.

Our framework's relation to habits of mind in other fields, such as ST and ET, has been reviewed favorably by relevant communities.[31] In fact, ST's inductive and deductive processes are no different than those used in everyday thinking by non-scientists.[7,26] We all use inductive reasoning to filter out details and place our focus on more general patterns, thereby assigning priority and importance to the newly acquired information. Deductive reasoning, on the other hand, helps us make decisions and draw conclusions from general concepts. Yet, using these CT skills in an iterative and cyclical fashion for inquiry and conceptual change varies for everyone, depending on the underlying brain structure and the quality and quantity of the environmental input it receives. A scientist is a good example of someone who does this in a frequent, consistent, and methodological way, leading eventually to a habit of mind that is often known as ST.

The two currently cited electronic CT skills that resonate with cognitive functions of a computational mind, as we defined here, are *abstraction* and *decomposition*. The rest can be considered as device-dependent skills that may or may not have any cognitive benefits other than being a routine or specific use of an electronic device. Abstraction is an *inductive* process that helps our cognition in important ways, especially at its developmental stages, by simplifying, categorizing, and registering key information for quicker retrieval and processing. Decomposition, on the other hand, is a *deductive* process that also helps us in many ways, including dealing with a complicated situation by distributing the complexity into smaller and simpler pieces in order to attack each one separately until a cumulative solution is found.

We all use abstraction and decomposition skills in our daily lives,[3] but not everyone is equally aware of their importance, nor are we all practicing and utilizing them fully and equally. In that sense, everyone, not just computer scientists, uses CT. But, since abstraction and decomposition skills are heavily used in programming and problem solving,[2] having students improve them has been a concern of educators. For example, abstraction is used to distribute the complexity of a code *vertically*, as shown in Figure 1, into seemingly independent layers and protocols in such a way to hide the details of how each layer does the requested service. Dijkstra, a pioneer in programming, regarded abstraction as the most vital activity of a competent programmer. In fact, a good programmer is expected to be able to oscillate between various levels of abstraction.[2] While being able to divorce one's thinking from low-level details and biases is key to finding solutions that can be transformed to different applications, most CS undergraduate students barely move beyond language and algorithm-specific details and biases. Similarly, decompo-

**Figure 4. A cognitive framework on the essence of electronic CT skillset in terms of biological CT skills.**[31]

sition is used in software engineering as well as in parallel computing to distribute the workload *horizontally*, as in Figure 1, among multiple processors. Unfortunately, automatic compilers are not here yet to help us write parallel codes, and teaching students parallel programming is still a challenge. There are no quick fixes but as mentioned in the next section M&S tools have been found to boost not only students' cognitive functions but also their motivation to learn programming and science content.

### Measuring the Impact of CT Education

There are instruments with good psychometric properties to measure the impact of technological pedagogical content development[17] tools on teaching and learning. M&S's interdisciplinary and changing technological nature require customization of its use in instruction and the assessment of its effectiveness in teaching of the content under consideration. Researchers may need to use not only quantitative methods to measure variables involved but also qualitative methods to initially identify those variables and to later understand and triangulate them for validity. The quantitative sources of data often include surveys to gather pre/post activity data, unit test scores, course passing rates, report cards, graduation rates, and achievement scores in standardized tests, while qualitative sources of data may be interviews, classroom observations, and computational artifacts.[5]

Education researchers have identified M&S as an exemplar of inquiry guided learning.[21,23] These findings are also grounded in learning theories that recognize the role of abstract thinking and reflection in constructing knowledge and developing ideas and skills.[3] However, because constructivist and unguided learning works only when learners have sufficiently high prior content knowledge to provide "internal" guidance,[16] use of M&S in K–12 education has been slow. Technological changes in the past decade have given birth to new M&S tools that can shield the learner from high-level content knowledge in math (for example, differential equations), computing (such as programming), and science

(for example, laws of nature), thereby making them accessible to novices for constructive learning.

As noted in peer-reviewed articles,[32,33] empirical data collected from hundreds of teachers and their students in 15 secondary schools for a period of seven years revealed statistically significant results to suggest that M&S inherently carries a mix of deductive and inductive pedagogies in the same setting. This is great news for educators who want to take advantage of both approaches of teaching. Basically, modeling provides a general simplistic framework from which instructors can *deductively* introduce a topic without details, and then move deeper gradually with more content after students gain a level of interest to help them endure the hardships of effortful and constructive learning. Simulation, on the other hand, provides a dynamic medium to test the model's predictions, break it into its constitutive parts to run various what-if scenarios, make changes to them if necessary, and put pieces of the puzzle together inductively to come up with a revised model. This kind of iterative and stepwise progression is consistent with psychology of optimal learning which suggests balancing skills and challenges.[3] Anyone who learns in this iterative cycle of inductive and deductive reasoning would, in fact, be practicing the craft of scientists.

Measuring the impact of M&S on generating awareness of and appreciation for abstraction and decomposition skills, particularly in their relation to programming, needs further study. A question would be: Once learners

---

**Anyone who learns in this iterative cycle of inductive and deductive reasoning would, in fact, be practicing the craft of scientists.**

---

gain experience and fun creating artifacts (for example, models or videogames) with M&S, could this help them develop an interest to look for mathematical, computational and scientific principles under its hood? Some afterschool studies report encouraging results in teaching students textual programming in the process of creating videogames that connect to K–12 math and science learning outcomes.[22] A quasi-experimental study of ours reports[32,33] similar preliminary findings, as briefly explained next.

Annually, 50 teachers taught math and science topics using M&S tools during formal instruction. Teams of four students selected by each teacher received additional afterschool instruction from college faculty on mathematical principles of modeling (that is, new = old + change) as well as basic programming (in Excel and Python) to construct hands-on simulations. A panel of experts scored team projects and coded narratives to find common themes.[5] According to these, hands-on modeling helped students realize the virtue of decomposition in problem solving, because finer decomposition led to more accurate answers. Other emerging themes included observations that textual programming provided better control of the decomposition (and desired accuracy) as well as easier coding (for example, via a simple loop). Finally, since computation of change in position, velocity, and acceleration necessitated a scientific formula to compute acting forces, this appeared to help students link computing and natural sciences. According to student interviews, it motivated them to plan on taking science and computing courses in later years. Follow-on quantitative data supported these anecdotal findings.[33] For example, while no physics courses were offered before in any of the 13 high schools of the urban school district, they became part of curricular offerings in two of them. The number of students taking general physics in the suburban high school increased by 50%. Also, the afterschool program led to design of a new computing course in one of the urban high schools, drawing high enrollment for three years until the teacher took a lucrative job in industry.

## Conclusion

An interdisciplinary perspective on the cognitive essence of CT has been presented here based on the distributive and associative characteristics of information storage, retrieval, and processing by a network of neurons whose communication for searching, sorting, and analogies is driven by neural connectivity, richness of cues, a trade-off between simplification and elaboration, and a natural tendency to minimize energy usage. This broad approach might help clear some of the trouble spots with CT while putting it on a higher pedestal through a link to cognitive competencies involved in science and engineering.

Everyone cognitively benefits from CT by the virtue of having a computational mind. All we need is to help them use it in a more systematic way in their lives and professions. Since M&S facilitates an iterative and cyclical process of deductive and inductive reasoning, it could be used to teach novices not only critical CT skills (for example, abstraction and decomposition) but also ST and ET skills, including formation and change of hypothesis, concepts, designs, and models. While these are no different than cognitive processes of ordinary thinking,[26] not everyone uses them as consistently, frequently, and methodologically as computer scientists, natural scientists and engineers. The good news is they can be improved later through training and education.

CT's universal value is far beyond its relation to cognition. I argue that all heterogeneous stuff behaves computationally, regardless of what drives it. And, iterative and cyclical form of such behavior appears to be the essence of natural dynamism of all discrete forms. M&S is such a pattern, and putting computation in this fashion at the heart of natural sciences provides an opportunity to claim that computer science deals with natural phenomena, not artificial (digital). The computational revolution started by Turing may eventually be how our knowledge can come together to make more sense of our world.

One of the calls for action here for the CS community is to put more emphasis on M&S as a crucial part of student practice and education. This may help pave the way to teach

> **The computational revolution started by Turing may eventually be how our knowledge can come together to make more sense of our world.**

computing principles to non-CS students.[12] Furthermore, while educational researchers have done a good job of measuring the impact of M&S on learning, a focus by the CS community can help generate interest among educational researchers to do similar research by measuring M&S's impact on conceptual change, abstraction, decomposition, and metacognitive skills, particularly in relation to CT and programming education. The second call is that prior to teaching students electronic CT skills, we need to teach them a habit of conceptual change through iterative and cyclical practices of inductive and deductive reasoning. Besides M&S tools, researchers should explore other modular and scalable design toys as well as reading and writing practices to offer similar CT practices. **C**

### References
1. Aho, A. Computation and computational thinking. *The Computer Journal 55*, 7 (Jul. 2012), 832–835.
2. Armoni, M. On teaching abstraction to computer science novices. *J. Comp in Math & Science Teaching 32*, 3 (Mar. 2013), 265–284.
3. Bransford, J., Brown, A., and Cocking, R. *How People Learn.* National Academy Press, Washington, D.C., 2000.
4. Brown, P., Roediger, H., and McDaniel, M. *Make it Stick.* Belknap Press of Harvard, 2014.
5. Creswell, J.W. *Educational Research.* 4th Edition. Pearson Education, Inc., 2012.
6. Denning, P. Remaining trouble spots with computational thinking. *Commun. ACM 60*, 6 (June 2017), 33–39.
7. Dunbar, K. and Klahr, D. Scientific thinking and reasoning. In K. Holyoak and R. Morrison, Eds., *The Oxford Handbook of Thinking and Reasoning.* Oxford University Press, London, 2012, 701–718.
8. Evans, J. and Frankish, K. *In Two Minds: Dual Processes and Beyond.* Oxford University Press, Oxford, 2009.
9. Findlay, S.D. and Thagard, P. How parts make up wholes. *Frontiers in Physiology 3*, 455 (2012).
10. Goleman, D. *Emotional Intelligence.* Bantam Dell, New York, 2006.
11. Grover, S. and Pea, R. Computational thinking: A review of the state of the field. *Educational Researcher 42*, 1 (Jan. 2013), 38–43.
12. Guzdial, M. Paving the way for computational thinking. *Commun. ACM 51*, 8 (Aug. 2008), 25–27.
13. Hawkins, J. *On Intelligence.* Times Books, New York, 2004.
14. Hebb, D. *The Organization of Behavior.* Wiley, New York, 1949.
15. Kant, I. *The Critique of Pure Reason.* (J.M.D. Meiklejohn, Trans.). eBook@Adelaide, The University of Adelaide Library, Australia, 1787.
16. Kirschner, P.A., Sweller, J., and Clark, R.E. Why minimal guidance during instruction does not work. *Educational Psychologist 41*, 2 (Feb. 2006), 75–86.
17. Koehler, M., Shin, T., and Mishra, P. How do we measure TPACK? In R.N. Ronau, C.R. Rakes, and M.L. Niess, Eds., *Educational Technology, Teacher Knowledge, and Classroom Impact* IGI Global, Hershey, PA, 2012, 16–31.
18. Kuhn, T. *The Structure of Scientific Revolutions.* U. Chicago Press, Chicago, 1962.
19. Montague, R. *How We Make Decisions.* Plume Books, New York, 2006.
20. Papert, S. *Mindstorms: Children, Computers, and Powerful Ideas.* Basic Books, New York, 1980
21. Rutten, N., van Joolingen, W.R., and van der Veen, J.T. The learning effects of computer simulations in science education. *Computers & Education 58*, 1 (Jan. 2012), 136–153.
22. Schanzer, E., Fisler, K. and Krishnamurthi, S. Bootstrap: Going beyond programming in after-school computer science. SPLASH Education Symposium, Claremont, CA., 2013.
23. Smetana, L.K. and Bell, R.L. Computer simulations to support science instruction and learning. *Int. J. Science Education 34*, 9 (Sept. 2012), 1337–1370.
24. Sun, R. *Duality of Mind.* Lawrence Erlbaum Associates, Mahwah, NJ, 2002.
25. Tenenbaum, J.B., Kemp, C., Griffiths, T.L., and Goodman, N.D. How to grow a mind: Statistics, structure, and abstraction. *Science 331*, (2011), 1279–1285.
26. Thagard, P. *The Cognitive Science of Science.* The MIT Press, Cambridge, MA, 2012.
27. Turing, A.M. *On Computable Numbers, with an Application to the Entscheidungs problem.* In *Proceedings of the London Mathematical Society 2*, 42 (1937), 230–265.
28. Wing, J.M. Computational thinking—What and why? *The Link Magazine* (Mar. 06, 2011).
29. Wing, J.M. Computational thinking. *Commun. ACM 49*, 3 (Mar. 2006), 33–35.
30. Yaşar, O. Modeling and simulation: How everything seems to form and grow. *Comp. in Sci. and Eng. 19*, 1 (Jan. 2017), 74–77.
31. Yaşar, O., Maliekal, J., Veronesi, P. and Little, L. The essence of scientific and engineering thinking and tools to promote it. In *Proceedings of the American Society of Engineering Education Annual Conference*, 2017.
32. Yaşar, O. and Maliekal, J. Computational pedagogy. *Comp. in Sci. and Eng. 16*, 3 (Mar. 2014), 78–88.
33. Yaşar, O., Maliekal, J., Veronesi, P., and Little, L. An interdisciplinary approach to professional development of math, science and technology teachers. *Comp. in Math & Sci. Teaching 33*, 3 (Mar. 2014), 349–374.
34. Yaşar, O. and Landau, R. Elements of computational science and engineering education. *SIAM Review 45*, 4 (2003), 787–805.

**Osman Yaşar** (oyasar@brockport.edu) is an Empire Innovation Professor at the State University of New York, The College at Brockport.