

See discussions, stats, and author profiles for this publication at: <https://www.researchgate.net/publication/371315550>

Prompting is Computational Thinking

Article · June 2023

CITATION

1

READS

384

2 authors:



[Alexander Repenning](#)

University of Applied Sciences and Arts Northwestern Switzerland

153 PUBLICATIONS 3,851 CITATIONS

SEE PROFILE



[Susan Grabowski](#)

École Polytechnique Fédérale de Lausanne

34 PUBLICATIONS 18 CITATIONS

SEE PROFILE

Prompting is Computational Thinking

Alexander Repenning¹, Susan Grabowski²

¹PH FHNW, School of Education, Bahnhofstrasse 6, 5210 Windisch, Switzerland

²EPFL, AVP-E LEARN, Station 9, 1015 Lausanne, Switzerland

Abstract

Prompting, a term informally introduced into the English language to describe the iterative process of writing or modifying a prompt to an AI system such as ChatGPT or Midjourney, could be considered a new kind of Computational Thinking. Computational Thinking describes a process of convergence combining human ability with computer affordances. In essence, Computational Thinking is an iterative process of humans engaging in problem-solving by thinking with computers. In the current literature Computational Thinking is often associated with programming. However, as we are trying to demonstrate here, the perspective that “Prompting is Computational Thinking” can help to elevate the notion of Computational Thinking to become a useful framework to promote the convergence of Human Intelligence and Artificial Intelligence.

Keywords

Computational Thinking, programming, convergence of human and artificial intelligence, ChatGPT

1. Introduction

The invention of machines offering functions assumed to be expressions of human intelligence has a long history. For instance, the ability to perform arithmetic operations was assumed to be an expression of human intelligence. The invention of the world's first automatic calculating machine [5] by Blaise Pascal, known as the Pascaline, had a significant impact on society during the 17th century. While the exact reception varied among different segments of society, the overall response was positive and marked an important milestone in the development of mechanical calculators. The term Artificial Intelligence, of course, did not exist at that time but could be described retrospectively to capture the nature of that innovation at the time. The Pascaline did not immediately revolutionize society or replace human calculators, as its cost and limited functionality restricted its widespread adoption. Technology evolved and gradually the performance of machines exceeded that of humans. Machines started to beat humans in Chess [3], ten years later Go [11] and many other human abilities. As the performance of computers is increasing exponentially and neural nets are being trained on previously unimaginable large data sets, Artificial Intelligence is evolving at an alarming speed. Ray Kurzweil called the point at which Artificial Intelligence would exceed Human Intelligence the Singularity [6].

The Singularity may not be here just yet but developments such ChatGPT make it quite clear that the impact of technology on society is much more profound and that technology is moving at previously unimaginable speeds. Unlike with Blaise Pascal's calculating machine this technology is affordable and not just relevant to the intellectual elite. In other words, it is extremely likely for AI to transform the very fabric of society. Some time ago, in his book “Program or be Programmed” Rushkopf [9], defined a critical ultimatum underlining the urgency of Digital fluency [2]. Essentially, if one is not willing to take control of technology, i.e., “to program”, then the machines will take over and “program” humans. While his position may be extreme it illustrates possible consequences of ignoring technological development. How can society coexist with seemingly superior technology where machine intelligence exceeds human intelligence?

Going back to the examples of games we can observe two fundamentally opposite strategies to deal with the looming Singularity: *capitulation* and *convergence*. Lee Sedol, after losing against Google's AlphaGo decided to capitulate. He saw no more value in further pursuing this career as a Go player. After all, in the context of

IS-EUD 2023: 9th International Symposium on End-User Development, 6-8 June 2023, Cagliari, Italy

EMAIL: alexander.repenning@fhnw.ch (A. 1); susanne.grabowski@epfl.ch (A. 2)

ORCID: 0000-0002-2165-7533 (A. 1)



© 2023 Copyright for this paper by its authors.

Use permitted under Creative Commons License Attribution 4.0 International (CC BY 4.0).



CEUR Workshop Proceedings (CEUR-WS.org)

Go machine intelligence had already eclipsed human intelligence. As machine learning further progressed this gap could only widen. Gary Kasparov, in contrast, after losing against IBM's Deep Blue was even further motivated to play Chess. Continuing to play against Deep Blue he claimed to have significantly advanced his Chess playing skills. This convergence model seeks to synergistically combine human intelligence with Artificial Intelligence into a total intelligence exceeding human abilities as well as computer affordances.

This paper introduces the idea that prompting is Computational Thinking, explains how Computational Thinking helps with the convergence of human abilities and computer affordances, provides some examples of Prompting and finally suggests some early principles of prompting.

2. Prompting versus Computational Thinking

Computational Thinking (CT) is an iterative process synergizing human abilities with computer affordances. The field of Computer Science education research [10] is still debating what exactly CT really is [7] and how it is different from Computer Science and programming. However, there are some common characteristics describing the CT process. Figure 1 shows the original three “A” CT process [8] consisting of the abstraction, automation and analysis stages. This figure is being shared widely, including in Wikipedia.

1. **Abstraction: Problem Formulation:** Problem formulation attempts to conceptualize a problem verbally, e.g., by trying to formulate a question such as “How does a mudslide work?,” or through visual thinking, e.g., by drawing a diagram identifying objects and relationships.
2. **Automation: Solution Expression:** The solution needs to be expressed in a non-ambiguous way so that the computer can carry it out. Computer programming enables this expression. The rule in Figure 1 expresses a simple model of gravity: if there is nothing below a mud particle it will drop down.
3. **Analysis: Execution & Evaluation.** The solution gets executed by the computer in ways that show the direct consequences of one’s own thinking. Visualizations, for instance the representation of pressure values in the mudslide as colors, support the evaluation of solutions.

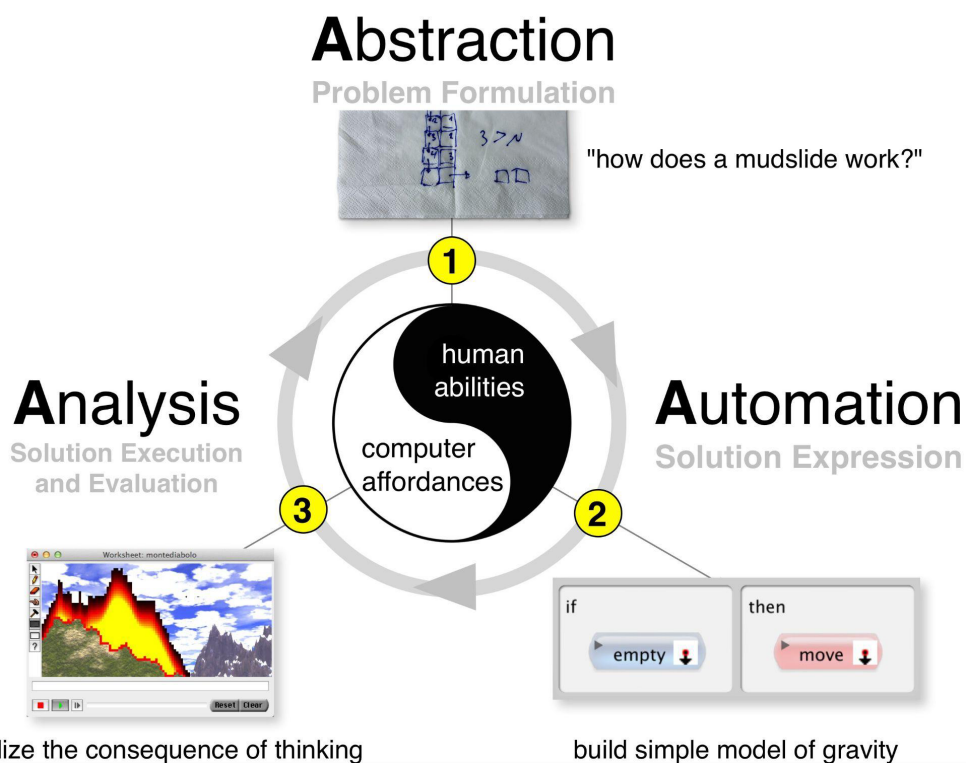


Figure 1: The Computational Thinking Process

The Yin and Yang in the center of the Computational Thinking process (Figure 1) describe the two complementary forces which make up the synergy between human abilities and computer affordances. There are CT models such as CT unplugged [1] exploring Computational Thinking without the use of computers. However, for our purposes to explore the role of AI we are exclusively interested in use cases where we do have computers.

The Computational Thinking process can now be reconceptualized as prompting (according to the Urban Dictionary this term has been introduced informally into the English language). By prompting we refer to *the process of writing or modifying a prompt to an AI system such as ChatGPT or Midjourney*. Lets now revisit the 3 stages of CT from the Prompting angle with an explicit eye on the role of humans and computers:

1. **Abstraction: Problem Formulation:** again, the human drives the CT process by starting with a question. We use the same question from before “How does a mudslide work?”
2. **Automation: Solution Expression:** Unlike with our previous example the user is not planning to program a simulation of a mudslide but is planning to use one or more AI systems to create some kind of output. Users may have different formats of output in mind. Do they want text, images, videos, audio, code or perhaps even some combination? The user requirements combined with pragmatic concerns such as previous experience will determine which AI system users will use and how they will formulate a promising prompt. The initiative here is still human. Users need to select, and combine AI systems. They need to express a prompt.
3. **Analysis: Execution & Evaluation.** An AI system may create one or more responses which users will have to evaluate. The analysis stage may be quite simple such as picking an interesting looking picture produced by an AI system such as DALL-E, or it may be quite intricate such as reading and perhaps even running code produced. The connection between input (prompt) and output may be difficult to understand and in some cases even unpredictable. For this stage to be productive humans will need to have some understanding of the various affordances of the different AI systems.

A single iteration is unlikely to produce the desired result. Users may have to adjust their problem formulation (abstraction), change the expression of their prompt (automation), or further process the output of the AI (analysis). That is, they will engage in prompting. In the next sections we present some prompting examples.

3. 3. Prompting Experiments

Here are some basic examples of prompting.

3.1 Mudslide

Abstraction: The user is an elementary school teacher trying to teach the concept of mudslides to 4th graders in a STEM class. She starts the CT process by formulating the original question from Figure 1: “How does a mudslide work?”.

Automation: Moving on to automation she picks ChatGPT 3.5 and uses the main question 1:1 as prompt “How does a mudslide work?”

Analysis: The output is elaborate text introducing the general notion of a mudslide

A mudslide, also known as a debris flow or a landslide, occurs when a mass of soil, rock, and water rapidly moves down a slope. It is typically triggered by heavy rainfall, snowmelt, earthquakes, or volcanic activity. Here's a general explanation of how a mudslide works:

A bullets list with 6 points including “Triggering event”, “Saturation and liquefaction”, “Failure of slope material”, “Movement”, “Expansion and destruction”, and “Deposition” follows. After a brief analysis, the teacher decides that the output is too complex for 4th graders.

Automation 2: She keeps the abstraction but expands the prompt: “please simplify for an 8 year old and be brief”

Sure! A mudslide is like a big, heavy pile of wet dirt and rocks that suddenly moves down a hill very quickly. It happens when there is a lot of rain or water, and the ground gets too weak and slippery. The dirt and rocks can destroy things in their path like trees, houses, and roads. After it stops moving, it leaves behind a big mess. It's important to be careful and stay away from areas where mudslides can happen.

Automation 3: This is a good start but to get a more hands on experience for her students the teacher would like to see a simulation. She prompts: “write a particle simulation of a mudslide using JavaScript”

She got a short JavaScript program from ChatGPT including some explanation of what it does “The particles are randomly generated with different velocities and sizes, and they fall from the top of the canvas to the bottom, simulating the movement of a mudslide.” She knows how to copy the program into the browser. The code actually runs but there is no evidence of a mudslide. The teacher does know a little bit of JavaScript but is not able nor does she have the time to fix the code.

Somewhat frustrated, the teacher wishes to at least draw a picture of a mudslide. Using DALL-E via the BING interface she enters the “mudslide” prompt only to be greeted by a “This prompt has been blocked. Our system flagged this prompt because it may conflict with our content policy” BING message.

3.2 DynaBook

Alan Kay’s 50+ year old vision of educational computing devices for children was called DynaBook [4] depicting Jimmy and Beth using hand held devices including networking and programming: “Two nine-year-olds were lying on the grass of a park near their home, their DynaBooks hooked together.”

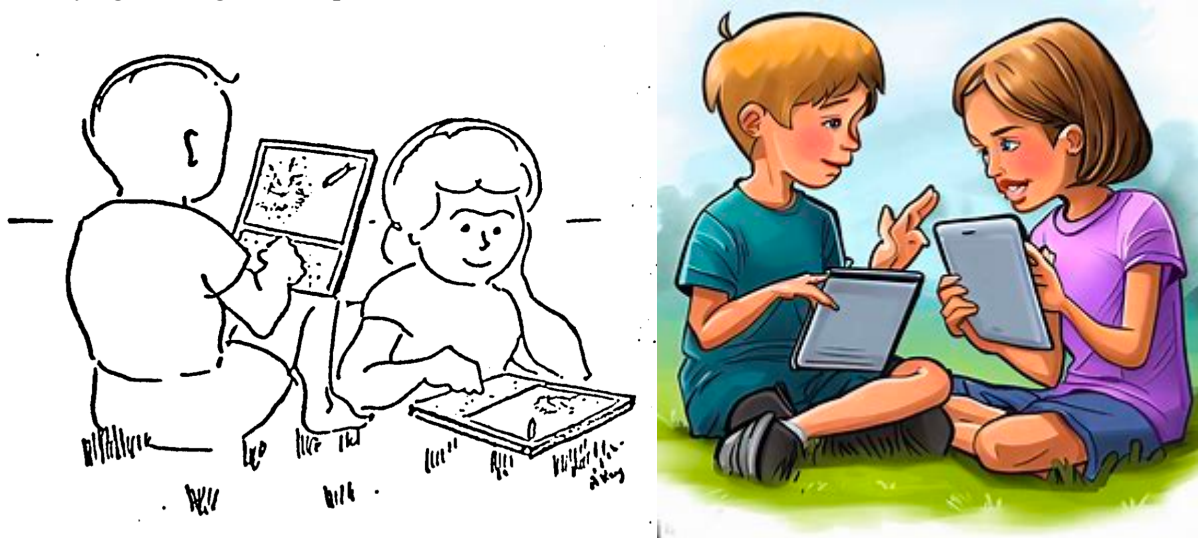


Figure 2. DynaBook: left drawing by Allan Kay 1972; right: AI DALL-E (via MS Bing) 2023

With more than 20 iterations within the Computational Thinking process (Figure 1), Prompting for the image to the right was quite involved. The prompt used for Figure 2 right was “two 9 year old kids collaborating with each other using iPads, sitting outside in the grass, cartoon style.” Without the “cartoon style” part DALL-E (via MS Bing) flat out refused to create an image with the warning “Unsafe image content detected.” The same was not true when using DALL-E directly. It is not clear why this image was considered unsafe.

The challenge suggested by this example is a high degree of unpredictability inherent to prompting. That is, the precise mapping from step 2 in Figure 1 (automation) to step 3 (analysis) can be difficult to comprehend because of the unforeseeable nature of many of the AI tools. Especially for the unexperienced users the result can be quite a surprise.

3.3 Modern Art

The third experiment is trying to use DALL-E to replicate the work of OP-Art artist Bridget Riley, who created the work "Shiver" in 1964 (Figure 3). In this experiment prompting consist of abstracting the geometrical principles of the original picture (Figure 1, step 1), automating the image by formulating prompts (step 2), analyzing the image generated (step 3) and repeating the CT process (steps 1 – 3) as many times as necessary.

The work consists of a simple matrix of 31 x 31 grid squares. In each white field is a black triangle. From row to row, the triangles are arranged and twisted slightly differently. The first and last rows are identical horizontally and vertically.

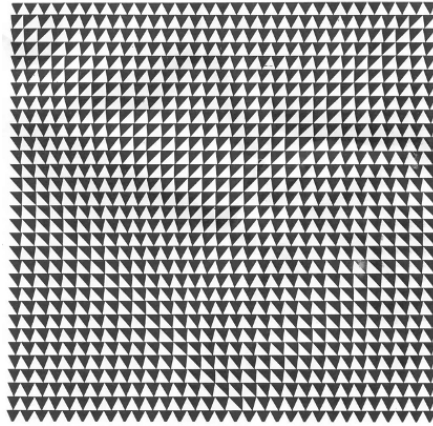


Figure 3: Bridget Riley: Shiver 1964

Provided the relative scarcity of online materials by Bridget Riley we assumed DALL-E would only have limited training data. To simplify the task, the matrix was reduced in size to 20 x 20 cells.

Prompt 1: “Generate a 20 x 20 grid. Each grid-contains a black triangle. Triangles sit-in a horizontal row are the same position. Vertically, the triangles shift minimally.”

The AI generates a matrix containing triangles. However, it does not follow the more specific description.

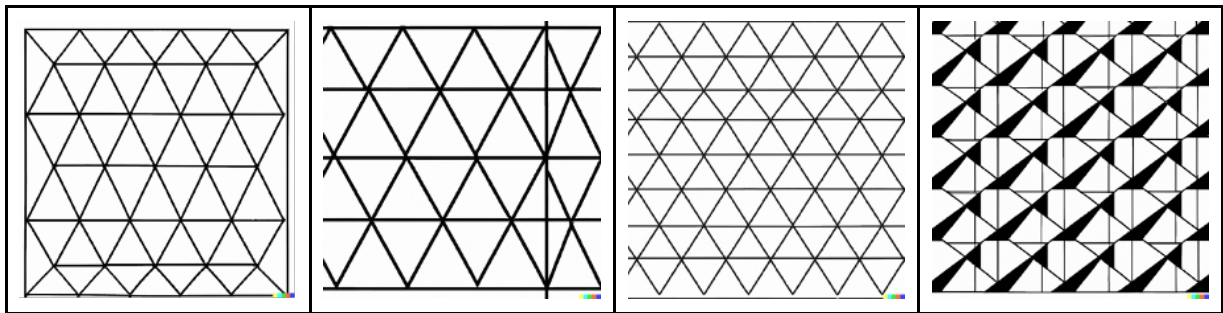


Figure 4: Four results of using DALL-E with prompt 1

Prompt 2: “Generate a 20 x 20 grid. Each grid contains a triangle filled with black, pointing downwards. Within a horizontal row, the triangles are in the same position. Vertically, the triangles in the rows are shifted minimally to the right or left.”

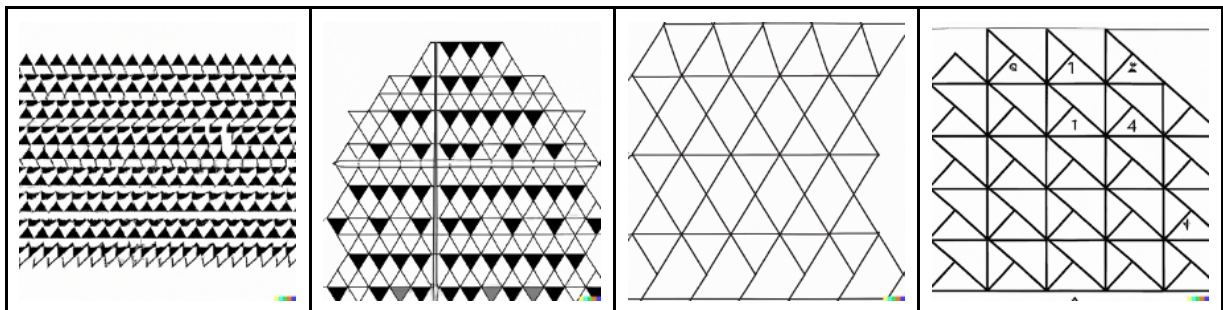


Figure 5: Four results of using DALL-E with prompt 2

The program thus generates a better matrix by additionally specifying that the first triangles point downwards and the rotation can be to the right or to the left. The first result is somewhat better, but far from satisfactory. The other results move away from the target again and are partly very questionable. In the last image, for example, numbers and letters appear. Maybe we have to tell the AI to stay as close as possible to our description.

4. Conclusions

Overall our experiences show that AI holds great promise but, in its current state, can also be the source of great frustration. At a time where AI as an end-user technology is exploding it will become nearly impossible for users to track the rampant evolution of AI tools. It will take AI tools to be able to use AI tools. Already extensions to AI tools enable the proliferation of next-generation meta-AI tools bundling up several AI tools into new services. The mere speed of this process is raising big concerns on multiple levels. Education will be profoundly transformed in good and bad ways. In this kind of world with so many quickly moving parts the idea that “Proompting is Computational Thinking” may provide some calm by offering some resilient principles. It will be necessary to establish common principles to all these systems. Here are some early principles that we have noticed:

1. ***Proompting is Computational Thinking*** is a universal framework offering some resilient principles that are valid across different AI tools.
2. ***Proompting is a social process***: The opaqueness of most AI tools makes it difficult to predict the output resulting from a prompt. The community of users showcasing examples and sharing suggestions is essential to work productively with AI tools. Some AI systems such as Midjourney include this kind of social embedding.
3. ***The unpredictable nature of prompting may require reverse Computational Thinking***. Some AI tools offer early support to reverse the Computational Thinking process. For instance, users can provide existing images to make Midjourney describe that image as a prompt. In other words, these tools make the proompting processes more transparent by providing a path back from Analysis (step 3) to Automation (step 2).
4. ***Competence models need to shift from answering questions to posing them***. Traditionally the competency of individuals is judged by their abilities to answer questions. For instance, can you label the various parts of the human body in pictures of anatomy? Emerging competency models will be about the skills to pose questions and to modify them. In other words, new models will be about proompting.

5. References

- [1] Bell, T., F. Rosamond, and N. Casey, "Computer science unplugged and related projects in math and computer science popularization," in *The Multivariate Algorithmic Revolution and Beyond*, L. B. Hans, D. Rod, V. F. Fedor, and M. niel, Eds., ed: Springer-Verlag, 2012, pp. 398-456.
- [2] DiSessa, A. A., *Changing minds: Computers, learning, and literacy*: Mit Press, 2000.
- [3] Hsu, F.-h., "IBM's deep blue chess grandmaster chips," *IEEE micro*, vol. 19, pp. 70-81, 1999.
- [4] Kay, A. C., "A personal computer for children of all ages," in *Proceedings of the ACM National Conference*, 1972.
- [5] Kistermann, F. W., "Blaise Pascal's adding machine: new findings and conclusions," *IEEE Annals of the History of Computing*, vol. 20, pp. 69-76, 1998.
- [6] Kurzweil, R., *The singularity is near*: Springer, 2014.
- [7] Repenning, A. and A. Basawapatna, "Explicative programming," *Communications of the ACM*, vol. 64, pp. 30-33, 2021.
- [8] Repenning, A., A. Basawapatna, and N. Escherle, "Computational Thinking Tools," presented at the *IEEE Symposium on Visual Languages and Human-Centric Computing*, Cambridge, UK, 2016.
- [9] Rushkoff, D., *Program or be programmed: Ten commands for a digital age*: Or Books, 2010.
- [10] Saqr, M., K. Ng, S. S. Oyelere, and M. Tedre, "People, Ideas, Milestones: A Scientometric Study of Computational Thinking," *ACM Transactions on Computing Education (TOCE)*, vol. 21, pp. 1-17, 2021.
- [11] Wang, F.-Y., J. J. Zhang, X. Zheng, X. Wang, Y. Yuan, X. Dai, J. Zhang, and L. Yang, "Where does AlphaGo go: From church-turing thesis to AlphaGo thesis and beyond," *IEEE/CAA Journal of Automatica Sinica*, vol. 3, pp. 113-120, 2016.