

Caring with Computers: a conversation between Computational Mama and Diwas Raja Kc

Computational Mama and Diwas Raja Kc

Abstract

Computational Mama met with Diwas Raja Kc for an online conversation about her practice in relation to artist-led public pedagogies. Computational Mama's work explores live coding and computational thinking as forms of friendship and care. She currently lives in Udaipur, India, in a multi-generational household of all women and her 5-year-old son. In this rich exchange ranging from feminist theorists, artists, and coders, they discuss how she chose this moniker as care-giving and motherhood increasingly impact her practice and strategies of creative facilitation with community.

Keywords

caregiving, creative coding, feminist strategies, social practice

Diwas Raja Kc: *The thing that draws me the most to your work is the way you dissolve the dissonances between care—as a feminist politics of building relations—and knowledge. There is, firstly, the vast legacy of European thought that has trained most of us to bifurcate the world into subjects and objects and treat knowledge as a practice fundamentally divorced from affections. Then there are also ordinary allegories of the ‘heart’ and the ‘mind’ that partition caring and knowing as belonging to separate (and gendered) domains of activities. In your work, I see you insisting instead that knowledge, thought, and learning can only be thought of as relational processes of care-giving and care-receiving. Could you tell us a little bit about why you care about care and how you have come to practise your art as a pedagogical exercise in fostering care?*

Computational Mama: If ‘knowing’ is an act of opening ourselves up to thoughts, processes, and practices, then to me it seems intrinsically tied to caring. Is it possible to embrace and be open without giving ourselves that space for care? Not only does knowing depend on the care work done for our physical and emotional well-being, but it is also inseparable from the interdependencies that, for instance, the authors of *The Care Manifesto: The Politics of Interdependence* imply when they define care as the social capacity involving “the nurturing of all that is necessary for the welfare and flourishing of life” (Chatzidakis, 2020, p. 9).

A censure on conversations about care in my professional life was something that started to emerge as a sore spot for me as a creator. I was working as an exhibition designer and project manager in the museum industry, and I was expected to only speak of my work and disengage from my context as a carer. As someone entangled in deep relations of care, this separation was untenable for me.

My project *Recreating with p5js* has been a touchstone for me in developing my own practice as a learning format premised on care. *p5js* is a coding framework created especially for artists and creators to extend their visual and sound-based practices with beginner-level coding. The intention of the project was to create an access point for womxn creators to discover how creative coding can be exciting. It was imperative that the content of *p5js* spoke to a group of womxn learning to code together.

I was drawn to trying out a format called “Recreating the Past” developed by the artist Zachary Lieberman (Carvalho, R., Lima, C. & Barreto, S., n.d.). This course used the work of earlier computational artists and allowed first-time coders to recreate them using contemporary coding techniques. I wanted to adapt this to explore the works of South Asian women artists, and I very quickly realised that artist Nasreen Mohamedi’s work—much of which uses geometric forms, mathematical progressions, and principles of precision—mapped very well onto the basics of learning *p5js*. Thanks to the extensive documentation of her work in the retrospective *Waiting as a Part of Intense Living* (Kumar, 2016; Cotter, 2016), I was able to include rich insights about her life and work in the learning module. This combination of learning the basics of coding with caring for South Asian women’s art history has really been a joyous space for building a community of womxn creators.

A large part of my practice is about sharing with others the creative computational concepts and processes I am learning myself. I do not hold a degree in a STEM field nor have I learnt to code formally. I learnt to code in between diaper changes and midnight feeds after my son was born. In a way, coding came into my life as a slow and unstructured practice of self-care in between work and

childcare. In the collective spaces that emerge through my projects, I find it difficult to think of myself as a ‘teacher,’ for it implies to me claims of imparting expert knowledge. Rather, I see myself as a facilitator and those joining the workshops as participants. As a facilitator, I create opportunities where I, along with others, can jointly participate in learning. And it is very important for all of us that the space that brings us together be defined by the care we show towards processes, friendships, collaborations, and community.

It is true that as a facilitator, I enter these spaces of learning with the mindset of providing care and the participants typically embrace the role of receiving care. In practice, however, this relation is not one-directional. Our roles as care-givers or care-receivers in a space of learning are always amorphous and porous. In my experience as a facilitator, learning and imparting knowledge are simultaneous activities and these require everyone to ‘listen,’ ‘understand,’ and ‘empathise.’ I think these important verbs also delineate the acts of care-giving. This understanding is not something that came to me naturally but has really come from the experience of being a primary carer.

DRK: *I can see from what you have said about your code-based practice and your position as a mother how your chosen moniker Computational Mama is apt for you. But there must still be a story behind how you arrived at it.*

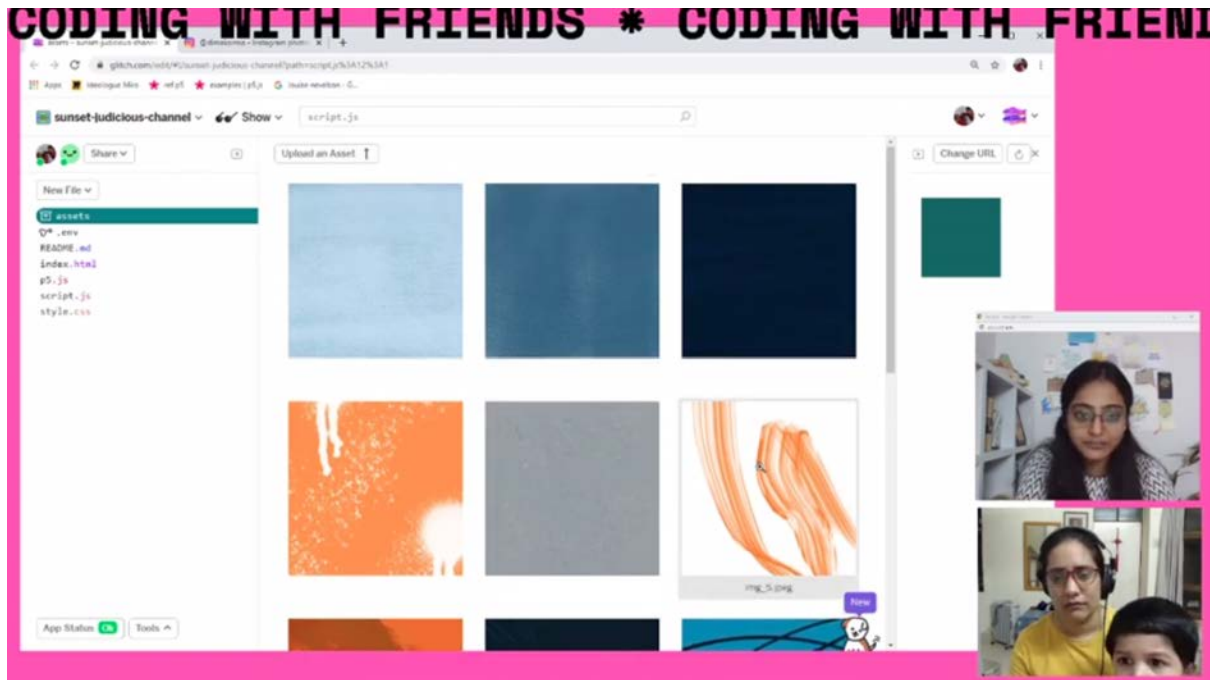
CM: I adopted Computational Mama as a moniker in the last trimester of my pregnancy in 2017 for an anonymous Instagram account where I posted all the coding sketches I was making. The moniker was a light-hearted attempt at manifesting something exciting for my new life as a mother. Computational Mama started as an alter ego that gave me space to meld my identities as a carer and a creator. But over time, as the work and the baby grew, the idea behind this moniker became more powerful. Care-giving and motherhood impact my practice as a creator even more deeply now. I am unable to see the time I dedicate to work, the focus I can ‘spare,’ and the conditions of my workspace as separate from my role as a mother.

More recently, Julie Phillips’s book *The Baby on the Fire Escape: Creativity, Motherhood, and the Mind-Baby Problem* (2022) has served as an excellent manual for me. The book looks at the lives of well-known artists and writers like Alice Neel, Doris Lessing, Ursula K. Le Guin, and Audre Lorde and meditates on the ways motherhood has affected their work and their lives. For Computational Mama, it has become important to assert vocally and unapologetically what Julie Phillips refers to in the book as “the blank spot on the map where mothering and creativity converge.” (2022, n.p.) There is no threshold between care and work. One accepts the presence of the child within the so-called ‘workspace’—a presence both in the physical sense as well as an absent figure that never leaves your mind. The child is a permanent resident in your working brain.

This is perhaps partly why in the workshops, live streams, and projects I run as Computational Mama, I cannot disentangle myself from the body, the feelings, and the sensations of care. I have to ensure that my fellow collaborators and participants accept, appreciate, and celebrate my contributions by understanding that motherhood brings a very direct value to learning and creative processes. To be content with incompleteness, to admit mistakes and messiness within the workflow, to allow interruptions and unproductivity: I espouse these as values and I allow for these to happen in the learning spaces I facilitate.

For a new project *Mother Machines*, I am working with AI to create several images of what a mother can be. It has been a process of imagining and then reading about mothers in workspaces or

creative spaces, questioning ideas of birthing, child-rearing, care, and so much more. Computational Mama started secretly and tentatively. Slowly, but surely, Computational Mama has allowed me to speak openly and confidently about care, work, and all the things I found hard to say before!



Screenshot from “Coding with Friends” livestream with guest Saumya Kharbanda on top and Computational Mama with son below. Photograph courtesy of Ambika Joshi.

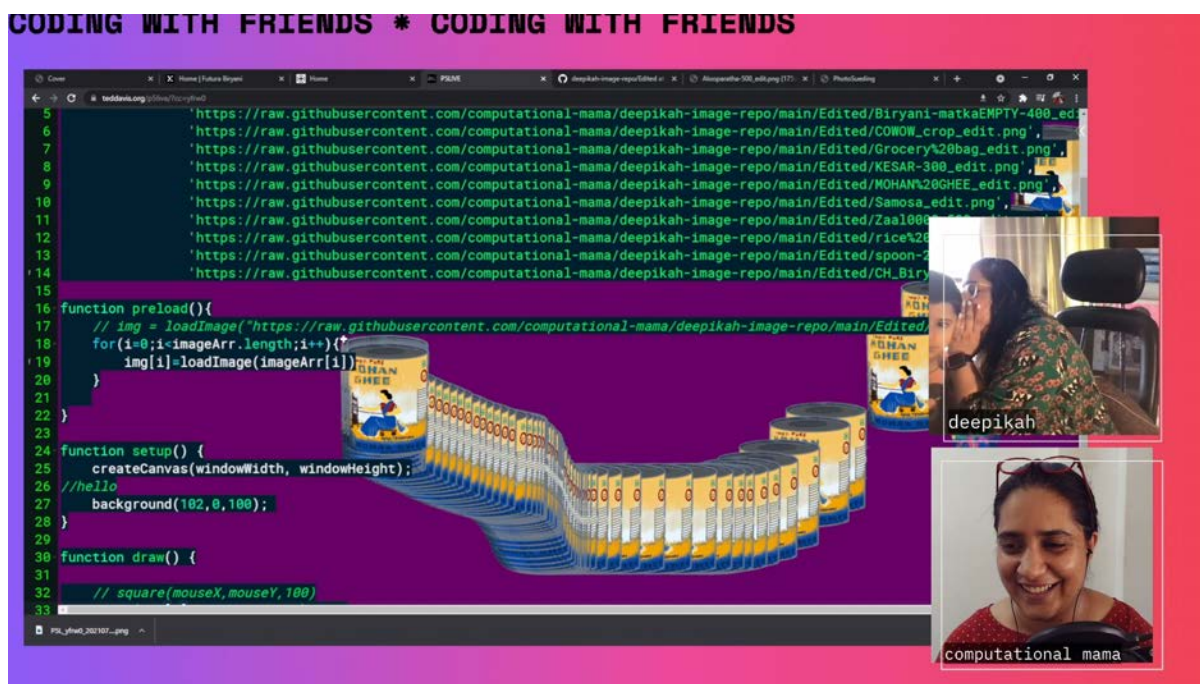
DRK: *Your definition of knowledge as an openness—an openness that is markedly tied to the giving and receiving of care—reminds me of Donna Haraway’s influential concept of “situated knowledge” (1988). One major takeaway from this concept has been the argument about positionality—that knowledge can only be gained from positional perspectives. This was an important point that was brought to light substantially by feminist interventions within STEM disciplines. But there is another point in Haraway that is emphasised less which involves this matter of openness and care. For her, what relations we open up to and how we enter these relations are key. Knowledge is thinking with and thinking through the worldly webs of relation one is entangled in and one cares for. As a corollary, practising caring relations is not a generalisable mode. One cannot practise care with everything and everyone; one does so only through an everyday commitment to a particular set of possible and positioned relations. When feminists have talked about care, they point to forms of concrete attachments, material dependencies, and intimate connections as the devalued but indispensable condition of our survival and sustenance. Knowledge, then, means whatever contributes to the flourishing of specific interconnections and communities. I can recognise that this particular heritage of feminist thought has impacted you.*

In addition, I see you grappling with the question of knowledge politics also by means of reclaiming pasts, appropriating art made by women artists, and compelling coding into doing feminist memory work. There is something of a genealogical orientation that appears to be a persistent presence in your thinking. By this I mean not necessarily a quest for origins or a desire for a canon, but a way of settling

on a usable past filtered through the sediments of history. This is related to the broader “archival fever” of feminism: the manners feminists pursue possibilities for the future by opening up alternative ways of being in history. Given how these myriad threads of feminist knowledge politics are current in your work, can you reflect on how you relate to them?

CM: This question really helps me to look back at my past and remember how I arrived here. My mother is a designer in the social sector, and she was influenced by prominent feminist figures like Kamla Bhasin, Abha Bhaiya, Ginny Shrivastava, and many others. Her work focuses on menstrual and reproductive health and is deeply rooted in community work around gender and sexuality. As a child, I often travelled with my mother for her various training sessions in rural parts of Southern Rajasthan. I grew up seeing rooms full of women talking, laughing, learning, being, and working together. Working together in these spaces never looked anything like what we mean when we say ‘production’. To the young and impressionable me, this was the image of joyous coming together that became associated with care, camaraderie, and community.

As I think back, I realise this is what I was longing for. This is why I craved connections with womxn creators. My other project *Coding with Friends* is so closely related to this image. *Coding with Friends* is an ongoing series of live streams where I casually invite old and new friends to code together over a simple intimate chat. Coding turns into something like making dimsums or folding paper cranes or playing a boardgame. As the livestream unfolds, a small approachable project generally related to the field of work of the invited guest is live-coded and developed together. The coding session is accompanied by meandering conversations with no specific agenda, as those that happen between friends. *Coding with Friends* seeks to destabilise the output-oriented legacies of technology, art, and coding. It hopes that the acts of friends co-creating will interrupt the logics of commodity, production, and performance.



Screenshot from “Coding with Friends” livestream with creator Deepikah. Photograph courtesy of Ambika Joshi.

All the while, *Coding with Friends* also casually and simply claims space for womxn creators and turns coding into a form of community-building. In doing so, we have challenged computational notions about what code can produce, who can produce code, and where it can be produced. This project has also allowed me to spend a lot of time speaking to womxn creators and to understand their positions and perspectives, and one thing I have been able to recognise through this experience is how reflexively womxn creators build networks of care around themselves. It continues to surprise me how quick we are to acknowledge produced works and individual achievements and yet discount, even dismiss, all the structures of care and support that go into accomplishing any task. I get so annoyed now by all the images of the individual artist or of the individual thinker, working in their studios or sitting in front of their monitors—alone, isolated, and immersed—as if creativity without isolation is unimaginable.

In order to undermine the patriarchal structures of coding, my gesture has been to shift the paradigm from STEM to Art and then from male artists to female artists. And I suspect this has something to do with what you called my genealogical orientation. In fact, my first project *Recreating with p5js* came about in the context of reading about past womxn artists. I draw a lot of inspiration, as well as joy, from learning about the many ways womxn artists have created and held their ground. It affected me to learn, for instance, how intensely the artist Nasreen Mohamedi practised care towards everything in her life after finding out about the loss of her motor skills and impending death due to a genetic disorder. Her students repeatedly emphasise how she taught them not only art but also how to care. Although she is remembered for her austere geometric abstractions, I experience her work through her rich thinking about care. It has become very important for me to know the artist outside the “studio” and through their interactions with the world around them and through what they cared for.

Similarly, when I first started to read about artist and printmaker Zarina, who is well-known for her geometric forms, I could draw parallels to my own life. In her interviews, Zarina has spoken about writing letters to her sister where she would evocatively describe the home that no longer existed for her. Zarina has a haunting effect on my own sense of displacement, grief, and loss. From my research into her life, I learnt that Zarina was the first woman in her family to leave the home and the protection of the male patriarch. I find myself getting hooked to these tiny details from the lives of past artists, because they help me observe their struggles in managing life and work under patriarchal conditions. And that, in turn, emboldens me to think of myself in the world and of the world through care. I have tried to seep my coding workshops into these stories and observe what happens when these feminist pasts permeate through the structures of code production.

DRK: *Your thinking about care in the context of coding and computation as well as your regard for past feminist experiences are also interesting when you consider how computing, at the crucial moment of its emergence, was an overwhelmingly feminised workforce. We have begun to see some retelling of this obscured history in popular culture now. But more than the retrieval and celebration of this past, it is fascinating to think of computation as something constructed on the notion of “feminine labour”. You alluded to one aspect of this labour when you spoke of making dimsums and folding cranes. They are not unlike weaving, knitting, crocheting, and other so-called feminine crafts that are based on making patterns and diagrammatic abstractions through mundane, routine, and repetitive action. Algorithms can very well be traced to these feminine crafts. The other aspect of this labour is, of course, care. It is very telling indeed that computing and programming were part and parcel of feminised*

labour, which has revolved around the provision of care, comfort, courtesy, service, and servitude. As Wendy Hui Kyong Chun (2011) has argued, the automated, command-and-control structure of computer programming, which is designed to keep key operations of the machine invisible, directly encapsulates the gendered history of labour that relies on the obfuscation and erasure of women's work (of care). I admire how your practice of coding as care makes it possible to pay attention to the computer's hidden history of labour. Since we are also talking about genealogy and alternative ways of producing usable knowledge, I was wondering if you could reflect on how these invisible histories have surfaced for you and your participants.

CM: Yes, the source word for computers is the women workforce whose mental and physical labour of computing had played a key role during the Second World War efforts, including for NASA's aerospace ventures. Portrayals of these in series like *The Bletchley Circle* (2012) and the popular movie *Hidden Figures* (2016) are not ideal representations, but they have nevertheless been important for me in discovering this connection. Over time, as I talked to people around me about my work, I also learnt that some of my friends and acquaintances had mothers who were mathematicians and had jobs in the 1980s working with algorithms. Among the stories that were relayed to me, almost all of them had ended up quitting their jobs for family and care-related obligations. Knowing this gave me the confidence to bring these narratives into the workshops. The workshops could be a place where womxn participants could be coding and at the same time be reflecting on these histories of computation.

Among the many things I was imbibing during the early days of the COVID-19 pandemic was the wonderful talk *Programming is Forgetting: Toward a New Hacker Ethic* (2016) by programmer and poet Allison Parrish, one of my favourite creators. She examines some points Steven Levy made in his seminal book *Hackers: Heroes of the Computer Revolution* (1984). She criticises Levy's guidelines on "hacker ethic" for ignoring the biases and exclusions that underwrite the hacker's ethos and proposes instead what she calls "Hacker Questions" that offers not a new set of rules for hackers but a set of questions that compel hackers to recognise their own assumptions and points of views. This perspective Parrish offered was the shift I needed in my practice. "Hacker Questions" gave me the tools to have my corner of coding as a form of care and invite womxn to coding in the ways I felt would work best. It also allowed me to work unapologetically from a space of motherhood, sharing honestly all the good and the bad that comes with it.

Hacker ~~Ethic~~ Questions

- ~~Access to computers should be unlimited and total.~~ Who gets to use what I make? Who am I leaving out? How does what I make facilitate or hinder access?
- ~~All information should be free.~~ What data am I using? Whose labor produced it and what biases and assumptions are built into it? Why choose this particular phenomenon for digitization/transcription? What do the data leave out?
- ~~Mistrust authority—promote decentralization.~~ What systems of authority am I enacting through what I make? What systems of support do I rely on? How does what I make support other people?
- ~~Hackers should be judged by their hacking, not bogus criteria such as degrees, age, race or position.~~ What kind of community am I assuming? What community do I invite through what I make? How are my own personal values reflected in what I make?

“Hacker Questions” by Allison Parrish from *Programming Is Forgetting* (2016) Image courtesy of Ambika Joshi..

I would connect the question you ask about computation’s hidden history of gender directly to Parrish’s proposal for revising programming through a series of ethical questions. Parrish’s (2016) simple methodology is also what inspired me to use terms like “making dimsums” or “folding cranes” in the context of coding. For me, this was an opportunity to break the notion of software and computation as labour. I say this not to deny or dismiss the value of all the activities that are undertaken as ‘labour’, but I do feel that the idea of labour is too tied to an output-oriented legacy. I wish to partake instead in arguing that there are other ways to organise our productive activities and other ways of being creative outside the concept of work. I speak of coding (especially creative coding) in terms of how it matters to me and the shared joy it brings, not in terms of flashy outputs or over-produced algorithmic art. Through the livestreams on coding and subsequent *Coding with Friends* series, I wanted co-creators and online viewers to see coding as a process. If I make coding sound like a feminine craft, as you say, it is first because this comparison makes coding seem familiar and welcoming and second because this allows the focus to be on the task of producing togetherness and community and on how learning can be predicated on a very different premise of meaning-making and memory-keeping.

DRK: *I want to hear more from you about coding as a creative practice. You seem to highlight the liberatory potential of coding, having yourself experienced the instructional, automisable, and*

recursive mechanisms of algorithms to be amenable to the demands of care and creativity. My own tendency has been to resist the algorithmic turn, unable to shake off thinking of it as essentially a means of value extraction. To me, this is particularly evident in the ways data-mining tech giants collectivise and leverage the affective confessions of our friendships, relations, positions, preferences, and opinions on digital platforms—even as the neoliberal rationality that drives these technologies ferociously seeks to delegitimise established forms of collectivities and eradicate the very notion of the public good. The seductions of algorithms with their problem-solving functions and the ability to turn everything into protocol have been at the centre of this, and these are aided by the ease with which we metaphorise “networks” to mean something like a community. So central are algorithms to the future of capitalism today, that I must ask you: is not imagining our liberations through coding equivalent to that famous maxim gifted to us by Audre Lorde concerning the “master’s tools” (1984, 2018)?

Of course, Audre Lorde may not have meant what we think she means. Just as many do also believe that the tools of producing and managing data can be reclaimed for the public interest. There are also artists like Harun Farocki, Hito Steyerl, and Forensic Architecture who have emphasised the need to learn, think, and see like the machine. For them, counterintuitively, seeing like the machine is the hope that we—the humans, or rather the oxymoronic post-human humans—will escape the machinic takeover of everyday life and preserve contingency, play, and freedom in the world. Alexander R. Galloway’s recent book ‘Uncomputable: Play and Politics in the Long Digital Age’ (2021) also comes to mind, where he constructs a counterhistory of the computer such that a range of human experiences from intuition, affect, embodiment, to care come to constitute a domain beyond computability.

Above all, it is art as the final reservoir of play and creativity that is held up as uncomputable. You will know for sure how powerful this idea has been from the resistance computational art has faced in being accepted as a legitimate form of art. The cutting edge of AI research also tends to view ‘artistic behaviour’ as the next major frontier and expects algorithmic and computational creativity to pave the way forward. I agree with you that the ways we have conjured and relied on conceptions of creativity and human genius deserve to be dismantled. But do you worry that the means of dismantling these concepts are the same ones that disarm us in our struggles against neoliberalism in its newer incarnations through data?

CM: Honestly, I do not know how much I can answer this question on the ‘larger evil’ of the tech giants. I do understand the dangers. My stance is that how can we even know what the master’s tools are if we do not have the language to use or understand them. As I said above too, my practice of coding is premised on not letting coding remain the preserve of the mainstream or of cishet men. If we think of algorithmic spaces as inaccessible or irrelevant to our struggles, I fear we will keep losing. Creative coding is a simple but powerful way to start understanding what is under the hood. We may not be able to beat those with the powers at their own game, but that is not to say that we cannot build our own web spaces and softwares to reach where we need to in order to fight. To me, this is not using the master’s tools.

One of the most popular creative coding tools called Processing was built in the early 2000s expressly to make coding more accessible to creators and designers, and it has been a testament to the strength of the loving, encouraging, and diverse community that uses it. Its most popular web framework *p5js* was conceptualised and built by the artist Lauren Lee McCarthy. *P5js* has become an important tool in many interesting political, social, and journalistic projects. The Processing Foundation is also at the forefront of making the black box world of AI accessible to artists and

creators through projects like ml5js. These technologies only need a simple, low-end computer and limited internet access; the basics are easy enough for anyone over the age of 12 to pick up. And the communities that form around them are so far-reaching.

Certainly, those who can access critical literature and texts of the world on the wrongdoings of big tech could arm themselves with a little bit of creative coding. I am happy to teach them! Since we were discussing Donna Haraway earlier, how can we forget her advocacy for fostering ethical relations and kinship with machines among other non-humans, even when these alliances are uneasy? I am often amused by the naivety of the machine and have found myself occasionally comparing it to my child who was still developing motor skills, learning from experience, and learning to form opinions. Maybe the presence of my child beside the computer has helped me see the soft, silly part of the machine. When you see that, you realise your kinship to it and how it might benefit us to befriend the machine rather than pull the plug.

The objective of my workshops is not to make the participants into hacker-level coders. They are merely an introduction to a language. We are just building a fearless and forgiving friendship with the computer. The best testament to *Recreating with p5js* is that it is attended not just by artists but also by tech safety and privacy advocates, teachers of primary school students, urban planners, and architects. It is my small and grassroots effort to give people a joyfully accessible way to learn about technology and not fear it.

***DRK:** Finally, can you talk about how the projects you do as Computational Mama have informed other areas of your coding work where, I assume, you have more likely had to deal with the broader publics, popular culture, and the mainstream coding networks?*

CM: Through all parts of the social practice of creative coding, I find that three things ring large and true for me and I have sought to pass these on to my other mainstream projects with collaborators from various creative and technical fields. These are coding for value over solutions, coding with care and inclusion, and coding as a community.

I run a boutique creative technology studio called Ajaibghar with my friend Nanditi Khilnani through which we offer creative solutions driven by new media and technology for artists and projects in Arts & Culture. The studio has consciously moved away from providing ‘tech solutions’ and implemented a strategy of starting the conversation with the values the client or the commissioning team is seeking to extend through technology. Coding for value over solutions is a simple and direct response to the mainstream methodology of software to be a solution for everything. By this, I mean the tendency to solve the problem of ‘can’t cook’ with a ‘build delivery apps’ mentality. This rampant solutionism is a typical patriarchal output. It does not pay attention to real problems, it has no empathy, and it is tactless. It chants the mantra of ‘technological innovation’ and wants everyone to believe that there is nothing greater than this mandate.

Care, both in terms of caring about your work as well as caring for your collaborators and their needs (including outside of work), is a key alternative that I believe can be transferred to platforms involving the broader publics. I have noticed through my workshops and learning sessions with my cohorts how valuable the assignments become when we work with the intent of caring for each other and when the process is indissociable from our friendships. It is something that I believe will be very meaningful to a lot of people. My hope is for coding to be an even more open and participatory space where diverse people can come together to create and belong. I imagine the coding community as an

overlay of networked friends and collaborators who share ideas, exchange feedback, and fuel the creative urges of every member.

Biography

Computational Mama's work explores live coding and teaching as a form of camaraderie, friendship and self-care. On her Twitch stream, she teaches the basics of creative computation and new approaches to computational thinking. Her streaming series *Coding with Friends* is rooted in creating inclusive spaces for womxn makers to explore creative computation. The moniker Computational Mama, was created in late 2017 when a bed-ridden heavily pregnant mama-to-be decided to learn creative coding. She currently lives in Udaipur, India, in a mutli-generational household of all women and her 5 year old son. She is also known as Ambika, a museum professional with over 10 years of experience and is the co-founder of Ajaibghar Cultural Services.

References

- Carvalho, R., Lima, C., & Barreto, S. (2022) Digital Media and Sustainable Development Goals Breathe New Life Into the Artworks From the Soares Dos Reis National Museum.
- Chun, W.H.K. (2011) *Programmed visions: Software and memory*. MIT Press.
- Collective, C., Chatzidakis, A., Hakim, J., Littler, J., Rottenberg, C. and Segal, L. (2020) *The care manifesto: The politics of interdependence*.
- Cotter, H. (2016) Globalism's New Spin. *The New York Times*, p. F4-L.
- Galloway, A. (2021) *Uncomputable: Play and Politics in the Long Digital Age*. Verso Books.
- Haraway, D. (1988) Situated Knowledges: The science question in feminism and the privilege of partial perspective. *Feminist studies*, 14(3), p. 575-599.
- Kumar, B. (2016) The Elegant Complexity of Nasreen Mohamedi. *MARG-A MAGAZINE OF THE ARTS*, 68(1), p. 16-25.
- Levy, S. (1984) *Hackers: Heroes of the computer revolution* (Vol. 14). Garden City, NY: Anchor Press/Doubleday.
- Lorde, A. (2018) *The master's tools will never dismantle the master's house*. Penguin UK.
- Parrish, A. (2016) Programming is forgetting: Toward a new hacker ethic. *Open Transcripts*. epub.
- Phillips, J. (2022) *The Baby on the Fire Escape: Creativity, Motherhood, and the Mind-Baby Problem*, Norton, NYC.
- Zarina, (no date) artist website accessed 1 May 2023, www.zarina.work.