

7

In the Mood
for Love
(In and Out
of the Code
Worlds)

Héctor Beltrán

I'm not sure if I've ever been on a date with someone you would consider a hacker. But I am sure that I've asked a lot of hackers about their love lives. Or, rather, I've asked a lot of hackers about their lives away from the computer. I've thus heard a range of stories about how they approach love, relationships, and dating. This started, because as I moved across different spaces with hacker-entrepreneurs, from Mexico to the US and back, from hackathons to their university spaces to their hacker school, I wanted to know how much of their hacker selves carried over to everyday spaces.

In her book, *Liquidated: An Ethnography of Wall Street*, Karen Ho studies the construction of financial markets and bursting of market bubbles. She focuses on the daily practices, the values and dispositions that come to define Wall Street's culture. She argues that investment bankers see themselves as incarnations of the market—they become one with its rhythms and movements.

I wondered, were there similar dynamics at play for code workers? Did they become one with logics that drive the code worlds, with the rhythms and movements from the worlds of computing?

My investigation was inspired by previous researchers of software development who understand that what's "in the code" is always influenced by more than the code itself. In *Software Studies: A Lexicon*, for example, Matthew Fuller convenes multidisciplinary scholars who develop a vocabulary for understanding how the speed and rationality of computation meet with its ostensible outside (users, culture, aesthetics). What ends up in the code proper, as well as the very style and approach one brings to coding, is influenced by the dispositions carefully cultivated by conditions around us, and vice versa. In her longitudinal and multi-sited ethnography *Coding Freedom: The Ethics and Aesthetics of Hacking*, Biella Coleman notes that because the technical craft of coding requires a constant awareness and rearrangement of form, software programmers develop competence in transferring mental dispositions into other arenas of life.

While much of ethnographic fieldwork revolves around conducting participant observation, I also wanted to ask specific questions that would provoke reflection from research participants themselves about how they saw particular subjectivities transferring from their coding lives to their everyday lives.

After all, my chosen research group was not the typical subaltern group, waiting to have their voices amplified, but many times aligned more with the "superalterns" other anthropologists have identified, ready and willing to provide their opinions and assessments. Many of the young people I conducted research with over the years were happy to narrate their own practices as they figured things out, as they moved between the

majority and minority world to understand how things work. They attempted to mobilize the “problem-solving mentality” inculcated into students by university computer science and engineering programs. So I began asking them: “What other aspects of your life do you feel are most influenced by your identity/role as a software developer? Can you think of examples that illustrate this?”

My interlocutors gave responses about how they build communities, how they feel about belonging to different collectives, and how they solve problems across various domains of their lives. They talked about the requirements of writing good algorithms (speed and efficiency) as well as how their flexibility on the labor market (contracts and ephemerality) end up creeping into their social relationships. Some of their responses can only be properly filed under “when hacking goes wrong.” The following stories highlight some of their representative responses and reveal how practices from the code worlds infiltrate their quests for belonging, intimacy, and even love.

One thread of answers fell along the line of “no separation.” These developers wanted me to understand that there was no distinction between their roles as software developers and other aspects of their lives. They carried their identities with them all the time even as they separated themselves physically from their computers. Cofi, for example, told me, “It’s not like I can suddenly turn it off or on. I’m always a coder and I’m always thinking like a coder... When I shower, when I eat, even when I sleep. I dream of code.” Cofi wanted me to understand that his coder self was never compromised and that it wasn’t something that he could simply leave aside.

Another line of responses revolved around the way coders approached specific everyday activities such as cooking and gardening as well as recreational social practices such as camping. They highlighted how they brought the same organizational and planning approaches to these domains as to their code. TecnoChica, or T.C., explained to me that before they even start cooking, they know exactly what they are going to do. They have the recipe and a plan, they know where the ingredients are, and they even know how long it is going to take. This was quite different from their partner’s cooking style, who gets into the kitchen without a plan necessarily, and frequently has to run out (or send T.C.) to the corner store mid-preparation because they are missing an ingredient. These self-reflections were based on comparisons with their partner’s approach to activities, as T.C. elaborated in another example: “El otro día mi pareja se sintió con ansias. Esto,

porque el jardín necesitaba una manita para mejorarlo y pensaba que nunca lo haría y que lo pospondría por mucho tiempo. Pero en realidad yo lo estuve pensando en mi cabeza durante días y lo tenía todo visualizado. Después sucedió que hice todo como en dos horas, rápido y eficiente. Franz cuando vio, se ha quedado super impresionada.” (The other day my partner was feeling anxious. The garden needed a little hand and they thought I was never going to do it and that I would put it off for a long time. But I was actually thinking about what I was going to do for days. It was all in my head. Then one day I did it all in like 2 hours, fast and efficient. Franz saw and was super impressed.) *Rápido y eficiente*, “fast and efficient” were the words T.C. used to describe their garden work. Their approach to everyday activities was carefully planned and streamlined, always with an eye toward saving labor. Indeed, these qualities are representative of good code and a good coder: well organized, efficient, and against unnecessary lag or repetition.

Lotar added another activity to this thread of examples when he mentioned camping during a recent trip to California. “Creo que es porque nunca he acampado pero investigué un montón antes acerca de esta actividad. Me gusta tener la mayor cantidad de información posible para el viaje. Así es como le entro a cualquier nuevo proyecto. Hago un chingo de investigación y me gusta tener opciones y herramientas disponibles antes de entrarle a cualquier cosa. Para cuando llegue el momento hacerlo de la manera más eficiente posible.” (I think it’s because I had never been camping before, but I did a lot of research about the activity. I wanted to have the most possible information for the trip. That’s how I approach any new project. I do a shit load of research and I like to have options and the best possible tools available before I jump into anything new, so that when the time comes I can do it as efficiently as possible). By including the keyword “eficiente” in his response, Lotar confirms the connection to self-learning and careful preparation that many coders see themselves possessing and bringing into their social lives.

Most of the responses from my research participants showed that they saw these crossings were positive. They agreed that an approach of self-cultivation, having the best tools at hand, and careful planning and efficiency could be successfully applied to other domains of life, and in fact, the world might be better if we all did. If one could hack away in the code worlds, why not hack the world around us?

But what about the flip side? What happens when hacking the world goes wrong? If my research participants brought their hacking sensibilities from the code worlds to other domains of life, then unsurprisingly they also brought them to their love lives. One respondent who had substantial experience with attempting to make the laws of the code

worlds function in the love worlds was Hiro. Hiro identified as more of a hardware hacker, adept at navigating the different layers of the computing stack; he was having luck in the labor market, as companies would hire him to create early prototypes of their products.

One set of stories Hiro told me were about how he attempted to hack the dating scene during a period when he was in San Francisco working for a company on the next generation of wireless chargers. He was being paid well but in San Francisco it was just enough to pay rent and to eat. Dating, he reminded me, was expensive! Even if both parties agreed to pay for their own meals and entertainment, dating still required that Hiro spent more than he would have budgeted in order to go out. So he came up with a hack. He bought himself an annual membership to the San Francisco Exploratorium. For a \$60 fee, he got unlimited visits and the annual membership allowed him to bring a guest for free. With the regular entrance fee of \$10, he only had to go on more than 3 dates to make it a good value. But the membership benefits also included two free drinks. “Because in this country they are so liberal,” he explained to me, “my dates end up offering to pay for both drinks, since I paid for the entrance. I go get the drinks, they don’t see that they were actually included, and they end up Venmo-ing for the drinks. I’ve actually made money off of the membership!” Hiro recounted his dating hack with much enthusiasm, letting me know that this actually encouraged him to go on dates every week.

But sometimes “hacking the dating scene” went wrong. Hiro went on dates frequently to make his annual membership a good value. As he executed his subroutine for dating, if you will, there were times when two series of dates might overlap, meaning there were various (potential) partners he might confuse. So Hiro made it a case to find partners with the same name. He told me the story of the two Marias. He was having coffee with Maria No. 1 when Maria No. 2 texted him, looking to make plans for later that evening. Hiro stepped out of the coffee shop, and texted Maria that he would call her later to make plans since he was currently on a business lunch. Of course, in his hurry, Hiro accidentally texted Maria No. 1, still inside the coffee shop. As he walked back in, she greeted him with a perplexed look, and asked “Why did you just text me this?” as she held up her phone in dismay. Maria No. 1 didn’t finish her \$7 iced coffee mint mojito. Hiro asked for his to go. It was their last date.

Hiro explained to me that it was a moment when “adding redundancy to the system had gone wrong.” While he was being facetious, and I think that dating the two Marias simultaneously was more coincidence than strategic maneuvering, the fact that he gave this story as part of his response to my prompt and that he referenced redundancy was quite revealing. In computer systems design, integrating redundancy can mean duplicating

critical components or functions of a system in case one fails to make the overall system more reliable. In this case, dating two persons at the same time provided Hiro with some sort of security that at least one would maybe work out. That they both had the same name seemed to provide security that he wouldn't screw things up by mixing up their names. It didn't quite work out that way, and for those wondering—it also didn't work out with Maria No. 2. Neither Maria No. 1 nor Maria No. 2 were available for comment.

Returning to his Exploratorium hack, I asked Hiro if he didn't get bored by going to the same place so many times. "It was boring since the first time I went," he responded. "The less physics and math you know, the more magical that place seems." Apparently Hiro wasn't enthralled by the hundreds of exhibits inside the galleries that promised to provoke "joyful exploration" or to allow guests' "curiosity to roam free." Hiro seemed to bring his "how things work" and systematic approach to explaining the behind the scenes "magic" that the Exploratorium sold. Some of Hiro's dates must have also felt his boredom or weren't excited by his approach, which might explain why some of them "ghosted" him during the date. "One told me she was going to the bathroom and never came back," Hiro laughed. Fortunately for him, this disillusioned date still Venmo-ed him for the drinks.

Having dates go wrong might not be particular to participants of the code worlds. Misadventures and faux pas in the dating world are the stuff that makes television series and interesting stories to share with friends across many social circles. But this lack of "magic" in dates and romantic relationships crept up in interesting ways in my respondents' stories. Rodo's response to my prompt, about how his life as a software developer influenced other aspects of his social life, started with, "Well my girlfriend once told me she didn't want MIT in bed with us." Rodo was a student in the MIT summer coding bootcamp I helped teach at UNAM in the summer of 2014. While I initially visualized his response quite literally, that Rodo was perhaps so attached to his laptop that he wanted to keep working on the coding assignments from the MIT course even while he was in his bed, his story was more complicated.

For Rodo's girlfriend, Estefy, having MIT in bed with them referenced the way in which Rodo was approaching their sexual life. Rodo continued to reflect: "It wasn't because I'm always thinking about the code. She was really annoyed with my *mala chamba* [doing a bad job], with the way I was translating the fastness and efficiency of a good algorithm into my performance in bed." Rodo started to understand that Estefy wanted him to slow down, take his time, and concentrate on the task at hand. Rodo had been discussing the MIT course with Estefy, a visual arts student, throughout the summer. He had mentioned our lessons on the core principles of writing good algorithms—their simple elegance

defined by their speed and efficient execution. So Estefy was associating this approach to their sex life, to which she felt Rodo was bringing an unwelcome “MIT approach.” On the one hand, Rodo told me that sometimes Estefy would get upset because he seemed to be somewhere else during their romantic couplings. He admitted that he was many times indeed thinking about how to better his code. Estefy was perhaps yearning for a sexual experience that wasn’t defined by the tempo dictated by the world of efficient coding, with MIT representing the apex of these practices. That she didn’t want MIT in bed with them seemed to be a warning to Rodo to leave the hacking in the code worlds, or at least at the door before he entered the bedroom.

Another of my respondents, Cofi, told me that his partner had even named his laptop to express her disapproval of his lack of attention to her. His partner would say things like, “Are you going to be able to have dinner with me or do you have plans with Mildred?” Mildred was not a name he used but one that his partner had assigned to his laptop, fully personifying the machine as an entity that had an important enough presence to intervene in their plans. The code worlds, the named machine that carried them, and the institutional leviathan that represented them had come to interfere in my respondents’ quest for love and romance in ways that they had not expected.

To be fair to my respondents, I have to mention that some did listen to their partner’s complaints and that there were attempts to rectify their shortcomings. On one of my return trips to Mexico City, about five years after my initial fieldwork and interviews, I visited with Rodo and I was happy to hear that he was still with Estefy. He was quite excited to tell me about one of his recent attempts to limit the degree to which he transferred the logic of the code worlds to the magic of the bedroom. He had signed up for an “alternative masculinities” Zoom workshop. These types of extracurricular and self-help workshops had proliferated and become increasingly popular during the 2020 pandemic with experts from diverse domains increasing access to their expertise and services using social media and Zoom.

Rodo hyped me up on the course telling me that it was not only super informative but that for him it had taught him how to be a better lover. Intrigued, I signed up for the Zoom course and connected with the instructor, a self-proclaimed alternative masculinities guru, as well as with Spanish speakers across Latin America for two days. As the guru led us on a journey to unpack masculinity, we learned about the myths of penis size, the importance of taking focus away from penetration in order to fully enjoy sex, self-exploration and practicing orgasm control with masturbation to concentrate on pleasuring your partner. We practiced “being there” by closing our eyes and eating

our favorite fruit slowly. I chose a mango, especially because I had privileged access to fresher, juicier ones in Mexico than I did in the US. Eventually we introduced ourselves, our names, professions/backgrounds, and what had inspired us to take the course. I was somewhat disillusioned when I didn't hear that any of the other participants were software developers. In a perfect conclusion to this story, I would have told you that there were many engineers and computer scientists working collaboratively on deconstructing their masculinist and technical approaches to the world. The truth is that the majority of the participants shared similar profiles: artists, academics, nature-loving types; most had long hair and scruffy beards; many lit up spliffs as we started the course. I wondered—why weren't there more programmers here, and on the flip side, why weren't more of these eco-friendly, spliff-rolling men getting their hands dirty in the code worlds? What would it take for them to be as turned on by the code as Rodo and other code workers? Isn't acquiring the efficiency and technique of the code worlds also sexy? Why might the subjectivities of "pleasure" from the world of alternative masculinities appear to be so bounded off from the worlds of computing? Might the problem not be more about our inability to communicate across these bounded domains, or better yet, to exist in the boundaries between them in order to reimagine and reconstruct these very boundaries?

I was certainly happy that Rodo had taken the course and started to make these connections, however. Now every time I'm eating a mango, I think of these questions, of Rodo's stories, as well as the stories of Estefy, Cofi, T.C., Lotar, and Hiro as they searched for love, in and out of the code worlds.

Héctor Beltrán is a sociocultural anthropologist who draws upon his background in computer science to understand how the technical aspects of computing intersect with issues of identity, race, ethnicity, class, and nation.

Acknowledgments

Thank you Borhane Blili-Hamelin, Itzamna Huerta, Dibyadyuti Roy, Ipshita Ghosh, and Rafael Poiate for listening to this story and providing thoughtful feedback and expertise.