On writing workshops for programming

Michael Nagle
mpnagle@gmail.com

Abstract

Over the past year, inspired by personal loss from COVID-19, I took a series of writing workshops in creative nonfiction, spanning nine months of time. In this paper, I am going to use those writing workshops as a jumping off point for a workshop of similar format focused on creative programming. The goal of this paper is to take insight from the form of writing workshops and use them in running a "writers workshop for programmers."

1. Introduction

At the beginning of the COVID-19 pandemic in April 2020 I lost my dad to COVID. Because of mandates early in the pandemic forbidding me to be in the hospital with him, I talked my dad through his death over Skype instead of by his side.

His death caused me to quit my job leading product research at Coda, move out of the Bay Area, and travel for the next eighteen months, living primarily in Portland, OR.

While there, I signed up on a whim in Fall 2021 for a weekend writing workshop on "Grief and the Lyric Essay." In the course I discovered Claudia Rankine's "Citizen," a lyric essay which explores race in modern America, from many angles: personal experiences, in the lives of sports celebrities, in the work of YouTubers, in intimate relationship.

It made me think: if Rankine can talk about her experience of race this vividly, then maybe I can talk about my own experience of grief.

2. The Writer's Workshop

2.1 Workshop Format

I proceeded to sign up for a 10 week writing workshop, on Personal Essay and Memoir, again based in Portland. This was my first time in writing workshop.

The format was simple: every week we'd read 2-3 pieces, each about 10-15 pages, and 2 or 3 participants would submit a piece of up to 5,000 words. This workshop had 12 participants and 1 facilitator.

The idea was to take inspiration from the weekly readings - tailored to the class and what people were submitting - and use those to learn about the craft of writing.

Getting peer feedback and faciliator feedback (live in group discussion and then outside of class in the form of a feedback letter from each participant) quickly gave a creative heatmap - what worked, what didn't, what needed more information, etc.

I found I began to be able to tell the specific grief story, and many more questions came up along the way (like: what made my father special to me, beyond the unusual circumstance of his death?)

I proceeded to sign up for a second 10-week workshop, and when that concluded, a third. These workshops were smaller: 9 participants and 1 facilitator, and instead of 30-40 pages of reading a week, it was much lighter on reading (2-10 pages a week,) and featured a weekly writing prompt of 2 pages. The core format of 2-3 chances to workshop a long (up to 5,000 words) piece stayed the same.

2.2 Workshop Impact

Finding great writing in the first class felt like finding a hidden store of intellect that had been waiting for me my whole life, out there in the world, a resource hiding in plain sight. I was palpably inspired by the great writing I hadn't known existed prior to the class.

Writing for an audience got my own work moving. I found it much easier to write into an audience, to know feedback was coming quickly.

Over the nine months, what seemed a huge task - writing 5,000 words and sharing them with people basically new to me - became a more familiar, or routine one. The challenge didn't fade, but the comfort grew, and I found myself taking more risks over those nine months.

The workshops also fit the rest of my life - they were a commitment of about 8-12h a week, unlike a fulltime MFA writing program.

2.3 Writer's workshops for programming

I have since found myself yearning for a similar environment for creative uses of programming, either as participant or facilitator. What appeals to me is it is not industry focused (like a bootcamp or working in the tech industry) - which gives room for creative expression - and it isn't a fulltime model (like grad school or environments like the Recurse Center.)

I am deeply interested in the soul of programming, how computing plays a role in the accelerating global challenges of our time. How computing can be bent and warped to more novel and creative uses, and how those uses can get discovered in the first place (as opposed to continuing to implement computing as is currently practiced, i.e. a handful of monolithic companies and codebases dictating the vast majority of online interaction.)

2.4 A Case Study (Optometry Vision Task)

Before the pandemic, I made friends with my optometrist. I shared with my interest in human learning and development, and she in turn told me about some of her graduate work, thirty years prior. She's been part of a foundational study in vision and neuroplasticity, and had stories to tell of giving subjects vision tasks to see if their vision could improve. (Many subjects had substantial emotional responses to the tasks she was asked to administer, and she remembered that thirty years later.)

Intrigued, I tracked down the original study, and tried to replicate it with a friend. Working in a friend's artist studio, his neighbor saw what we were doing, and asked why we didn't just recreate the experiment in software instead of the original hardware, whose optics we were struggling with? A lightbulb went off, and we wrote a first draft in Processing.

On showing it to the optometrist, she became interested in trying the tool out with a vision rehabilitation client, I realized it was going to be a lot to ask someone who wasn't a programmer to install Programming and run a Processing sketch. I asked a second friend, who kindly implemented the code in the browser.

The pandemic came, my dad passed, and I forgot about the work. In remembering it a few months ago, I tried to make some changes to the code, only to discover it was all minified. When I asked if my friend if he had the original code, he apologetically told me he didn't.

I'd like to run a workshop where I could re-build a simple application like this over the course of a 10 week workshop.

2.5 Active Design Questions:

2.5.1 Genre:

Writing workshops organize themselves by genre. Fiction and non-fiction tend to be separate, and in non-fiction, lyric and braided essays can be distinguished from linear essay and memoir.
What genre of programming is right for a 10-week workshop? Right now I'm considering: tools for learning and human feedback, but I wonder if that's too narrow.

2.5.2 Scaffolding:

Programming takes a level of technical knowledge that writing really doesn't. I can't get stuck on my essay not compiling.

Part of what makes a writing workshop powerful is the easy transfer from accomplished works as well as peer works into one's own. It's easy to assimilate new words, ideas, or techniques once they're made conscious. Assimilating pieces of other software though, requires a barrier of engineering. Recurse Center, for example, bills itself as not for new programmers. Is there a necessary floor of experience people need to make a creative programming workshop successful? Is there scaffolding provided for specific tasks or techniques (for example working with audio or video, working with a 2d or 3d game engine) needed to create an environment where assimilating inspiration from professional and peer works happens easily?

2.5.3 Workshop Constraints

One useful constraint pattern in writing workshops is length of work. If writing was assigned weekly, it was 2 pages a week. On a 4-6 week rhythm, it was 15 pages per workshop session. The project of writing a longer form piece, like a book, naturally got broken up into 10-15 pages a month.

Are there similar constraints for creative workshop material? For example, does a single page web application enforce a similar constraint to "no more than 15 pages of writing"? A literal analog would be establishing a constraint in # of lines of code, though I am more drawn to constraints about the length and shape of the experience than it's implementation (i.e., what if a webcam API takes 5k lines of code?)

The constraint of "no user logins or databases" seems to speak against a class of program like the optometrist's example: where you want track someone's progress or lack thereof over a period of time. A softer constraint may be "no data storage more complicated than a simple table that could be expressed in a .csv."

Writing workshops providing an ambitious but acheivable goal in 4-6 week timespans, and I wonder what that the analog is for a creative programming project.

2.5.4. Workshop Inspiration

Inspiration seems hugely important in motivating any creative medium.

Writing is naturally open source.

I expect good pieces of creative programming to be open source (or easily understandable from an inspector), and an experience that can be understood, at least at a surface level, in an evening's worth of time.

I think one piece of work to generating a good creative programming workshop structure will be to come up with the equivalent of a reading list to pull from, week over week.

Independent games and pieces of computational art (like Jason Rohrer's *Passage*,) educational experiments (like Chaim Gingold's *Earth Primer*) easily provoke new ideas.

I think collecting a list of emotionally compelling demoes and web experiences which will be an important piece of creating a workshop.

2.6 An Experiment

I want a creative programming workshop to exist like the workshops I took in creative non-fiction.

There are questions specific to programming: for example, how much technical scaffolding to provide participants (how to deploy an app, what if any UI framework to use, for example), how to invite an audience with good group cohesion (does “apps and games for feedback and learning” create too narrow a focus or too broad of one?) but in general, I am writing this up to try to generalize what I learned in nine months of writing workshop, and to start to see how I can create a similar experience for myself and others in the domain of computing.

Any ideas: great pieces of computing to take as inspiration, other workshop experiments to learn from, and any other feedback is welcomed. I believe there is something deep in the simplicity of the writing workshop format, and I’d like to try applying it to the medium of computing.