

Woes of Programming



SOCI 3500: Occupations and Organizations



September 25, 2024

Programming: Dream Job of the 21st Century?

Ever since I was a young boy, the new and exciting field of programming, or doing anything related to computers for that matter, was touted by all as the ideal job. It was thought that programmers had cozy work environments, high pay, even higher job growth and therefore excellent job security, and the complexities of the work gave it a sort of mystique that would invariably impress most lay people and sound important. There is usually a solid and positively perceived status implication with any career that requires education, is based on complex thinking, and pays well. What is less well-known to the general public, is the dark side of programming. In recent years the tech bubble has burst and growth has just about halted for programming jobs; the stress and hours involved can get to anyone; and in a job that requires logical and concise thinking, it is impossible to solve problems rationally because of myriads of industry standards and illogically compiled development teams. Programming has its perks too, sure; but for anyone who has become even remotely disillusioned with technology already, they probably are not worth the downside.

Family Strain

There are positive aspects to programming in regards to how it allows one to interact with his family, so I will begin there. The nature of the role makes it an ideal salaried job for remote work. If one can put in the hours expected, and then leave work at work, it helps provide more time to be with the family. Even further, if you can remove the commute and work from home; this also gives the bonus of being able to spend lunch and breaks with the family, or to be nearby if there is a family emergency that needs hasty action. These are all great benefits, but they vary and they also exist in a great many other careers and jobs. For example, an employee who is paid hourly is more incentivized to spend more time working overtime rather than to go relax with his family. At the same time, a salaried programmer who is in a position such as team lead may not be able to stop working if there is

an important deadline coming up and the project must be completed on time. These are just two examples of variations that reduce or eliminate most of those benefits.

The stresses of a programmer are largely related to their particular field. Anything near the internet is likely to be higher paying, and much worse on work-life balance. The internet as a whole grew too quickly and now runs mostly on broken and hacked together code that is designed to keep it running just a little longer, ad infinitum. On his blog, Peter Welch writes, “Somewhere there’s a database programmer surrounded by empty Mountain Dew bottles whose husband thinks she’s dead” (Welch, 2014). He is making strong use of hyperbole here, but the point is fully based on the reality: web programmers work crazy hours and have very little time with their families. This is not healthy for their relationships and it is not healthy for them either.

Families are particularly sensitive to high levels of stress due to proximity, importance, and other reasons. APA has an article about the effects of stress on families, “parents who cope with stress in healthy ways can not only promote better adjustment and happiness for themselves but also promote the formation of critically important habits and skills in children” (Managing stress for a healthy family, 2022). The way parents deal with, or do not deal with stress, will ultimately end up impacting their children’s methods of stress management into adulthood. It is important to be living in a healthy and balanced life, both to help set up children for better lives and also to live a fuller and happier life for oneself. Work can be a major factor, “home, workspace, and even social environment can influence your behaviors. Altering you environment can help alleviate stress.” With a job that demands more hours, the workspace becomes an increasingly large contributor to stress. Most fields of programming, but especially web or internet based programming, tend to combine long hours and stressful situations; spelling disaster for one’s mental health and family life.

Networks or Real People?

Community is a vitally important aspect for anyone or any society to have healthy balance and to function well. A healthy community requires seeing people face to face, but a large portion of the tech industry's growth has been based on deliberately dismantling communities for money, and replacing them with cheap imitations. A huge part of the growth of the internet in the past fifteen years has been the development and promotion of social media sites. These all claim to "connect" people together, but the reality is the opposite, as people who frequent these sites can never stop looking at their phones, and have virtually no real-life relationships of any substantive value or health. The 'relationships' online are no better, as they cannot be more than shallow connections, in the vast majority of cases; as it takes years of talking to a small group in private contexts to really develop a relationship, and most social media caters to public interaction rather than private conversations. Real life interactions cater to more private interactions as it is logistically challenging to talk to more than a few people at one time, intrinsically leading to semi-private conversations where true connecting can take place.

What is worse though, is the negative health effects of social media sites, which are ubiquitous and unambiguous; it would not be a stretch to call them "universal," as the determining factors are essentially the frequency of use and the importance attached to them by the user; both of which tend to be highest among younger users. A medical journal article by Dr. Igor Pantic (MD), found that many studies show strong evidence of the negative effects of social media sites (Pantic, 2014). Dr. Pantic writes, "Several studies have indicated that the prolonged use of social networking sites (SNS), such as Facebook, may be related to signs and symptoms of depression." This matches with the easily observed anecdotal and empirical evidence of what happens to most teens when spending all day online; as do the other findings. "Certain SNS activities might be associated with low self-esteem, especially in children and adolescents," this is particularly prevalent in social media centered around images, which are extremely detrimental to the self-image of its users. He goes on, "researchers have

associated online social networking with several psychiatric disorders, including depressive symptoms, anxiety, and low self-esteem,” and these are connected to the design of the websites, as they are engineered to exploit the reward mechanisms of the brain, as the engineers themselves have admitted to. Research has been done on this, but anecdotally it is easy to see that exploiting dopamine and similar functions leads to other problems like anxiety, as these are directly related to the physiology of addiction. This is why it is observed that many of these problems, especially anxiety and the associated depression, go away when the usage of the websites is stopped; because the addiction mechanisms are starved and the cycle breaks; restoring the brain to equilibrium.

Programming as a career has contributed heavily to the decay of communities everywhere, but the career itself is also touched by this culture. The mindset that goes into a career in programming does not leave much room for going out and engaging with the community, it is largely reduced to the work community itself. Programmers who work from home have the best chances of engaging with real communities, if they live in a neighborhood that is active in promoting get-togethers. The nature of programming separates one’s mind from reality in an abstract way, as Peter puts it, “Doing this all day leaves you in a state of mild aphasia as you look at people’s faces while they’re speaking and you don’t know they’ve finished because there’s no semicolon” (Welch, 2014). The intense focus and strange thinking required can cause an almost dissociative effect, and not in a good way. What might a good form of dissociation be? The phenomena that occurs when locking-eyes with a stranger, wherein it seems that reality fades away, can be described as dissociative; but the key factor being that it involves connecting with a person rather than disconnecting with people. There is one particular place where one may say programmers thrive as a community, that is in the educational stage.

Programmers’ Best Years

One of the main distinctions of programming is that it is an industry that requires education. This is not necessarily formal education, as many good developers teach themselves or learn informally

and have no college degree; but the overwhelming majority take the university route to a career. This usually involves a bachelor's degree in Computer Science for any type of software development programming, or some similar associate's degree for web development or other information technology roles. The educational years can be a highlight for the tech industry, not only because of the age bias the hiring managers have, but because of the clubs and events targeting college aged programmers for engagement. One of the biggest of these is Codebreakers, a Cybersecurity oriented challenge that has colleges competing against each other to solve simulated cyber scenarios like the FBI or NSA would, and since it is hosted by the NSA, it is likely a big recruiting tool as well. At the right schools, there is a lot of opportunity for social growth for young programmers; but it is the last place they will be encouraged socially, as most tech companies do not care about them at all.

Show Me the Money

Any young person in America today has grown up in a culture that has taught them the importance of money. Some of this is good, a lot of it is bad. It is good to be concerned with providing for oneself and one's family, and to be financially responsible and fiscally literate. It is profoundly unhealthy to obsess over money as a status symbol and base one's entire life on the pursuit of money. I would not say this applies directly to programmers, there are other easier ways to make more money; but it is common for careers to be chosen because of what seems like a high salary for low physical effort. Unfortunately the mental stresses and emotional tolls are not usually considered, as can be seen with many high-class lawyers who hate their jobs but are addicted to the first-class lifestyle and have to work in those jobs to pay off their law school loans and maintain their lifestyles. This applies to programmers in a more subtle way, as it is one of those "perfect" jobs that has everything someone should look for on paper, according to the older guiding generations; and many have been pushed into computer work under the idea that they would be able to "write their own ticket." It was also perceived

as an extremely safe industry, with record growth and no clear signs of slowing, but as anyone living in 2008 knows, bubbles burst.

One of the clearest ways to see the tech bubble bursting in recent history was in 2020, when “technology stocks reached irrationally high valuations” (Tokic, 2020). He makes the case that this was a bubble by comparing the PE ratio for S&P, which is normally around 15, but in the late Summer of 2020 it “reached above 35, which is comparable only to the peaks of prior bubbles”. The PE ratio is price to earnings, meaning a higher ratio signifies a higher price on a stock than the company is earning, relative to a lower PE ratio. On the stock market, the bursting of this bubble was simply a loss of value for the stocks and those who held them; but stock market actions tend to reflect (or be reflected) in real-world conditions as well.

The Occupational Outlook Handbook is published by the United States’ Bureau of Labor Statistics (BLS), and updated frequently. It offers insight into the trends of employment with average salaries, requirements to get a certain job, and the growth or decline trends for the decade. According to the BLS, the trend for Computer Programmers is declining at a rate of ten percent from 2023 to 2033 (U.S. Bureau of Labor Statistics, 2024). This represents an elimination of about 13,400 programming jobs over the next ten (now nine) years. As a result of most tech companies losing money from the stock bubble bursting, many of them have been laying off all non-essential personal. Many startups failed or ended, and many programmers who had a cornucopia of employment options are finding that they cannot leave their job because nothing better is available now. Anecdotally, my brother is unhappy with his job as a programmer, but when he looked he realized there are no good alternatives right now. In another case, I have a friend who is a web developer, who requested to be transferred away from a team that was working on exploitative software, and that entire team was later laid off; leaving him with a job only because he moved to more ethical and important tasks. This job market is a very different situation from what we saw a couple of decades ago. Keith Mander wrote, “The late

1990s were characterized not by massive unemployment among programmers worldwide, but by considerable shortages of such skilled personnel” (Mander, 2001). Indeed, the prediction that Ed Yourdon had made was premature, and perhaps exaggerated quite a bit too, but the American Programmer is declining now because of “international competition” as he predicted (Yourdon, 1992).

It is still possible to survive and live well as a programmer financially, but times of endless cushy job-offers have changed, there is much less freedom and less job security now as the tech bubble has begun to find an equilibrium after inflating too far. The high average wage of programmers and relatively low barrier to entry make it a decent option for the right kind of people, but it still remains much less glamorous than it has been promoted as for the last few decades. The median pay according to the BLS is a \$99,700 salary per year (U.S. Bureau of Labor Statistics, 2024). A decent amount, but in today’s world of inflation and an ever-rising cost of living, that is coming closer and closer to being just barely enough to get by with. Programming is not the “get rich easy” career that it used to be, and it seems that a similar fate is rushing towards Cybersecurity professionals as their field’s growth is grossly over-estimated in a similar way, leading to a bubble of demand that will quickly be popped by a saturated labor market. When a career starts being promoted by culture for the sake of earning a lot of money, chances are high that it will crash sooner than later, because that money is limited and a lot of people are going to be chasing after it, first come first serve.

Programmers in Politics

The field of programming is under increasing scrutiny by the federal government for the effects some of the popular social network sites has had on young people. Public Finance reports that many top CEO’s of these companies have directly faced Senate Judiciary Committee hearings for fallout caused by their apps or websites, “including Mark Zuckerberg of Meta and Tiktok’s Shou Zi Chew” (Dark day for web bosses, 2024). The previously mentioned negative effects of social media do not end simply with negative mental health outcomes, but those outcomes themselves often end in very real

and troubling actions. The Public Finance article lists some of those things, “But the parents, whose children had either self-harmed or committed suicide, said [Zuckerberg’s apology was not] enough and called for tougher legislation.” The main outcome of the Senate Judiciary Committee hearing, perhaps besides raising awareness of these problems, was promoting the concept that social media sites must be held accountable as de-facto public squares of today. They should either allow free speech or be held accountable for what goes on in their platforms; many believe both should be done, but as the law stands neither is true. At the hearing, Senator Lindsey Graham said, “I am tired of talking. I’m tired of having discussions. Open up the courthouse door. Until you do that, nothing will change. Until these people can be sued for the damage they’re doing, it is all talk.”

As the prominence of tech increases, the liability and legal troubles facing programmers will too. Zuckerberg started out as a programmer, and he now owns and operates one of the largest tech companies in the world. Should he not be held liable when people use those platforms for a purpose they were essentially designed for: to stalk and harass others? He did once state that all of his users were “idiots,” bringing up serious ethical concerns with the concept of corporate immunity. These issues will only get more complex before they get solved. Is a small time programmer liable for helping to program a small piece of code that eventually gets modified into hacking software? Ideally not, but laws have to be carefully written and thought out to decipher who the guilty parties are, and what the intent of their work was. Freedom of information comes into play too. Jack Sweeney is a college student who would track celebrities’ private jets as a hobby. He used public information to do so, “the system relies on publicly available data collected by amateur enthusiasts” (Wendling, 2024). This has angered Elon Musk and Taylor swift, but Sweeney has done nothing wrong, nor anything to endanger them. After an incident where someone was on top of Musk’s plane prompted him to ban Sweeney, it was revealed that Sweeney’s tracking had nothing to do with it, “Police later identified a member of Mr Musk's security team as a suspect, and said that Mr Sweeney's account had nothing to

do with the incident” (Wendling, 2024). Sweeney tracks jets for transparency and accountability, but also because it is just interesting to him, and that should be his right as a programmer. Unfortunately those with money and influence can often buy the law, and an amendment was passed making the tracking of private jets more complicated (Mishra, 2024). Mishra writes, however, that this does not make the hobby impossible according to Sweeney, who says he will “push even harder on tracking” because of it.

The politics of programming are complicated, and staying legally protected will only get more challenging as technology continues to evolve and reach into more areas of peoples’ lives. Some things are permitted that should not be, and many things are scrutinized or banned that should be protected freedoms instead, and programmers get caught in the middle quite often.

Programming for Faith

Many religious institutions have websites in the modern age. Having seen many different church websites, it is easy to see when a skilled web programmer has been tasked with creating one as opposed to a novice or a do-it-yourself service being used. These websites are increasingly important as people like to research anything before they give it more time, and it is particularly important to people to see what a particular institution of faith, such as a church, is about before they spend time and effort to visit one. Everything from the website’s organization to its style can speak volumes about the type of people the church wants to attract, and who would fit in well with them. Sometimes the aesthetic design of a website can tell a user more than the content of the site itself. So the task of programming a website for a religious institution becomes a very important task for the success of that institution in gaining potential followers and spreading its message.

In other areas of institutionalized religion that sometimes overlap with more areas in society, we see a similar importance in programming. Many private universities are tied to a particular religious faith, becoming both educational and religious institutions. For these universities, many have adopted

using the latest and best technology for their websites and application systems. A majority of universities today have entirely electronic application systems, and it was important that a programmer made those systems work well and reliably, so that prospective students could apply and become admitted.

Thank a Programmer

Programming is not such a glamorous job, it is not easy, and it is often thankless. The world today relies on their role however, even the most techno-skeptic among us, if we remain in society at all, rely on the work of programmers to keep things running smoothly for us and make life more convenient and easier. There are a lot of contentions around technology, how it is used and where it is going, and every side has valid points, but at the end of the day most individual programmers are doing their best to improve the world. Anyone who sees the profession for what it is, the social strains, the long hours, the irrational assignments, and the personal legal liabilities; and still goes into it, is a special person who probably has a very selfless personality. Even the government requires programmers to run well, as all government websites and technological systems were created and are maintained by programmers, providing numerous public services and data. So while programming is not a job I can recommend, it is a profession that deserves some consideration and respect. If we enjoy our modern conveniences at all, we should thank a programmer.

References

Dark day for web bosses. (2024). *Public Finance*, 14–15.

<https://research.ebsco.com/linkprocessor/plink?id=086b5a86-f676-320b-afc4-b53f28a029f9>.

Managing stress for a healthy family. (2022, December 21). American Psychological Association.

<https://www.apa.org/topics/stress/managing-healthy-family>

Mander, K. (2001). THE DECLINE AND FALL of the AMERICAN

PROGRAMMER? *Communications of the ACM*, 44(7), 43–44.

<https://doi.org/10.1145/379300.379309>

Mishra, P. R. (2024, May 23). *Taylor Swift's jet emissions row leads to US barring private plane*

tracking. Interesting Engineering. <https://interestingengineering.com/culture/taylor-swifts-jet-emissions-us-bar-tracking>

Pantic I. (2014). Online social networking and mental health. *Cyberpsychology, behavior and social*

networking, 17(10), 652–657. <https://doi.org/10.1089/cyber.2014.0070>

Tokic, D. (2020). Robinhoods and the Nasdaq whale: The makings of the 2020 big-tech bubble. *Journal*

of Corporate Accounting & Finance (Wiley), 31(4), 9–14. <https://doi.org/10.1002/jcaf.22473>

U.S. Bureau of Labor Statistics. (2024, August 29). *Occupational Outlook Handbook*. Computer

Programmers. <https://www.bls.gov/ooh/computer-and-information-technology/computer-programmers.htm>

Welch, P. (2014, April 27). *Programming sucks*. Still Drinking.

<https://www.stilldrinking.org/programming-sucks>

Wendling, M. (2024, February 21). *Jack Sweeney: The planespotting student who angers Taylor Swift*

and Elon Musk. BBC News. <https://www.bbc.com/news/world-us-canada-68248168>

Yourdon, E. (1992). *Decline and Fall of the American Programmer*. Prentice-Hall.