Does the thrill of competition propel computer science forward?

The two men had different reasons for playing a game of foosball that night, but neither indicated what they were. One was taking time out to compete in a casual game, where the stakes were low; the other was facing the only human competition of the day. They both stood over the table with the same hunched stance, sending the ball flying with equal force.

For the CEO of a midsized software services company, the game came at the end of the day. He had completed hours of meetings to raise capital, find customers, and keep the organization working harmoniously. He was involved in a competition with his board, having crossed swords with them earlier this year, temporarily losing his job. He was not about to lose it again.

For his opponent, one of his IT programmers, the game was an opportunity for competition with a human before settling into his nightly race against time. Before the sun rose, he needed to complete a given number of lines of code, assure their accuracy, and integrate them into the system.

By day, many computer science practitioners work in dull office buildings with gray walls and long hallways. By night, they play—for their own benefit, their own fun. Whether that means playing games or creating them, the thrill of competition is their prize for the working hours.

“Why buy when you can code?” asked one computer scientist friend as he demo-ed his new app. “I created this just for fun. It’s in the store, but I don’t expect many to buy it.” That’s what they all say, I thought.

Most wannabe technology tycoons have heard the stories of Silicon Valley garages. Yet, after spending time in the office or otherwise supporting the health of their bank accounts, the creative often find ways to compete against themselves. Can I make this, and will it work? Will people like it and find it useful? If the answer is “yes” to all of the above, then it’s not just a personal success but also likely a commercial one.

Forecasts say the worth of the gaming industry will amount to more than $80 billion in the next five years. Software for the workplace is widespread; there’s a growing market for software to fill free time.

Gaming has an odd relationship with software and software engineering. Many individuals entered the field because they had played videogames in their youth. Few probably realized that the work of designing and developing programming systems, even when those systems are videogames, has little in common with the act of playing a game. Nonetheless, games and gaming seem to hover over software engineering, and our technical staffs turn to competition even when their jobs fail to provide it.

Both foosball players that night cared deeply about the game’s result. They cheered at each point and cursed each misplay. When the CEO scored the final goal in the waning moments of play, the programmer demanded a rematch, which would certainly happen when time, tide, and the demands of the software business calendar allowed.

David Alan Grier is an IEEE Fellow and author of the forthcoming book The Company We Keep. Contact him at grier@gwu.edu or on Twitter @dagrier.

Erin Dian Dumbacher is a research director and consultant in Washington, DC. Contact her at erin.dumbacher@fulbrightmail.org or follow her on Twitter @erin_dian.

Selected CS articles and columns are available for free at http://ComputingNow.computer.org.