

# Personal History Statement

Whitaker Thompson

---

I took my first theory of computation course during my sophomore year, and I was obsessed with the Cook-Levin Theorem and the existence of natural NP-complete problems. Though I was initially disappointed to learn that the P vs NP question was out of the reach of current techniques, the disappointment was short-lived when I learned that there have been many meaningful, elegant results about certain relaxations of that problem. I took an algorithms course the following semester that was taught by Euiwoong Lee, who is mainly interested in approximation algorithms, and I was quickly entranced by these algorithms and the accompanying inapproximability results.

The following semester, I signed up for courses that would primarily boost my software engineering resume, none of which were theory courses. However, I found that in all of my spare time, I was reading about algorithms and complexity and the ideas that had so excited me a year before. Instead of looking for an internship for the upcoming summer, I reached out to Euiwoong Lee and asked if I could work on a research problem with him. He said he would be happy to work with me, and in the words of Chris Peikert, who taught my introductory theory of computation course, I “joined the big-O life for good.”

The theory courses I took during my last year of undergrad were mainly small, graduate-level courses and it was in these classes that I began to see the importance of not just technical ability but of having a *personality* that lends itself well to studying theory. In addition to having a strong technical grasp of the subject, the professors in these courses also were clearly still amazed and surprised by the material, and their enthusiasm instilled something similar in their students. When I met with Euiwoong Lee to speak about our research problem, I usually had at least one other unrelated theory question for him, and he always took the time to answer them. I look forward to making the department at New York University a place where the academic ecosystem is at its best; a place where the mentor and teacher can transfer to the mentee and student not just technical knowledge, but an appreciation for theory and a recognition of its beauty and importance. In pursuit of this goal, albeit on a different scale, for the 2025-2026 school year I accepted a job as a supplemental math tutor at a high school in Boston through Americorps. Unfortunately, in

late June, I found out that due to funding issues, the job would no longer be offered. In order to continue developing as a computer scientist, I opted to stay in Ann Arbor, working a retail job while I unofficially audited some graduate theory courses at Michigan. This has been a fantastic way to stay connected to a strong department, and to grow intellectually while supporting myself financially.

While this year is not what I planned it to be, I have learned two main things. Since my job through Americorps fell through, I have been put in a situation that has befallen countless scientists in American academia this year. I learned that funding is fragile, and that computer science does not happen in a vacuum. The current political moment influences the things that we, as computer scientists, love and think are important, and I am glad that I will be cognizant of this fact as I am taking the next step in my academic career. The second thing that this year has taught me is the somewhat paradoxical fact that contributing to the body of knowledge of computer science does not have to be limited to those who receive funding to do so. I have been able to learn more theory, to work on research problems, to think about what the leaders in the field are thinking about, all while surviving off of a salary from an hourly retail job. I am learning to internalize my locus of control, and I know that when I eventually do earn a salary to study computer science, I will be better for having been through this year.