Computer Science Coding Ethics

I Understand

- **I understand** that, in solving computer science homework and projects, I am allowed to work with other students, Teaching Assistants, and the Instructor **at an algorithmic and conceptual level** to help me better understand how to solve the problem at hand. The take-aways from any such sessions are pictures and potentially English-based steps to take, and should never involve code itself.
- **I understand** that using **any** code that **I did not myself write** from any student, Teaching Assistant, or obtained from the Internet is a violation of the Denison Academic Code of Conduct, and is henceforth called cheating.
- **I understand** that such cheating, and representing as my own work, may involve sequences of instructions (instruction blocks), functions, or whole programs, and the particular size is immaterial to the cheating offense.
- **I understand** that such cheating is defined for notebook-style computing just as much as for compiled programs and complete-program scripts.
- **I understand** that attempts to mask the cheating by

  - changing variable names
  - rearranging code block order
  - using different docstrings or other commenting style

  does not change the underlying cheating.
- **I understand** that a student that **provides** code, enabling others to then represent it as their own, is **equally culpable** in the violation of Academic Integrity.
- **I understand** that the instructor, in order to provide an honest, fair, and ethical class is **obligated** to find such violations. This includes automating the process of extracting code, and packaging it for software comparison tools, including the Stanford **Measure Of Software Similarity** system, which accounts for the superficial differences named above in attempts to mask cheating.
- **I understand** that, other than the most trivial individual homework exercises, any two students code submissions will be substantially different, given the many many ways of solving the same problem, the different choices of functional decomposition, and, in the case of sophisticated modules (like `pandas`) the myriad of interface and tool options.

Signature

> **Print Name:**

-

> **Signature:**