**COMPENG 2SH4 Project – Statement of Contribution**

Your Group Name \_\_\_\_\_\_\_\_\_\_\_\_\_sudo rm -rf /\*\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Your Name \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_Ethan Su\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Your Team Member’s Name \_\_\_\_\_\_\_\_\_\_Dylan Manamendra\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

**You must complete this statement of contribution without discussing it with your project partner, i.e., individually. Your statement should be concise (at most one-and-a-half page). It has three parts:**

1. Tell us about your own contribution to the development of your COMPENG 2SH4 project. For example, you can tell us about which project iterations (as mentioned in the project manual) and C++ project classes that you worked on and completed. You can provide a concise answer either in paragraph form or through bullet points.

I completed the special member functions, developer 1 tasks in both phase 1 and phase 2, and integrated most of the functional code including the above-and-beyond tasks.

* Special member functions as a part of preparation stage
* Developer 1 tasks (Phase 1 and Phase 2)
* Array integration, collision detection, base game snake body growth, base game score system and game over condition

1. Repeat Part 1 above but this time tell us about your project partner’s contribution to the development of your COMPENG 2SH4 project.

* Developer 2 tasks (Phase 1 and Phase 2)
* Game over messages
* Above and beyond snake body growth and game score system

1. Tell us about your experience in your first collaborated software development through this project – what was working and what wasn’t. If you are a one-person team, tell us what you think may work better if you had a second collaborator working with you.

Individually working on code was easy since I would be familiar with what I wrote, but working on integrating my code with my partner’s required gaining an understanding of their code before I could make changes. This was especially difficult in the main code, where changes would be made often by both of us. We would have to coordinate who pulls and pushes code so that they didn’t overlap or interfere with the other person’s progress. This also meant that I sometimes needed to wait for my partner to finish their part before I could implement mine as it would rely on their code.