COMPENG 2SH4 Project – Statement of Contribution

Your Group Name	Greg Cousins Fan Club
Your Name	Joshua Guo
Your Team Member's Name	Suchir Laddas

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.
   - Completed parts 1a, 2a, and the first two parts of iteration 3 regarding snake movement and snake growth logic when eating the food.
   - Also did part of 1b when completing the further iterations(I implemented the required accessors in order to progress work on the project)
   - I was mostly in charge of debugging the project and fixing any memory issues the program experienced.
   - I also finished the bonus part of the project.
2. Repeat Part 1 above but this time tell us about your project partner's contribution to the development of your COMPENG 2SH4 project.
   - My partner did everything else
     - 1b, 2b, and the last part of iteration 3(the snake death logic)
     - He also oversaw cleaning up all the unnecessary code and deleting comments
     - Added comment for all the code for ease of readability (in all files)
3. 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.

Overall, I think the project flowed very smoothly. I explicitly mentioned that all changes should be tested before pushing to GitHub, and every time you wished to work on it, pull the project. Also push changes whenever possible. We ended having over 40 commits, but that meant that working together was seamless and there were no conflicting changes. However, I was very tempted to implement everything myself, as it did take more effort to integrate all the classes together at the end, because it required me to read and understand my partner's code. I feel like most of the prevalent issues of working together on a software team wasn't shown here, because the separation of tasks was pretty clear and after breaking the project into parts, it was very individualistic in the work being done.