COMPENG 2SH4 Project – Peer Evaluation

Your Team Members	Jana Nofal	Lujain Nofal
Team Members Evaluated	Tasmiah Gani	Tajmim Maisha

Provide your genuine and engineeringly verifiable feedback. Ungrounded claims will lead to deductions.

Part I: OOD Quality

1. **[6 marks] OOD is about sensible code modularization. Looking at the header files of each object, can you easily interpret the possible behaviours of the objects involved in the program, and how they would interact with each other in the program? Comment on what you have observed, both positive and negative features.**

- Yes, I can easily interpret the possible behaviours of the objects involved in the program by looking at the header files of each object. Moreover, the code had many good methods with self explanatory names. This aspect aided with the organization of the code and allowed us to interpret the behaviors of the methods easily.
- Comments throughout the program were very straightforward. Each section in the program had at least a title that helped us through this evaluation. However, it would be a good idea to add more comments explaining the behavior of the code to aid the reader, who might not have any idea about the project, to better understand the function of the code.

2. **[6 marks] Examine the main logic in the main program loop. Can you easily interpret how the objects interact with each other in the program logic through the code? Comment on what you have observed, both positive and negative features.**

- Examining the main logic in the main program loop, we can easily interpret how the objects interact with each other. The code is well organized in that section. Moreover, the group used good variable names that aided with the professionalism of the program and in our overall understanding of the code. However, it would have been a good idea to specify with a comment the function of each variable and what it will be used for later, as it is hard to keep track of what the many variables are used for. Having the title as a main comment at the top of a section helped us know what each section's function was about.

3. **[5 marks] Quickly summarize in point form the pros and cons of the C++ OOD approach in the project versus the C procedural design approach in PPA3.**

Pros:

- The header files of each object allows the code reader to predict what a specific object does and what it is associated with.
- In the C++ approach, we were able to make header files that specify our functions in the project, allowing for more organization. In the C procedural design approach in PPA3, the program was

less organized and it was jumbled up. This made it hard for the reader to predict and understand the code.
- C++ OOD approach in the project is easier to follow in terms of understanding, organization, and predication than in the C procedural design approach in PPA3. In PPA3, you would waste your time just trying to remember and look for the functions you've created. This aspect restricted efficiency and professionalism.

Cons:

- C++ OOD approach has a steeper learning curve. It takes a while to get used to, and get familiar with how the different classes work together.
- The C procedural design approach for PPA3 required less brain power to make the program work than for C++ OOD approach as we lacked experience working with OOD.

Part II: Code Quality

1. **[5 marks] Does the code offer sufficient comments, or deploys sufficient self-documenting coding style, to help you understand the code functionality more efficiently? If any shortcoming is observed, discuss how you would improve it.**

-The comment made in this code provides some information about the purposes of each function and certain objects within the code. However, there could be room for improvement by adding more detailed comments to enhance understanding .

-For example, the comments in the RunLogic function can include a bit more detail about the specific logic involved in the food collection and self collision detection. In addition, the comment in the DrawScreen could be commented out more to explain the logic behind drawing the player and game board. However, there is a sufficient amount of comment present and a person can easily understand the overall code.

2. **[4 marks] Does the code follow good indentation, add sensible white spaces, and deploys newline formatting for better readability? If any shortcoming is observed, discuss how you would improve it.**

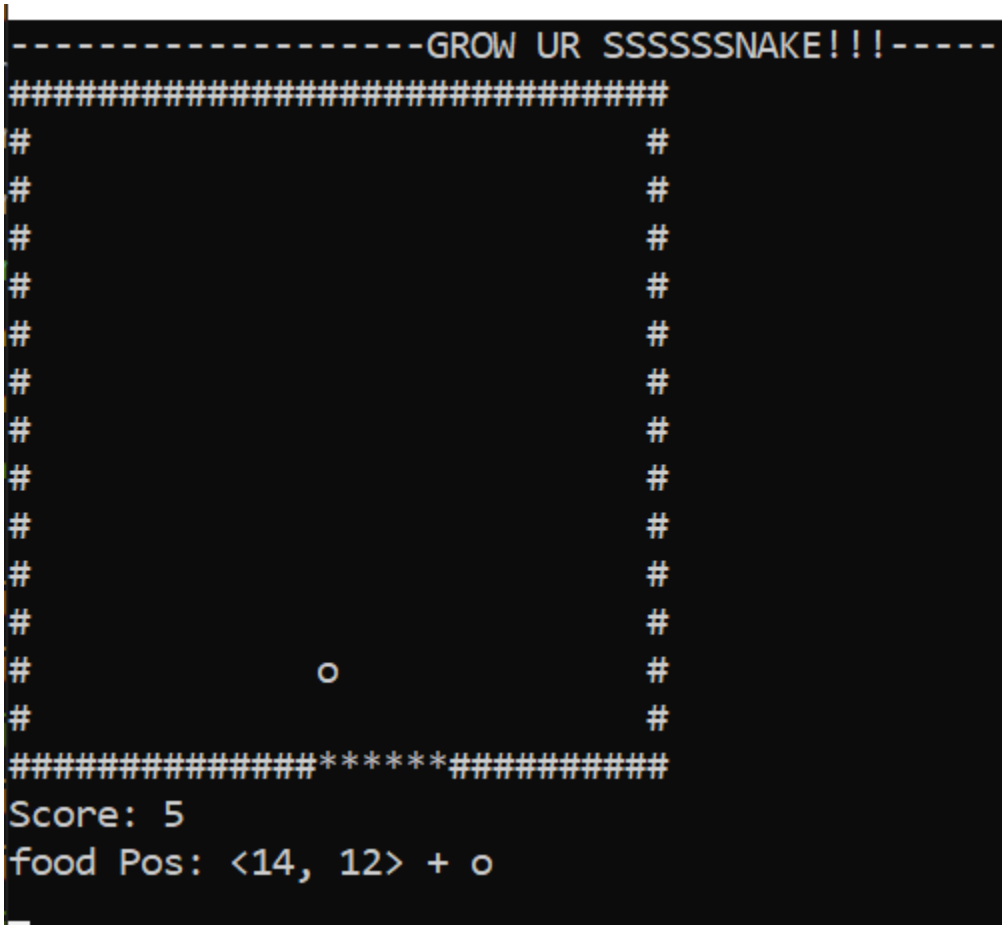
-The code is well organized and easy to follow. The code follows an overall good indentation for every function, for loop, and if statements. The comments are consistent and are helpful to understand the code better. Overall, the code is thoroughly organized and well constructed. Moreover, the deployed newline is very well implemented for better readability especially in the game board.

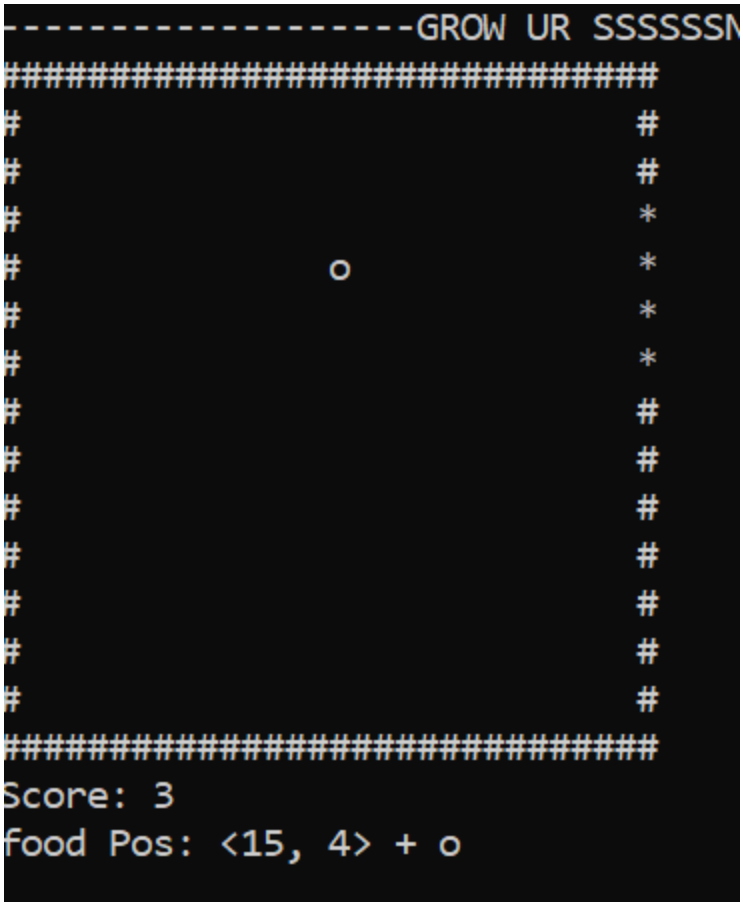
-Furthermore, including a large blank space between each major function could enhance organization and readability.

Part III: Quick Functional Evaluation

[8 marks] Does the Snake Game offer smooth, bug-free playing experience? Document any buggy features and use your COMPENG 2SH4 programming knowledge to propose the possible root cause and the potential debugging approaches you'd recommend the other team to deploy. (NOT a debugging report, just a technical user feedback)

There is a bug with the code. The snake does not do the wraparound feature properly. When the snake reaches the bottom edge of the board and the left edge of the board, it moves through the # border, instead of going to the other opposite edge. This bug only occurs when the player goes down and right. This bug can be seen in the two images below.





The possible root cause of this bug might be a result of line 105 and line 119 as shown in the pictures below. The condition should not be evaluated when the current head reaches the BoardSizeY or the BoardSizeX. To stop the head from reaching the borders, we should compare the currentHead with a value that is 1 or more units less than the board size value and try out these different values until the program works accordingly.

```
103             case DOWN:
104                 currentHead.y++;                //move down array
105                 if (currentHead.y >= mainGameMechsRef->getBoardSizeY()) {
106                     currentHead.y = 1;
107                 }
108                 break;
```

```
117             case RIGHT:
118                 currentHead.x++;                //move right of array
119                 if (currentHead.x >= mainGameMechsRef->getBoardSizeX()) {
120                     currentHead.x = 1;
121                 }
122                 break;
```

[6 marks] Does the Snake Game cause memory leak? If yes, provide a digest of the memory profiling report and identify the possible root cause of the memory leakage.

There is no memory leak for the team's Snake Game.

```
~~Dr.M~~ ERRORS FOUND:
~~Dr.M~~       0 unique,     0 total unaddressable access(es)
~~Dr.M~~       9 unique,    12 total uninitialized access(es)
~~Dr.M~~       0 unique,     0 total invalid heap argument(s)
~~Dr.M~~       0 unique,     0 total GDI usage error(s)
~~Dr.M~~       0 unique,     0 total handle leak(s)
~~Dr.M~~       0 unique,     0 total warning(s)
~~Dr.M~~       0 unique,     0 total,      0 byte(s) of leak(s)
~~Dr.M~~       0 unique,     0 total,      0 byte(s) of possible leak(s
)
```

Part IV: Your Own Collaboration Experience (Ungraded)

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.**

Our first experience in our first collaborative software development through this project was positive. We both did our responsibilities accordingly. Moreover, we would collaborate together if we needed help and to explain to each other how our specided parts work to make sure we were both on the same page and gaining the proper experience. An issue that we faced was with time. We doubted how much time and effort the project would take and ended up having to cram up the project over a considerably short period of time. We should have committed ourselves to a specific schedule and followed the professor's recommendation. But overall, this collaborative experience was positive and working closely alongside each other ensured that we ran into less errors.