

COMPENG 2SH4 Project – Peer Evaluation

Your Team Members

Brayden Roberts & Diya Bahl

Team Members Evaluated

Liam Sullivan & Nihal Inel

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.
 - a. Yes, the header files of each object can be easily interpreted since the functions in each of the header files have attributes that can be associated with the main ideas of each of the files. Everything that was in each of the header files was appropriately placed. However, it seems that they were missing quite a few functions in the Player.h file; one of them being to check for the collision of the player- a requirement in the game.
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.
 - a. The code makes it easy to interpret how different objects interact with each other. In the drawScreen function it's clear that the player body is being drawn on the gameboard stored in the GameMechs class and the same with the food items from within the GameMechs class. It is also clear what keeps the game running: in the main loop, the while statement is calling for the exitFlag from the GameMechs class, so it seems to be that is what is determining the game continuation.
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.

Part II: Code Quality

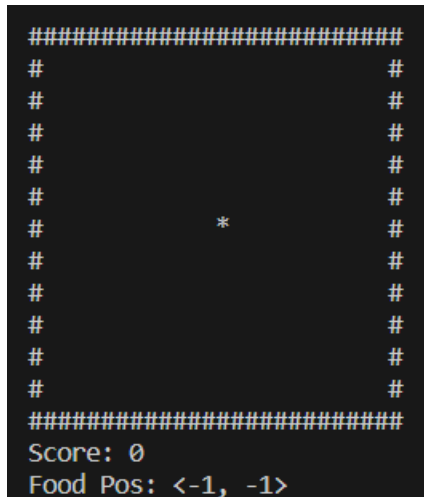
1. **[5 marks]** Does the code offer sufficient comments, or deploys sufficient self-documenting coding style, to help you understand the code functionality more efficiently? If any shortcoming is observed, discuss how you would improve it.
 - a. When analyzing the code, it seems that it does not deploy enough commenting. A lot of old guiding comments are also left in the code which do not seem to have been cleaned up. Furthermore, there are some comments in the code which seem to have been only implemented for temporary usage (which should have been omitted but are not). To improve code documenting, I would recommend adding comments in functions with more lines of code to describe what is happening instead of having the reader assume

and guess what the code could be doing. Furthermore, I would recommend eliminating starter code as well to clean up sections of comments which are not necessary.

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.
 - a. Yes, the code follows good indentation, newline formatting and white space use. The code is easy to interpret with regards to conditional statements and loops, and code lines are grouped into reasonable sections with newline separations based on their functions.

Part III: Quick Functional Evaluation

1. **[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)
 - a. The game offers a mostly bug free experience, with a few bugs that cause issues in playing the game. The first bug was that occasionally the food would start at the position -1, -1 as shown in a figure below. This made the game unplayable as there was no way to get food. The root of this looks to be in the generate food function where -1, -1 is being fed in as a blockoff position, but the function only runs once. My guess is that if an unusable position is generated it doesn't make a second attempt, meaning it goes to -1, -1 by default. Another bug is that the food can generate on the player, which is likely because the function to generate food is not checking the potential position against each snake segment. So, the generate food function needs to be further checked for if it is accepting and iterating through the snake segments before generating a food position. Finally, there does not seem to be any lose condition/snake overlapping checking, which seems to be either a lack of a function checking that, or whatever function does exist is not checking each segment of the snake, in much the same way as the food generation function has that issue.



2. **[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.
 - a. There was a memory leak, but it was not caused by any of their code. The photo of the report is below, and it is reported to be coming from MacUILib.c so it was not a result of their code or something they would have been expected to prevent.

```
Error #9: LEAK 2 direct bytes 0x01720c90-0x01720c92 + 0 indirect bytes
# 0 replace_malloc [d:\a\drmemory\drmemory\common\alloc_replace.c:2580]
# 1 msvcrt.dll!_strdup [C:\Users\nihal\VSCode\COE2SH4\Project\Project\2sh4-project-liam-sullivan-and-nihal-
inel/MacUILib.c:45]
# 2 .text [C:\Users\nihal\VSCode\COE2SH4\Project\Project\2sh4-project-liam-sullivan-and-nihal-
inel/MacUILib.c:45]
# 3 __mingw_glob [C:\Users\nihal\VSCode\COE2SH4\Project\Project\2sh4-project-liam-sullivan-and-nihal-
inel/MacUILib.c:45]
# 4 _setargv [C:\Users\nihal\VSCode\COE2SH4\Project\Project\2sh4-project-liam-sullivan-and-nihal-
inel/MacUILib.c:45]
# 5 .text
# 6 mainCRTStartup
# 7 ntdll.dll!RtlInitializeExceptionChain +0x6a (0x774ebd2b <ntdll.dll+0x6bd2b>)
# 8 ntdll.dll!RtlClearBits +0xbe (0x774ebcaf <ntdll.dll+0x6bc9f>)
```

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.
 - b. Being able to have different people working on different sections through dividing up the classes was definitely very helpful in reducing the workload. It was nice to only have to worry about the functionality of one section knowing someone else has made the other class and made sure it works as it should. Meaning you only need to read the header file and some comments to understand how to use it and how it can be useful. Some challenges we had early on though were that we didn't fully communicate what sections we were working on, which lead to some programming overlap and then some issues in pushing and pulling the code through GitHub. Clear communication would definitely be a factor to improve on in order to prevent reprogramming the same section twice. It also felt weird to try and work on it sometimes because it felt like we needed to never both be coding at the same time to prevent difficulties in getting our changes to the remote repository, especially as we got into later iterations and sometimes debugging would cause you to cross into all the major class files.