Jeffrey Wong
Jessy Chahal

# CMPT Assignment 3: Code Review

## Identifying code smells that require refactoring and where they are located

| # | Code Smell | Classes that Need Refactoring |
|---|---|---|
| 1 | Bad / confusing variable names | - Reward and Bonuses Classes<br>  - Inside the constructors<br>  - Parameter, that stores the score value of a reward, called "rs" can be confusing for others looking at the code<br>- Enemy Class<br>  - Some private variables do not really describe what they are doing (i.e. characterLeft - does this refer to characterLeft position, image, coordinates - hard to determine just by looking at it). |
| 2 | Unused / useless variable | - Enemy (Constructor):<br>  - Notice that there is a variable that can be initialized outside the constructor and is not required inside the constructor itself as it does not change or update. |
| 3 | Dead code | - ScreenPanel Class (Constructor)<br>  - Setting enemies to null, then instantiating right after in the subsequent lines of code is redundant |
| 4 | Methods that are too long and that could benefit from being refactored | - Enemy Class (MoveToPlayer method):<br>  - The moveToPlayer method is handling too much and can be split into smaller helper functions that are later called inside the method itself. |
| 5 | Unnecessary if / else | - MoveToPlayer class (Enemy):<br>  - There are too many nested if else statements that make it hard to understand and read the code<br>- checkObjectCollision (Enemy):<br>  - The if statement essentially repeats with only one variable being updated. Can create an array and iterate over. |
| 6 | Data Clumping | - ScreenPanel Class<br>  - Certain reward objects (water and bread) are clumped up due to their large quantities |

Jeffrey Wong
Jessy Chahal

**How the Refactoring was accomplished**

| # | Code Smell | Changes Made |
|---|---|---|
| 1 | Bad / confusing variable names | - Reward and Bonus Classes<br>    - Changed parameter name from "rs" to a more suitable and fitting name "score"<br>- Enemy Class<br>    - Changed private variables so they are more informative (i.e. characterLeft to characterLeftImage, imageDirection to imageFacingDirection). |
| 2 | Unused / useless variable | - Enemy (Constructor):<br>    - Removed useless variables that do not have to be initialized inside the constructor |
| 3 | Dead code | - ScreenPanel Class<br>    - Removed the redundant code within the constructor<br>    - Set enemies to null as their default attribute values in the class |
| 4 | methods that are too long and that could benefit from being refactored | - Enemy Class (MoveToPlayer method):<br>    - Created helper functions that are later called inside the MoveToPlayer method |
| 5 | Unnecessary if / else | - MoveToPlayer class (Enemy):<br>    - Made each if else case into a function that is later called in replace of the statements inside the method.<br>- checkObjectCollision (Enemy):<br>    - Created an array with all the id's and iterated over that instead of having many "or" operators in the if block. |
| 6 | Data Clumping | - ScreenPanel Class<br>    - To prevent clumping, we bundled all the water into an array and all the bread in another array<br>    - Within the constructor, loop through these arrays to instantiate the rewards onto the map |