HW3 Code Review

Salman Ayaz, Chen Min

1. **Code Smell Identified:** code duplication
   Git Commit: dbb6032e329572e43180c8c892784fd69c5ee587

   When trying to extract a singular sprite out of a sprite sheet, we would have to write a full while loop to iterate over the image and extract each subimage. Instead we created a new method in the Entity class that extends to all subclasses, This method allows us to skip the while loop and extract the sprite subimages with a single method call

2. **Code Smell Identified:** dead code
   Git Commit: 9ad8c32dc8ca2bd55a5adc9243753c96db60e2cd

   drawMovementDirection(), debugRender(), drawPositionDot() were debugging methods we created while developing our game. Now that the game is complete, these methods are no longer being used so we decided to remove them.

3. **Code Smell Identified:** lack of documentation
   Git Commit: 39c939cbd6092feee4e2b74657099e21e48e6a59

   We had separate visible and interactable boolean variables but not enough information as to why. We added more comments to clarify the reason we needed both.

4. **Code Smell Identified:** lack of documentation
   Git Commit: 61eb9c0f28b54a0c3278872439aa493cbcf0b103

   The render method in the entity class had a lot of different scaling variables. So we decided to add comments to clarify what each variable does.

5. **Code Smell Identified:** code duplication
   Git Commit: 8a7e6115d1b2e285d33bb13e263c630d4ff20e86

   We had multiple methods in the Pathfinding class that would check to see if the method parameters were within an array index. These checks were duplicated across multiple methods, so we refactored the check into a new method that all the other methods call.

6. **Code Smell Identified:** unused or useless variables
   Git Commit: b617225aed48bd40e8a6d549613c22bef2c8f4ed

   Remove variables that were forgotten in the Inanimate class after deciding to refactor Inanimate entities into multiple classes instead of one class that would have its behaviour change depending on constructor parameters.

7. **Code Smell Identified:** code duplication
   Git Commits: 2fd28fd1c5e3c47657e4ac527bee47f8fbcfe4a3, dafe0e8685afffa92966808914fcdcd5bcdaddb9

   When changing hitboxWidth or hitboxHeight variables, it would break the entity's position, so setPosX() and setPosY() had to always be called right after. This would create a lot of duplicated code. We Instead made a function to change hitboxWidth and hitboxHeight that also updated the player position.

8. **Code Smell Identified:** bad variable name
   Git Commit: 99859669420b466fe73f8d3cb57985f908400905

   The constructor parameters for classes in the entities package all had posX and posY as methods in their parameters which was not the same as the value returned from getPosX() and getPosY(). The getters would return the point in the middle of the entity, while the constructor parameters would center the Entity in the middle of that tile. For example, setting the constructor posX and posY to 1 then calling getPosX() and getPosY() would return 1.5.