IEnemy
        Wizard.cs
        WizardFire.cs

LevelLoading
        RoomTextureFactory.cs
        LoadRoom()
        RenderRoom()
Collisions
        EnemyCollisionHandler.cs

_____

Code Review

Author: Kristin Bradshaw
Date: 3/14/2021
Sprint: 3

Wizard.cs

Quality: Maybe some magic numbers can be added to the LoZHelpers file? Other than that, everything else looks fine. Just another enemy.

Readability: Maybe move the Knockback() method at the end of the file and have the Draw() and Update() in consecutive order. Just a style choice so not that big of a deal.

WizardFire.cs

Quality: Pretty straightforward. Just another IEnemy with specific properties.
Readability: Nothing looks too crazy. Nothing to change here.

RoomTextureFactory.cs

Quality: Looks like the other Sprite Factories. Nothing to complain about here.

Readability: Looks good.

LoadRoom()

Quality: Magic numbers can be added to the LoZHelpers file.

Readability: The switch cases take up a lot of lines. Don't really know how that can be refactored to shorten it. I think in the code review the professor said something about maybe just making a list of AddIObjects/AddIEnemy and accessing them by index depending on which number is called in the .csv file. That's a lot of refactoring so maybe we can save this for the next Sprint if it actually works.

RenderRoom() - Renamed to Draw()
(I don't see this file? Idk if I'm blind)
Quality:
Readability:

EnemyCollisionHandler.cs

Quality: Looks like every case that we need for this sprint is handled here. Is there a specific reason why you made the namespace LegendOfZeldaClone.Colisions instead of just LegendOfZeldaClone? I noticed you did it for IEnemy last sprint, but I don't know how important it is to have all the files have the same namespace. If it isn't important, disregard my concern.

Readability: Rectangle lines are long because of parameter length but it's fine. Not important enough to change since it's just a style thing. Other than that it's fine. I can't remember if it's ok to leave comments or not. I personally don't mind them but I feel like the professor said our code should be "good enough" to not need comments. Up to you.

_____

Minutes: 40
Hypothetical Change: Possibly put some of those magic numbers in the LoZHelper file and check to see if leaving comments is fine. The switch cases in Room.cs would probably be the biggest change I'd suggest for the next Sprint.

_____

Code Review
Author: Yonace Ghebrehiwet
Date 3/14/2021
Sprint 3

Wizard.cs
Quality: Code quality looks fine
Readability: Knockback force is a little hard to understand but it works properly in the game.

WizardFire.cs
Check out the maxHealth variable. I don't think it's used anywhere and it's privated.
Quality: Code quality is good similar to wizard.cs
Readability: Can understand everything that is going on.

RoomTextureFactory.cs
A standard texture factory not much to comment on here.

LoadRoom.cs
Parser works well. Need to find a way to reduce switch cases (I don't think it's possible).
Quality: Code quality looks good.
Readability: Readability is good.

EnemyCollisionHandler.cs

Quality: Code quality looks fine
Readability: My suggestion here would be to group and move the empty methods at the bottom and remove comments (*if required) . Other than that readability is fine.

Time spent: Approx 20mins