**Part 3: Questions**

**What limitations do you think your game engine would have? Try to**

**explain what these limitations are.**

**I.**

We have one main game loop as you can see in Fig.1. It only processes one frame at a time and our software is hard-coded. There are two limitations in our game engine: loop and software.

a.

The simple loop, in Fig.1, limits our engine because it only follows one direct path. This path is what sets our engine to a single frequency. This means that everything that you see on the screen is drawn simultaneously. Therefore, if wanted to introduce a new actor, like a new alien, who’s animation would have to be two times faster, we couldn’t. This is because in order to make the animation run faster it needs to be updated and drawn much faster than the other actors. But since all the actors follow one frequency we wouldn’t be able to two have two different animations playing at a difference pace.

b.

Our software is hard coded. If wanted to introduce small changes to the game that might be used, we need to write it in the main source instead of an external source. Since our game runs on one loop, it would read accumulated code over and over again, consuming more time and memory. Our game engine works best with simple games. But if it’s used for more complicated games it will become sloppy and take more time to update and render, it lacks adap

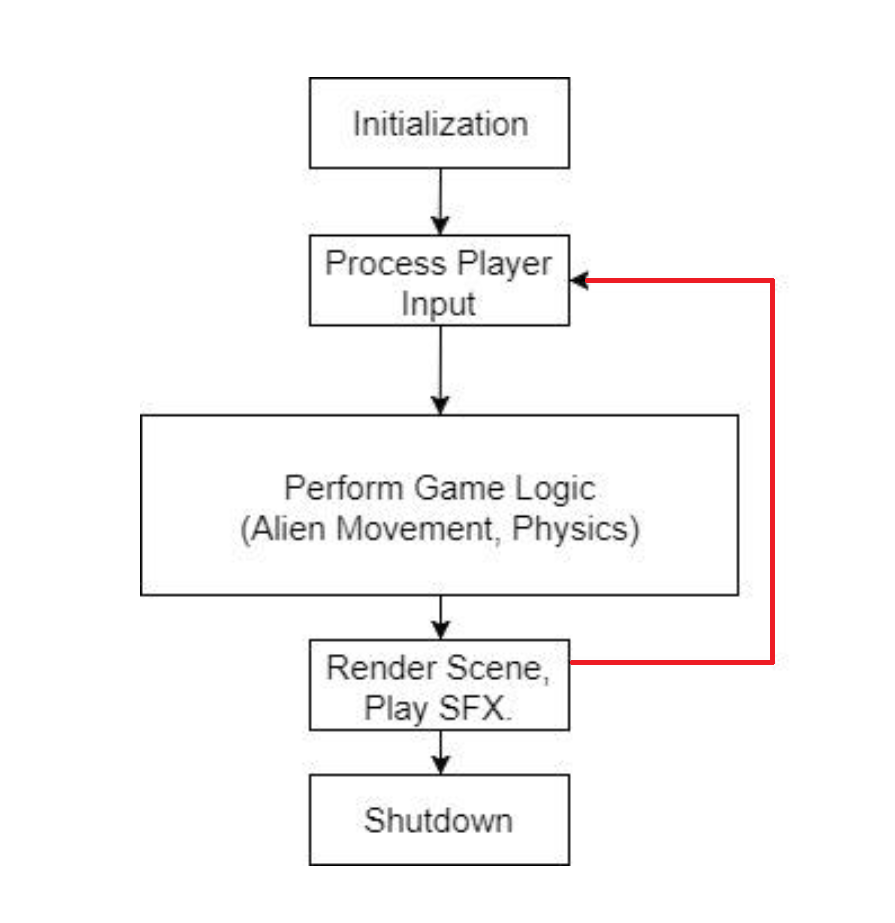


Fig 1.

**II.**

**How can you adapt your game engine in order to accept a different  
game genre? You may use diagrams, UML, etc. to help illustrate your  
example. (if your game genre was FPS pick a different genre for  
instance RPG and describe how your engine would be different).  
(Approx. 200-300 words)**

*Alien Invasion* is a retro style shooting game like the original *Space Invaders* arcade game that was released in 1978. Owing to the fact that our game engine uses a hard-coded simple main game loop, adapting it to other simple retro games should be doable without completely rebuilding the engine. For example, if we would want to adapt the game engine to a platformer like *Super Mario Bros,* we could easily do this within the limitations of our simple main game loop because enemy AI would not need to update at a different interval than once per frame.

In further detail, we can adapt our game engine to the platform game genre by changing our player and enemy classes to correspond with the requirements of a platform game. An example of this would be changing the player and enemy classes to correspond with platformer gameplay mechanics, such as appropriate enemy AI or pathfinding, and allowing the player to perform actions such as jumping and picking up items.

In addition to altering pre-existing classes, we would need to add new game classes for game objects that are not within *Alien Invasion*, such as pick-up items (coins and power ups). All of the above would still allow us to utilize our engine architecture without major overhauling.

