Program Overview:

A megaman/mario style game where the player is placed in a premade level and has to fight enemies. The goal of the player is to survive as long as possible without having any parts of their body touched by the enemy. The player can kill enemies by shooting them or jumping on their heads.

List of bugs:

* there is no win conditions for the game so the later levels cannot be unlocked yet
* Very rarely, when you jump on the enemies, occasionally you’ll also die even though jumping on them is supposed to kill them
* You jump lower when moving left and right
* You can walk through spikes like as if they aren’t there
* You can potentially live forever by just constantly jumping on the enemies’ heads

Notes to future programmers:

* Give the game win conditions
* Add more obstacles to the game because for right now, the game is too easy to survive
* Improve enemy AI so they don’t just act like zombies
* Make the game more aesthetically appealing

Scratches:

1. Game Engine

* The base for the game created by Don last semester
* Contains the code that imports the map, creates the player with animations, allows player movement, plays the background music, and handles all the collision detection between the player and the ground

1. Enemy Scratch

* Creates enemies who spawn at a fixed point that can follow the player around

1. Movement Scratch

* Implements touch controls for the game without using the d-pad

1. Spawner Scratch

* Makes enemy spawning more allowing custom set properties for the enemy spawn points straight onto the tiled maps

1. Firing

* Allowing player to shoot bullets that cleans themselves out of the game when in contact with world objects

1. BulletVsEnemy

* Combines the enemies and player shooting into one project which then became the base for future collision detection logic

1. Stomper Scratch

* Allowed player to jump on the enemies and kill them. It differentiates the player and enemy fixtures that collide in different ways with each other

1. Inheritance Scratch

* Using interfaces and abstract classes to make the function of a class more obvious, code more concise, less duplication of code, easier to manage, and less like “This class does everything”
* Giant entity list that makes updating and rendering everything a breeze

1. Raycasting

* Handles slope collisions by detecting angled surfaces with raycasting

1. Screen Scratch

* Switching between screens with a new main and options menu. In the options menu the player can select a skin and which level he/she wants to play. These levels are locked until the player completes the level prior to the selected level.

Table of Contents:

|  |  |
| --- | --- |
| Program | Documents |
| User Manual | core/docs |
| WIP 1 | core/docs |
| WIP 2 | core/docs |
| WIP 3 | core/docs |
| WIP 4 | core/docs |
| WIP 5 | core/docs |
| Sources | core/docs |
| Program Overview | core/docs |
| Lessons | core/docs |