Wednesday, November 9, 2016

ThunderPocketEdition

Description

- Thunder pocket edition is a third person shooting game. The player controls an airplane named Thunder, which can fly up and down the screen and shoot lasers. The player's main goal is to avoid obstacles and attack enemies. Enemies are red diamonds that can be killed by the lasers, where as the obstacles are green blocks that are not shootable. The user will use keyboard to play the game. W and S controls whether the airplane flies up or down, and when space is pressed, the airplane shoots a laser. When the airplane collides with an obstacle or an enemy, the screen goes black and will not start the next round until the reset button R is pressed.

Technologies from CSC258

- Flexible Rate divider to control the rate of obstacles and enemy appearance.
- Finite State Machine to control the laser. (appear, move to right, collide w/ obstacle , reach the end)
- Finite State Machine to control the Obstacle. (appear, move to left, collide w/ plane, reach the end)
- Finite State Machine to control the Enemies. (appear, move to left, collide w/ plane, collide w/ laser, reach the end)
- Finite State Machine to control the plane. (move up, move down, killed)
- Random generators to generate obstacles and enemies.
- Data-path for drawing all the objects

Milestone I

- come up with a data path that shifts but not bounce (data path: shifter + register, erase and draw)
- implement plane control and data path (first implement 4 * 4 square and make it move up and down. control take go up or down regular inputs as inputs and data path decides how to go up and down according to previous work)
- Learn about Random data generator (which we thought was hard but it was not so we already finished it)

Milestone II

- Make the plane plane-shaped (change the counter of x and y in data path so the pixels drawn on the screen for each row and column will change so the shape will change)
- implement obstacle control and data path (randomly generate obstacles on top, middle or bottom)
- Implement enemy control and data path (diamond shaped, modify x and y counters in data path to achieve, erase when hit by laser)
- Collision detector

Milestone III

- Implement Laser control and data path (pass through obstacles and enemies, straight line through the screen)
- Final combination, combine all the modules and connect control to keyboard