Mack Tang
Mar 22th, 2018
Programming Projects
Project (Homework 4)

Link to Version Control: https://github.com/macktang/AtomJumper.git

Planned for Today's Assignment (copy & pasted from last week):

Wk #	Required Learning	Task	Due Date and Time Allotted	Deliverables for Due Date
3,4		Game working		Code for final game
5		Unintelligent bot 1 moves working (moves left or right randomly)	Mar 21 (HW 4) [3 weeks]	Including unintelligent bot moving

I would not like to make any changes to the above plan, but here are some more detailed requirements:

Task: Game Working
-Make gates infinitely spawn (instead of only spawning 2)
-Add score on-screen
-Add menu page
-Add sound
-Tune up player controls (make jump bigger, remove up arrow key "center jump")
-Make player die if they hit gates

-Add Testing Framework

Task: Unintelligent bot 1 moves working
-Avoids jumping if jumping will make player sprite collide with an obstacle

Actual Accomplished for Today's Assignment:

Added the following code Features:

-Made gates spawn infinitely, it does this by storing the Last Generated Gate's Height. Whenever the player height exceeds the height of the Last Generated Gate, they must have passed it, and a new gate is generated. Finally, the Last Generated Gate Height is updated.

-Added score on-screen (score increments for making a gate disappear at bottom of screen)
-Added menu page (press key for human play, click mouse button to watch bot to play)
-Added sound (+1 score sound plays, explosion sound if player touches an obstacle)
-Tuned player controls (Added more gravity, made player left/right jump greater)
-Made player die upon falling off screen or collision with obstacle. Player death restarts the game and respawns the player, and resets score = 0.
-Made a first bot, the bot adjusts its position left or right based on its position with respect to the center of the latest generated gate.

Additional code features:
-Made 2 additional brick/square obstacles appear above and below every gate.
-Changed player's appearance to a small square instead of a stick figure, updated colors of game to match the original "Amazing Brick" mobile game
-Updated instructions and formatting of .md file
-Moved images to its own folder

Things that were not accomplished:
-Add Testing Framework
-The bot is different than I originally stated it would be, it does not does not "avoid jumping if jumping will collide with an obstacle". Instead I made the bot jump left or right based on the center of the last generated gate. I think the original bot would be much harder to implement and would require it to think ahead a few jumps.

More about this Week's Assignment:

Discussion about why I think the Bot fails:
The bot is able to move left or right to center itself on the approaching gate. However, I think the bot fails because it jumps too much. It is not able to move left or right fast enough compared to how fast it moves upwards. So, even though it tries to aim for the center of the gate, it is simply moving upwards way too fast to adjust in time.

I tried to limit the bot's upward movement by jumping only when it had fallen to a certain height. This successfully limited its upward movement. However, the bot then started to fail because it could not move left or right fast enough.

In both cases, the bot ignored the positions of the smaller square brick obstacles (it only focused on the gates). It would be difficult to take into account the bricks and the gates simultaneously.

For the assignment, I left the code in the state described in the first paragraph of this section

6	Evolving Neural Nets	Machine learning tutorials, Unintelligent bot 2 moves working (moves left or right depending on gate location)		Code for machine learning tutorials, code for unintelligent bot 2 playing game
7	Evolving Neural Nets	train bot, bot plays game	Apr 4 (FP 1) [2 weeks]	Code for first trained revision of bot playing game

(bot aims for center of gates but moves upwards too quickly to adjust left or right in time).

Planned for Next Assignment:
For this week, I already made an unintelligent bot that moves left or right depending on gate location. For next week, I would like to find some way to upgrade the unintelligent bot to increase its score.

Things to add:
-Add Testing Framework

Other than these 2 changes, I would like to keep everything else the same.

Screenshots of Project