Gene Rhee, Vignesh Kuppusamy, Guillaume Bailey

Stealth Basketball

Lab 7

Game Design

**Introduction:**

Stealth Basketball is an absurd misuse of basketball terms and concepts in a Metal Gear Solid or Splinter Cell-styled top-down stealth game. Stealth Basketball is a time-based action game on a grid. The player will traverse a 2-D grid with certain elements being restricted to the level’s rhythm or tempo.

**Prototype:**

Setup

Our prototype is one level. The player’s sprite is Mario who starts at the bottom-right corner of the map. The guard is located at the upper-left corner of the map. The coin sitting on top of Mario is our placeholder basketball. There are walls that impede movement for both guards and the player. The game currently has no win/fail state but the player can traverse the grid and if bored can trigger the guard by being within 4 grids of him and then try to run away.

Time

The prototype currently executes actions every 1 second. Every 1 second, the guard will either do nothing or move 1 tile based on his AI, and the player will move in a direction of his/her choice.

Controls

The game supports the directional keys for player movement. The UP key will move the player up on the grid, DOWN will move the player down, LEFT will move the player left, and RIGHT will move the player right.

Grid

A high-level view of our game will look like this:

+---+--+---+--+---+

| P | | | |W|

+-------------------+

|W|W|W| |W|

+-------------------+

| | G| | | |

+---+--+---+--+---+

Where P is the player, W are walls, and G are guards. Each object has a *GridSprite* associated with it. The *GridManager* will have a 2-D array which represents the game grid and holds *GridSprites* based on their position on the grid. Both player and guard movements are tweened to make a smooth movement between grid spaces.

There is collision detection in the game. The player cannot leave the grid and any input that results in the player leaving the grid will just end in the player not moving at all. This wall detection expands to in-grid walls as well which are just represented as grid edges with thicker lines. Guards will try to approach the player under certain conditions but will not be able to overlap grid spaces with the player.

AI

Guards have an AI. They are set to stay still until the player gets 4 units (might change before we submit) away from the guard in which case will trigger him. Once triggered, the guard will advance towards the player using an A\* algorithm to find an efficient path towards the player. Right now, the guards do not do anything once they collide with the player, but we plan to include functionality in the future.