**15-112: Term Project Proposal**

“Herding game” by Angela Liu (amliu)

**Problem:** High demand of free games

**Project purpose:** Create an engaging, fun game for all ages (1-2 players)

**Description of ideal result:**

The player controls a “herder” using the arrow keys or ‘wasd’ keys on the keyboard. Meanwhile, animals wander around the board. It is the player’s job to pick up these animals and return them to a safe pen. However, a monster wanders the board as well, and it is capable of eating the animals and the player. The game has no levels – instead, it ends once time runs out.

An easily seen timer displays the time left per game. It constantly ticks down. Successfully retrieving an animal will refill the timer by a certain increment, thus extending the game and rewarding constant activity. However, if an animal is eaten by the monster, the timer is decreased by a time penalty. Being eaten by the monster yourself also punishes the player with an even greater time penalty.

Points are scored by returning animals to the pen. Points are lost when an animal is eaten by the monster. Points cannot go negative.

The monster seeks out the closest animal to attack. It can also seek out the player, but only does so if the player is currently carrying an animal. The player, if not carrying an animal, can fight back against the monster with an attack that will stun the monster for a given amount of time.

Animals wander randomly across the board. However, if an animal is within a certain radius of the monster, it will run in the opposite direction of the monster. In order of speed, animals are slowest, then the monster, and then the player. In addition, there will be a semi-constant number of animals on the board at all times. When the player retrieves an animal, a few seconds later, a new animal will spawn to take its place. As the player accumulates more points, the number of animals on screen may be bumped up to increase challenge. The monster may also move slightly faster as well, or more monsters may be added.

Powerups spawn randomly on the board. The player can carry up to three powerups. Powerups can provide a temporary speed boost to the player, freeze the monster in its tracks, add additional time to the timer, add extra points, and more.

The point of the game is to score the most points. In order to do so, the player must play strategically in order to conserve/extend play time as much as possible.

**Methods:**

Pygame will be used to update the board, manage sound effects, import images, sense moves from the human players, and set timed events. As I explore the other capabilities of Pygame, I will try to include them in place of methods implemented currently.

Classes will play an important role. Players, animals, powerups, and monsters are each their own type of object. Objects store information such as current place on board, unique sounds and images, move speed, and more.

Other classes will not create objects and instead be involved in different aspects of the game, such as moving objects, picking up objects, and more. In the future, I hope to also implement a class to help facilitate sprite animation.

The main function will call these classes and run the entire program.