Team Dad Jeans
Maryann Foley, Shruthi Venkata, Jessica Wu

A Day in the Dog Park

Outline

You are a taking a stroll around your local dog park, walking your new “default” dog when see a dog walker accidentally set all his/her dogs loose! Their owners have gathered at the local pet shelter, hoping that their lost pets will eventually be found and brought in safe. You must go on a quest to collect and bring the dogs back to their respective owners, on the way defeating enemies, including:

- A ball (while you are still at the dog park)
- A squirrel
- A bird
- A cat
- The final boss: a dog catcher!

The “fighting” will consist of bite, charm, bark, and pounce. The effectiveness of each of the moves will be determined by the type of dog one chooses at the beginning of the game; there will be a few different dog breeds with different agility, loudness, power, and cuteness levels. The enemy can also scare the dog and if the dog gets scared enough it loses the fight. Choosing which dog to use during a fight requires strategic thinking. With each turn, you have the option of using your most agile, loud, powerful, or cute, dog next, or you could just continue using your current dog. Certain traits would cause more damage on certain enemies.

Implementation

This will be written in processing so that we can have the graphics of the dog and enemies. We hope to be able to draw them outside then import the images of each character.

Goals

MVP:

- Working battle simulation
- Dog selection
- Heap algorithm to sort dogs by specific traits

Reach:

- Story w/ graphics
- Map w/ correct and incorrect paths; incorrect paths can lead to changed stats
  - Ex. An incorrect path may lead into a muddy field, lowering cuteness

UML

Class DogPark
+ Stack<Enemy> Enemies + Dog player + Int gameStatus //0 if not started, 1 if started, 2 if over

- displayStats() //will be updated with draw()
- defeatedBoss() //will pop the top boss off, moving on to next
- enemyAttack()

Team Dad Jeans
Maryann Foley, Shruthi Venkata, Jessica Wu

Abstract Class Enemy
+ Int HP + Final Int BASEHP + Final Int BASECUTE -Int withstoodCuteness + String name
+ Boolean isDefeated() + Abstract ModifyHP(int damage) + Abstract modifyCute(int charmed) + returnHP() + returnCute() + abstract attack() + Abstract winMessage() + Abstract loseMessage()

Class Ball
+ ModifyHP(int damage) + modifyCute(int charmed) + attack() + winMessage() + loseMessage()

Class Bird
+ ModifyHP(int damage) + modifyCute(int charmed) + attack() + winMessage() + loseMessage()

Class Squirrel
+ ModifyHP(int damage) + modifyCute(int charmed) + attack() + winMessage() + loseMessage()

Class Cat

Team Dad Jeans
Maryann Foley, Shruthi Venkata, Jessica Wu

+ ModifyHP(int damage) + modifyCute(int charmed) + attack() + winMessage() + loseMessage()

Class DogCatcher
+ ModifyHP(int damage) + modifyCute(int charmed) + attack() + winMessage() + loseMessage()

Abstract Class Dog
+ Int cuteness + Int agility + Int loudness + Int power + Int scared + String name
+ Boolean isDefeated + ModifyCuteness() + modifyAgility() + modifyLoudness() + modifyPower() + modifyScared() + bite() + bark() + charm() + pounce() + Abstract scaredMessage()

Class GoldenRetriver
+ scaredMessage()

Class Pomeranian
+ scaredMessage()

Class ChocolateLab
scaredMessage()

Team Dad Jeans
Maryann Foley, Shruthi Venkata, Jessica Wu

Class Samoyed
scaredMessage()

Alternate projects:
2D racing game
2 player chess game
Checkers, Chinese checkers, and Othello arcade