Tank : Actor

* Has a shape/text
* Has a position
* Has a velocity
* Has a color
* Has bullets

Bullet(s) : Actor

* Has a shape/text
* Has a position
* Has a velocity
* Has a color
* Has multiple bullets?
* Returns list of bullets and their respective properties?
* Has a bounce method

Wall : Actor

* Has a shape/text
* Has a position
* Has a velocity
* Has a color
* Returns the list of walls in that level?

CollideBorder (Handle/update

* Checks to see if the bullet has hit the border
* Changes the bullet’s velocity (by calling a bullet bounce method?)
* Changes the bullet’s text
* Public bool for number of bounces

CollideTank (Handle/user)

* Checks to see if the bullet has hit the opposing tank
* Changes the bullet’s location to (0,0)
* Changes the bullet’s velocity to (0,0)
* Changes the level??

CollideWall (Handle/User)

* Checks to see if the bullet has hit a wall
* Changes the bullet’s velocity (by calling a bullet bounce method?)
* Changes the bullet’s text
* Public bool for if the bullet has already bounced off a wall/border??
  + How do I get Border and Wall to have some collision bool

Score

* Add points when P1’s bullet hits P2’s tank
* Display points at all times?
* CHECK FORMAT OPTIONS

Lives

* Subtract lives if P1’s bullet hits P2’s tank
* Display lives at all times?
* CHECK FORMAT OPTIONS

Constants

* Contains all the constants for the game
  + Screen size
  + Cell size
  + Frame rate
  + Colors
  + Default lives
  + Lives, level, score formats

Levels

* Creates each level in the game
* Can be added to cast and then each wall is part of the overall level list

Program

* Creates all of the cast (tanks, bullets, scores, lives, levels, services, scripting)
* Calls the StartGame() method in Director class?

MoveBulletAction

* Assigns position to bullet using tank’s current location plus their velocity
* Assigns velocity to bullet using keys

ControlTankAction

* Assigns velocity to tank using keys

