**Things to do/Notes**

**Ship class**

Need:

* The ship can still fire even if isAlive = false

Future:

* Make a method which takes in bool menu.isMultiplayer and initializes either one or two ships (inside Game:: initializeGame)
* Make Structure of keystrokes for each ship

For readability:

* Make friendlyFire a ship class member variable
* Make a method (Ship::update) which packages all of the switch cases for ship movements, checks for friendly fire, and updates the position of the ship
* Make a method (Ship::draw) which draws all ships and all missiles

Goal/stretch:

* Update the ship rendering to a ufo or airplane shape
* Allow the ship to rotate about its center or change the shooting angle

**Asteroid**

* I made some changes to how the manager class is initialized
* Currently “difficulty” (speed of the asteroids) and “level” (number of shots to break up) are tied to each other – in our original description these were two separate things

**Sound**

* I refactored the mml stuff into the method ‘updateStream’
* I still have no idea why, but on my computer the blaster.wav, explosion.wav, and rocket\_thruster.wav all give “Warning: WAVEfmt not found” and do not make any sound. When I change them to other .wav files I’ve used before in homeworks, the files load perfectly file.
* The background sound stops after a few seconds (I’m not sure if that is the current implementation or if its my computer)

**Menu**

For readability:

* Could you make the difficultly levels integers instead of chars (int’s is what the asteroid class is expecting)? (then delete convertDifficultyCharToInt) \*\* actually this depends on how we want to implement level v difficulty (see asteroid comments)

**Sound**

For readability:

* Can lines 18 – 31 in SoundManagerTestMain be packaged into a method in SoundManager? \*Complete\*
* Can lines 67 – 80 in SoundManagerTestMain be packaged into a method in SoundManager? \*Complete\*

**Summary**

Need:

* Needs guard clauses -> can specify level that is not initialized methods

Future:

* Visual representation on window

**Master game**

Future:

* Level up game if all asteroids are destroyed