

## CSCI 150 – Sample Final Project Proposal

**Name:** Lars Seme

I am writing a simple, one-player graphics based space battle game. The user will control a ship which will fire at randomly generated targets. The user gets a point for each target destroyed.

### **main()**

Sets up the screen with random star field

Plays background music

Calls function to draw *Enterprise*

while loop controls action

### **Timeline:**

Design static *Enterprise* (11/18)

Add ability to pivot (11/20)

Add ability to move forward and then “coast” (12/2)

Add in firing ability (12/4)

Create target generator (12/5)

Determine whether torpedoes hit target and remove both from game (12/8)

Add random starfield (12/10)

Add music/sound effects (12/12) – if I have time

### **How does this go beyond/extend what we have done?**

The game makes extensive use of classes – each of the ship itself, the torpedoes, and the targets are objects in the game, and they need to be able to interact with each other. I am combining ideas from multiple labs (Faces, Water Jugs, for example) and from lecture, along with learning how to get pygame to import and use music/sound effects. Finally, there is a significant amount of mathematics necessary to control the movement of the ship itself.

### **Classes, Methods, and Functions**

There will be three classes, each with their own methods:

Ship – will contain the coordinates for the center of the ship, its rotation angle, and its speed.

Methods: rotate(angle) will rotate the ship by a given angle

update\_speed() will increase/decrease the ship's speed

draw() will draw the ship

update() will update the location of the ship

Torpedo – will contain the center of each torpedo when fired, at set velocity at 5 pixels per update

update\_torp() will find the new location of the torpedo for each frame

draw() will draw each torpedo

Target – the round targets, placed at random spots by the user – they have a location and radius, elected randomly

draw() – drawing them

We also have a function which populates the random starfield (since the stars don't move or change, I elected to not objectify them)

The function *hit* will attempt to determine whether or not a torpedo has hit a target, and if so, remove both the target and torpedo from the game. It returns a boolean so we can know whether or not to play the explosion sound.