Henry Adams

11

Block 7

Bullet Project

My project is now finished. It has the same idea and goal of the original, with some changes. Meteors will fall out of the sky periodically, that must be avoided or shot down. You control a space ship. The rocket ship is still controlled by the WASD keys. Holding the space key will launch a pattern of bullets that damage the meteor. Holding shift will slow down your space ship for more precise movements, and if you fire while holding shift, you will get a very accurate and powerful laser beam. After about 30 seconds, more stronger and faster meteors will start to spawn. You lose a life by getting hit by a meteor, and you have 5.

On the right side of the screen, you will see your stats and the controls. You will see your time, score (+5 for every bullet hit, +50 additional for every destroyed), meteors destroyed, and lives you have. Controls will also show.

The class that creates the Ship object is based off the Block object, except it doesn’t use the bounce method, has an image over the actual “hitbox” or block, and other modifications. The bullets that the ship fires are based off the Dot object, that has a constant Y speed when fired. The green bullets will always move up by 5 units until it hits a meteor or hits y = 0. The laser beam is actually a straight line of Dot objects, except has a much faster speed, has a rectangle printed over the dots, and follows the Ship’s x and y coordinates.

To track each meteors, ship, and bullets(dot) object, i use ArrayLists to traverse through each existing object, and checks if they collide or reach top/bottom of the window.

The driver class (FinalProject22) uses the keyAdapter package to record the WASD keys, shift, and space. It also technically uses the mouseAdapter package but isn’t used or modified other than when the mouse exits or enters the window.  
  
The Ship and Background of the driver class use images. For the background, the background slowly moves downwards, instead of being static/still image. The background is actually the “space” image that has another space image on top of it, and scrolls down. When the first image reaches Y = 0, it sets its Y back to the top, and it repeats. When the meteors get harder (after 25 seconds of starting the game), the background also speeds up.

The meteor class is also based off the block object, but prints out an oval with health. It also falls, has a random position spawning, and random size. Health is scaled by size. It also checks for collisions with bullets (projectile.java, based off of dot.java), and the ship.

For the timer you see on the right, I just used the game timer to run everything that increments a variable every tick. Everytime it reaches the equivalent of a second, it sets that variable back to 0, but the minute variable back to 1. This repeats, and the information is printed on the right.

For the collision checks, it uses the distance formula to calculate every point within the area of the object, and sees if one object is inside the other’s range. Based off the one from our notes.

The timer stops when you lose all lives, and prompts “You died!”. When you destroy 50 meteors, it prompts “You win!”.

Old goal:

My goal for this project is to create a Rocket Ship that can be controlled by the WASD keys (Same as arrow keys). While the “Space Bar” is pressed, the rocket will fire a volley of bullets. From the top of the screen, “meteors” will fall from the sky that the ship has to avoid or shoot down. Periodically, an enemy object will spawn that can fire back. Destroying it can drop a “powerup” that can either multiply the fire-rate or damage. You gain points as you destroy meteors or enemies, and if you get hit 3 times, you lose. Additionally, holding the “Shift” key will slow down the speed of the ship for more precise movements or shots. On the right-hand side of the screen, the score, amount of lives, kills, and time will be shown next to the game (similar to the Graphic Design Project).