**ASTEROIDS**

Abstract Design

**Overview**

Asteroids is a game about controlling a space ship and shooting asteroid. The goal of the game is to shoot and destroy all the asteroids in the level. Once this is done, level is complete and you will move onto the next level.

**Basic Rules**

The player controls a ship in game. The ship allows the player to use its thrusters to accelerate in a direction, turn left or right, and shoot. The ship itself will not move, instead, the game will use the movement of asteroids to create the illusion of movement.

When the player starts a level, asteroids will spawn inside the level, in which the player must shoot and destroy to complete the level. If the player’s ship collides with any of the asteroids, the ship will blow up and the player will lose a life. Asteroids will move randomly in any direction when they first initially spawn in the level. Asteroids will also come in different sizes, which includes large, medium, and small asteroids. Each of these different sizes of asteroids differ in their speed and their health. Larger asteroids move slower and are harder to destroy while smaller ones are much faster and may take only one bullet to destroy. When the larger asteroids are destroyed, they break apart into smaller ones. The small asteroids, however, will be completely removed when destroyed.

The game will have a lives system that will be tracked throughout the game. If the player’s ship gets destroyed by an asteroid, the player loses a life, and will respawn in the level with temporary invincibility if lives are left over. If the player dies and has 0 lives left, the player gets a game over.

The game will be level based, each level getting progressively harder as the player moves on. For the player to move onto the next level, the player must destroy all asteroids in the level. If the player loses all lives and gets a game over, the player is reset to the first level.

**Objects**

**Ship**

The ship is what the player controls and represents the player itself. Player can control the ship using keyboard inputs. W will thrust the ship forward, A will turn the ship to the left and D will turn the ship to the right. Spacebar fires the ship’s weapons, which will be used to destroy asteroids. The ship has a limited number of lives, which when it reaches zero, it is game over and the game is reset to the first level. When the ship collides with any asteroids, the ship will explode and the player will lose a life.

Ship movement will be an illusion. When the game begins, the ship will be placed in the center of the room and it will remain there. When thrusters are activated, the ship itself will not accelerate, rather, the asteroids will move in a direction opposite the direction of the ship’s direction while still maintaining its current direction and speed.

Ship object will have the step and draw events. The step event will constantly check whether the ship has crashed. If it has, the ship will explode, the instance will be removed and a life will be subtracted. The draw event will draw the flame animation at the back of the ship when the ship is thrusting (When the W key is being pressed). The draw event can also draw other sprites if needed.

**Asteroids**

Asteroids will randomly spawn in a level. The player must destroy all instances of these asteroids to move onto the next level. Asteroids can come in different flavors, such as large, medium and small asteroids. Different kinds can also be added if desired. Large asteroids have more hp, thus takes more hits to destroy, medium asteroids have slightly less and small asteroids only take a single hit to destroy. When the large and medium asteroids are destroyed, they break apart into smaller asteroids. When small asteroids are destroyed, they are removed from the game room.

When asteroids first spawn in the level, they will have random placement, speed, and rotation. There will be set limits to speed and rotation depending on the size of the asteroid. Smaller asteroids will move and rotate faster while bigger asteroids will move and rotate slower. Hit points, however, will be strictly defined for each size of asteroid. Every time a bullet collides with an asteroid, it will lose 1 hp. When hp reaches 0, the asteroid is destroyed and will break apart into smaller asteroids or, in the case of small asteroids, be removed from the game.

How asteroids break apart depends on the size of the asteroid. The breaking apart of asteroids can be handled by the destroy event. Large asteroids will break apart into two small asteroids and a medium asteroid when destroyed. Medium asteroids will only break apart into two small asteroids. Small asteroids do not break apart and are removed when destroyed.

Asteroid’s speed and direction are randomly set when they first spawn in a room. However, their movement will also be affected when the player thrusts the ship forward. When asteroids leave the room boundaries, they wrap around and appear at the opposite end of the room. This way, asteroids are never lost.

**Bullet**

Bullets are a simple object. When the ship fires its weapons, a bullet will be fired. Bullets will travel in the given direction of the ship and when they collide with an asteroid or leave the room, they are destroyed. When a bullet collides specifically with an asteroid, they will subtract 1 hp from the asteroid.

**Controller**

Will control the flow of the game. It will handle lives, spawning of asteroids, and scale difficulty depending on the level. It will also enforce other game rules such as detecting level completion, switching levels, incrementing or decrementing lives, and setting global variables. It will also handle timing in the game.

When the player crashes into an asteroid, the controller will detect it, subtract a life and respawn the ship with short invincibility time. When player lives reach 0, it will be a game over.

Controller will handle spawning in the game. At the start of each level, it will spawn asteroids randomly in the room and spawn the ship. When the ship is destroyed, it’ll respawn the ship with temporary invincibility.

Controller will detect if all asteroids in the room had been destroyed. If so, it will advance the level. Difficulty will scale as the level increments. Each advancement of a level will increase the number of small, medium and large asteroids that will spawn in the next level. It will also decide if special types of obstacles will spawn depending on the level.

**drawGUI**

This object is responsible for drawing GUI elements on the screen. This includes level headers and number of lives left. The main draw event that this object will use is the drawGUI event. This object can draw other elements if needed, but the number of lives and level headers are the only ones planned for now.

**Button**

Button object is simply that, a button. It should be able to detect mouse events for when the user hover’s the cursor over it and clicks. When the button is pressed, the button should simulate being pushed inward, and when the button is released, it runs it’s attached process. Scripts should be able to be attached to buttons as member variables. Buttons should also be able to specify text to be drawn as a string variable.

**Rooms**

There will only be two rooms in Asteroids. The first room is the main menu and the second room will be the game room where the player gets to play. Both these rooms will be the same in its size and camera and viewport properties. The camera will be positioned in the middle of the room and will not move and will be half the size of the actual room. The viewport should be the same size as the camera and be positioned at 0, 0 on the camera.

**Main Menu**

This room will simply be the main menu of the game. It will have the title of the game along with two buttons, start and quit. Pressing start will enter the next room, which is the game room and pressing quit will exit the game. There will be only three instances in this room, including a drawGUI object to draw the title and two buttons objects for start and quit.

**Game Room**

The game room is where the game is played. Here, the ship is positioned directly in the center of the camera and will not move. This room will have a drawGUI object to draw static elements such as lives, level start headers, mission success, or game over. This room will have a controller object to keep track of states, spawn asteroids, and track level progression. Asteroids will be spawned in the room from the controller object and will be placed randomly.

Global Variables

**Lives –** This variable will keep track total lives and will be persistent throughout the game.

**Level –** keeps track of the level number. This will be used to scale difficulty.