Game Title: Defender

Author: Axyl Carefoot-Schulz

Author ID: 201694318

Course: COMP 2659

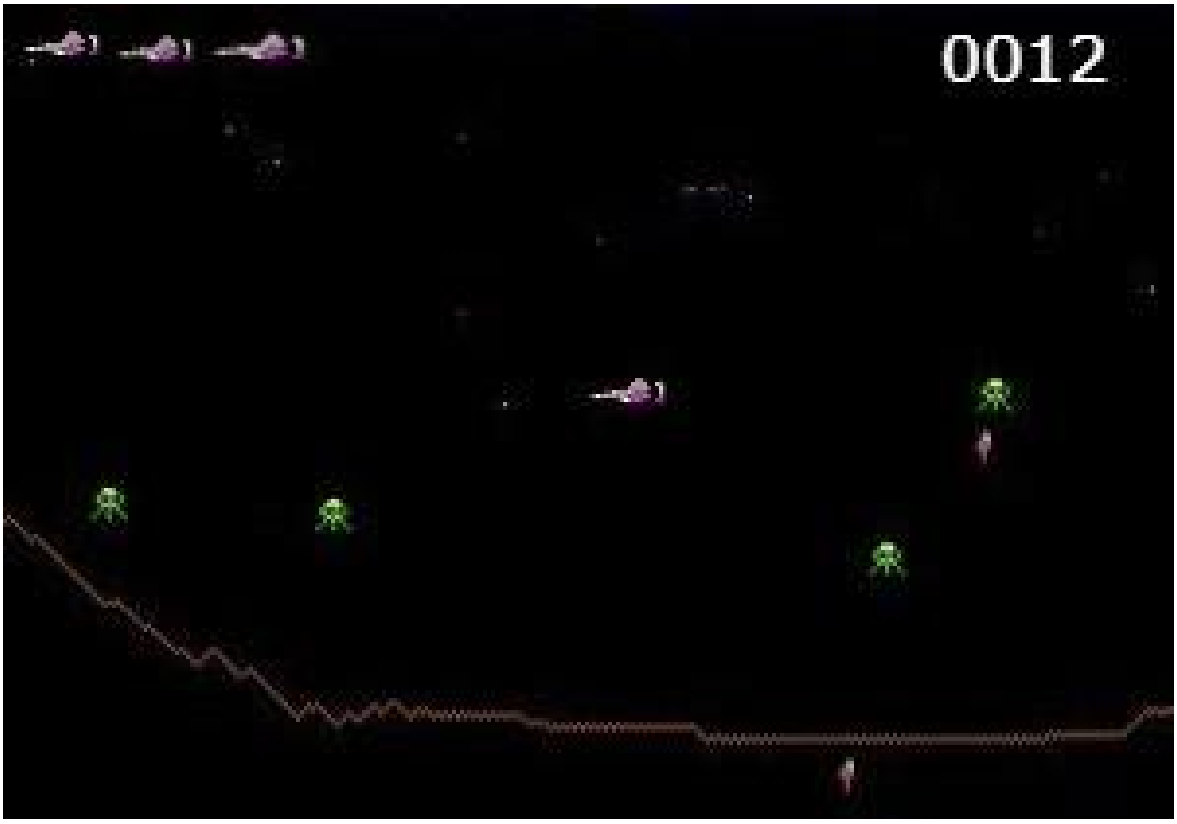
Instructor: Paul Pospisil

Due: January 26, 2024

Overview:

Defender is a 2D side-scrolling space shooter game. The player controls a spaceship using their arrow keys with the objective of defending Earth against alien invaders. The game features both 1-player and 2-player modes. In Defender, you are tasked with flying a spaceship across the face of a planet and shooting at the invading aliens. The aliens will fly towards the player from the edges of the screen. They will also shoot towards the player. The player can fly either left or right, and press space to shoot in the direction that they are facing. The player will appear in the middle of the screen and have the ability to move up and down using the arrow keys. This will physically move the sprite on the screen up and down. However, the left-to-right movement will be emulated by moving all other objects on the screen in the opposite direction to where the player is facing.

The game begins with the spaceship in the center of the screen, ready to engage the incoming alien fleet (shown below).



(cropped from https://gaminghistory101.com/2012/03/21/defender/ and then edited for clarity)

Intro Menu

When the player loads into the game, they will have an intro screen open up. This screen will have the following options: “Singleplayer”, “2-player” and ”Exit”. The singleplayer mode will launch the game into the 1-player mode. The 2-player mode will launch it into the 2-player option starting with

player 1. And exit will close the game. The user can select the different options using the arrow keys and press space to choose an option.

Gameplay - 1-Player Mode:

The 1-player mode will start if the user selects the “Singleplayer” mode. This will place the user centred in the middle of the screen. The user will use the arrow keys to move up and down, or left and right as described in the general overview. The player will begin with three lives, indicated in the top left of the screen. In the top left of the screen there will be 3 copies of the player's spaceship indicating how many lives the player has remaining. On the top right of the screen, there will be a score text. The player's score will increase with every alien ship that is destroyed.

Objective of the game

The main goal of this version of Defender will be to get the highest score possible. The score counter will be in the top right of the screen. Every time an enemy space ship is destroyed the counter will increase. When the player runs out of lives, their score will be displayed.

Ship Movement

The player's ship will have continuous input readings for the arrow keys and will have discrete readings for the space press. This means that you can hold down the arrow keys to move the ship up, down, left or right. But, you will have to press and release the space key to shoot.

Enemy Spawning

The enemies will spawn in a randomized fashion. They will first choose a side randomly. (Left or right). Then a random y coordinate will be generated.
This will be based on the following formula:

player_y + rnd(-10,10)

Where rnd generates a random number from -10 to +10. The x coordinate will be 1.5x the width of the screen, so the aliens are spawned off the map and fly inward toward the player.

An enemy will spawn every 5s to begin. Every 5 points awarded to the player will decrease the spawn timer by 1s. This will cause more enemies to spawn as the game goes on.

Combat

The main form of combat for the player will be to press space and shoot. When the player shoots a bullet will form where the player is and fly in the direction the player is facing. The forming of the bullet

will be the player's x and y positions. The speed will be 3x how fast the player moves, this is the same speed that the environment moves left and right relative to the player.

The bullet fired from the player will either hit an enemy or miss the enemy. If the enemy is hit, both the enemy and the bullet will disappear and a sound will be played. This will also increase the player's score by one.

On the other hand, there are bullets fired by the aliens. These have the same properties as the player bullets but they spawn on the aliens. Aliens will shoot every 3s. The alien will be to the left or right of the player. The bullet will spawn on the alien's current position, then move horizontally towards the player. If the player is hit by the bullet, they will lose a life and become immune to other bullets for 3 seconds. The bullet will despawn, however, the player will remain and a sound will play. The player's life indicator will also decrease by one when this happens.

There is a possibility that the Alien makes it to the player and will collide with the player. In this case, the alien's ship will be destroyed and the same procedure as being shot will apply to the player. (The player will lose a life and become invincible for 3s)

Edge Collisions

There will be a few types of edge collisions. This will be when the bullets of either the player or the alien reach the edge of the screen. This will cause the bullet to disappear. When the player sprite reaches either the top or bottom of the screen with their sprite, they will be stopped and unable to keep moving in that direction.

Objects:

Object Name	Object Properties	Behaviours	Image
Player	Position (int x, int y), size(int height, int width), Score (int) Speed(int)	moves up and down, x position is screen width - half of sprite width to center the player sprite	
Aliens	Position (int x, int y),	*Direction velocity is the	

	size(int height, int width), Direction Velocity(int d) Speed(int)	amount added or subtracted from the x-axis They will constantly move toward the player's ship	
Player Shot	Position (int x, int y), size(int height, int width), Direction Velocity(int d)	Flies in the direction that the player was facing and collides with aliens or out of the screen	Either a pixel or a square mxm grid of grey pixels depending on what looks better.
Alien Shot	Position (int x, int y), size(int height, int width), Direction Velocity(int d)	Flies downward from where the alien that shot it was and collides with the player or the bottom of the screen.	
Player Score	Position (int x, int y) Value (int)	Fixed four digit number that sits in the top right of the screen.	
Player Life	Position (int x, int y) Value (int)	Set of sprites in the top left of the screen that show how many lives the player has.	(Same image as player)

* sprite sizes, screen size and other details are unknown

** Note for all “Direction Velocity(int d)” where the direction the entity is moving is gotten from the sign of the value, and the number of pixels moved per update is in the absolute value of variable d itself.

Asynchronous Events (input)

Event Name	Triggering Input Event	Description

Move left	Left arrow is depressed	Set player flight direction to the left
Move right	Right arrow is depressed	Set player flight direction to the right
Move up	Up arrow is depressed	The ship moves towards the top of the screen
Move down	Down arrow is depressed	The ship moves towards the bottom of the screen
Shoot	Space is depressed	A player shot is created on the position of the ship
Quit Game	Esc key is depressed	The game closes.

Synchronous (timed)

Event Name	Trigger Timing	Description
Move enemies toward the player	Every 1/70th of a second	Check where the player is and move the enemy towards the player
Move Stars	Every 1/70th of a second	Move the stars left and right to emulate the movement of the player's ship.
Move the player shot	Every 1/70th of a second	Move the player bullet in the direction it was shot
Move aliens shot	Every 1/70th of a second	Move aliens shot toward the player.
Generate alien shot	Every 3s	Generate a shot at the location of an alien, and move it horizontally toward the player

Generate Alien	Every 5s	Generate an alien on one of the sides of the player to attack the player.

Conditional Events

Event Name	Condition	Description
Player shot collides with alien	One of the players shot collides with an alien ship. This would be anywhere on the ship and anywhere on the shot.	The alien will be set to inactive and make play a sound. The player score will also increment by 1 and be reflected in the score display.
Player shot out of screen	When the players shot is greater than the screen width + sprite width	This will make the player shot inactive.
Alien shot collides with player	One of the aliens shot collides with a player ship. This would be anywhere on the ship and anywhere on the shot.	The shot will become inactive and make a player hit sound. One of the player lives will be removed and play will resume.
Alien shot out of screen	When the aliens shot is out of screen	This will make the alien shot inactive.
Player runs out of lives	When the player has been hit 3 times (player_lives == 0)	The player will be shown their score. After a confirmation they will be taken to the title screen. Or, in multiplayer then the next player will begin their turn if they have lives left.
Player collides with Alien	When the player sprite and alien sprite collide	This will cause the alien sprite to become inactive and the player to lose a life and become momentarily invincible for 3s.

		(The same procedure as if the player were shot)

Sound Effects:

Sound Name	Description of sound	Event to Trigger Sound
Shooting sound for firing weapons.	pew	Space pressed
Explosion sounds for destroyed aliens.	wub wub	Play shot collides with alien
player's spaceship damage	boom	Alien shot collides with player
Player loses all lives	High to low wrrr	Player loses their last life

Hypothetical gaming session:

The game begins with the player sprite in the middle of the screen flying towards the right-hand side. The player spots an alien flying toward him from the right. So the player will press the UP key and match the vertical level of the alien. After that, they will press space to fire and gain a point. Then they decide to press the left key and explore the other side. There, the player spots an alien floating towards him from the bottom-left-hand side. The player moves in position to shoot but gets hit by the aliens shot first. The player loses his first life. The top-right corner only has 2 player spaceships remaining. The player tries to get back into position to shoot at the alien that had previously killed him but notices an alien coming from the opposite side of the screen towards him. The player focuses on lining up a shot to the alien that had shot him before while dodging the shots from the alien on the opposing side. The player was able to hit the alien and get an elimination but was hit right after by the alien that was on the opposite side. The player then positions themself to take out the alien on the right-hand side but notices that there are two that are shooting at him. He took too long to shoot at the closer alien and collided with it head-on. This ends his last life. A menu then pops up and shows him his score of 1 point.

Music:

- Background music is 4 notes played on repeat. If time permits, increase the music tempo over time and make the music follow closer to what is played in the real-life version of Defender.

Gameplay - 2-Player Mode:

- Not a lot will change between modes. A separate score tracker will be given to each player. It's similar to 1-player mode, but players take turns. So one player would play through a full game as though it were singleplayer, and the next player would do the same.
- Once both players lose all their lives, their scores are compared and a winner is chosen and displayed.

Time permitting

- Additional enemies with how they move and how they shoot
- Better movement system not tied to locking the x-axis
    - In the original game, you would move left, and the sprite would lock on the right-hand side. If you moved right your sprite would lock on the left-hand side and face right. This made the gameplay feel a lot more smooth.
- The mountain range showing up on the screen
    - Lines are drawn in diagonal patterns along the bottom of the screen to simulate the look of mountains.
- Firework-esque animations for when either enemy or player ships explode.