

CEN 326

Software Project Development

Project 1

Spring 2024

Contents

1.Introduction	3
2.Basic Requirements and Corresponding Algorithm	4
3.Demonstration	5
3.1 Enviroment-1 (mkeykernel)	5
3.2 Enviroment-2 (basekernel)	9
3.3 Enviroment-3 (termios.h)	12
3.4 Enviroment-4 (SDL).....	16

1 Introduction

This project is a game project developed in multiple environments as it will be useful to learn the project development requirements of different development environments. The basis of the game is that we prevent the meteors falling from above from hitting our planet by moving left and right and shooting. In other words, it is a space shooter game. I think that developing games in such different environments and different structures (graphics and console) has a positive impact on my algorithm knowledge.

Let's examine the basic algorithm of the game.

2 Basic Requirements and Corresponding Algorithm

Let me explain it with the example of the main function, which remains the same in every environment for the basic game setup.

```
while (true)
{
    if (game_restart || game_over) // Proper reset with goto in case of
gameover or restart of the game
        goto Restart_Game;

    if (game_pause) // Pause by just freezing the screen without resetting
game objects (Buffer is not a game object, it makes the printing process
faster and in one piece.)
        goto Pause_Game;

    sleep(35); // Sleep 35ms

    // Clear
    draw_space_ship(false);
    draw_bullets(false);
    draw_asteroids(false);

    // Move and do Events
    move_and_fire_space_ship(); // Where spaceship events take place
according to user input
    move_bullets(); // Keep moving until the bullet leaves the screen area
    if (loop_step++ % 90 == 0) // (35 * 90) ms Delayed Spawn
    {
        spawn_asteroids();
    }
    if (loop_step % 15 == 0) // (15 * 90) ms Delayed Move
    {
        move_asteroids();
    }
    collision_event(); // If the bullet hit the asteroid, add points and
disable the asteroid.

    // Print
    draw_space_ship(true);
    draw_bullets(true);
    draw_asteroids(true);
    print_score();
    print_life();
}
```

3 Demonstration

Screenshots from the game taken from different environments.

3.1 mkeykernel

Game Start



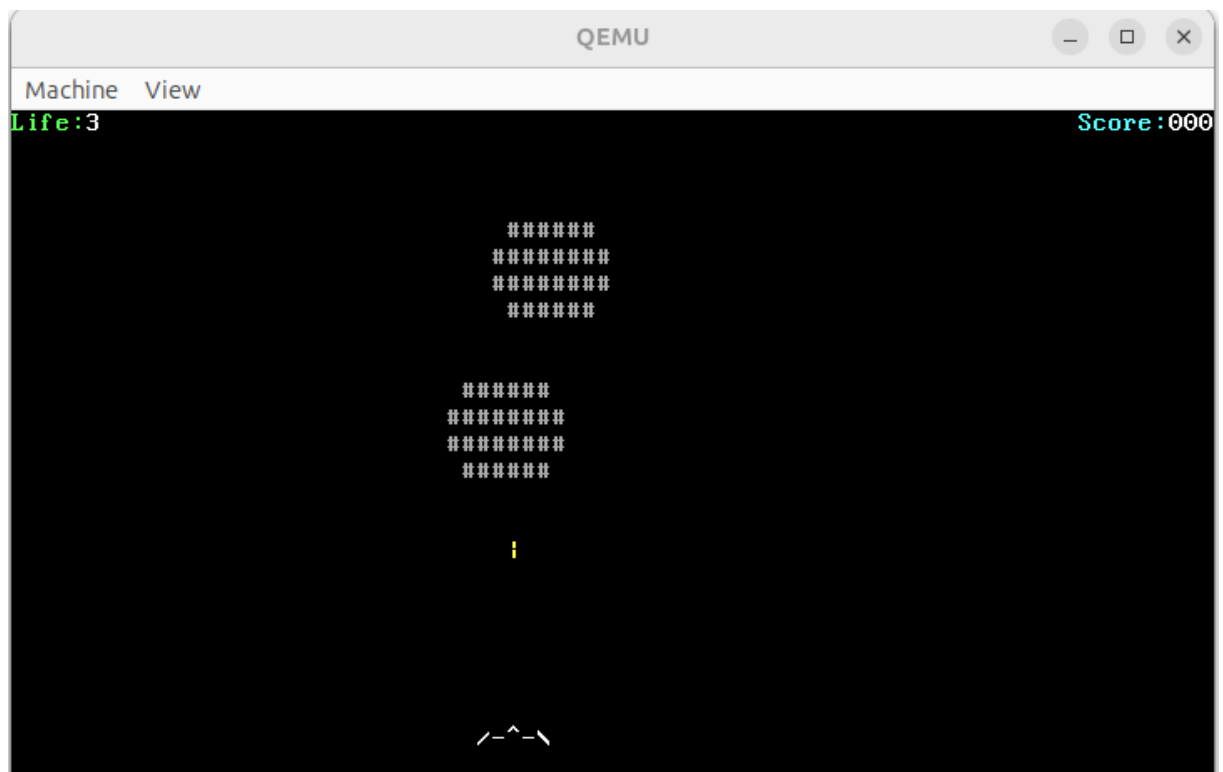
In Game



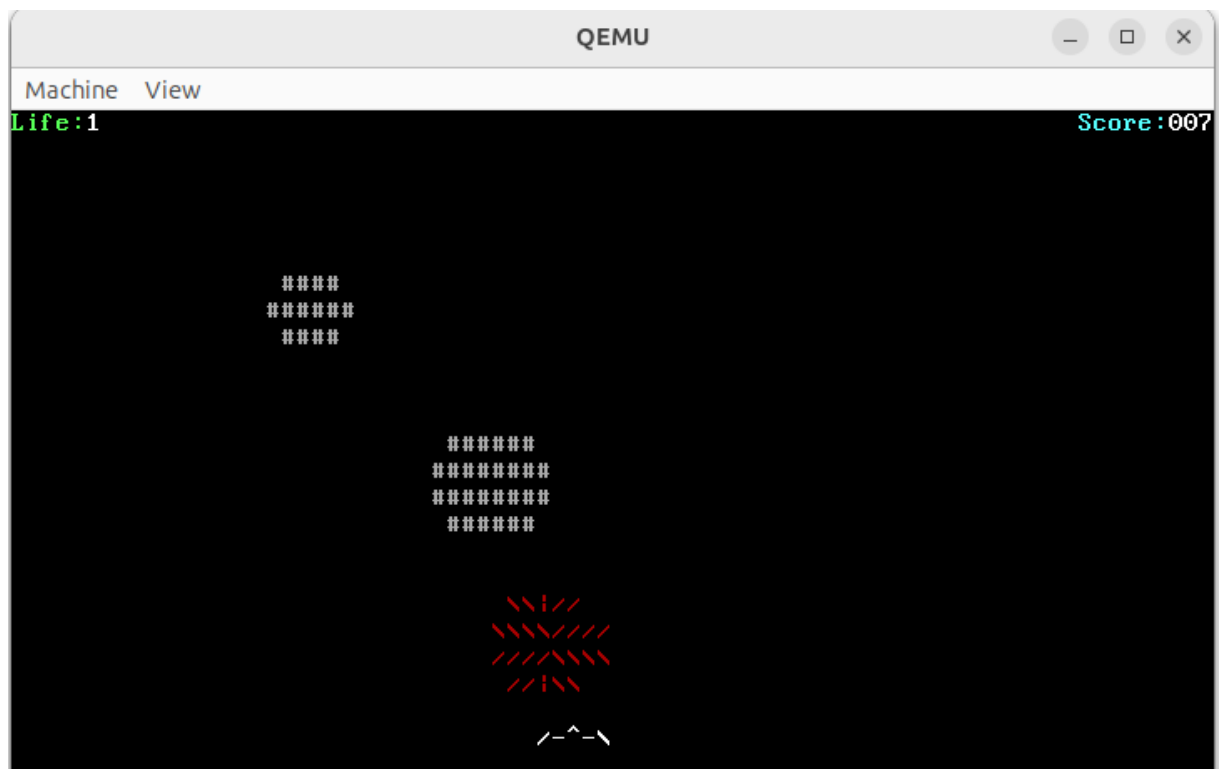
On Pause



On Shot



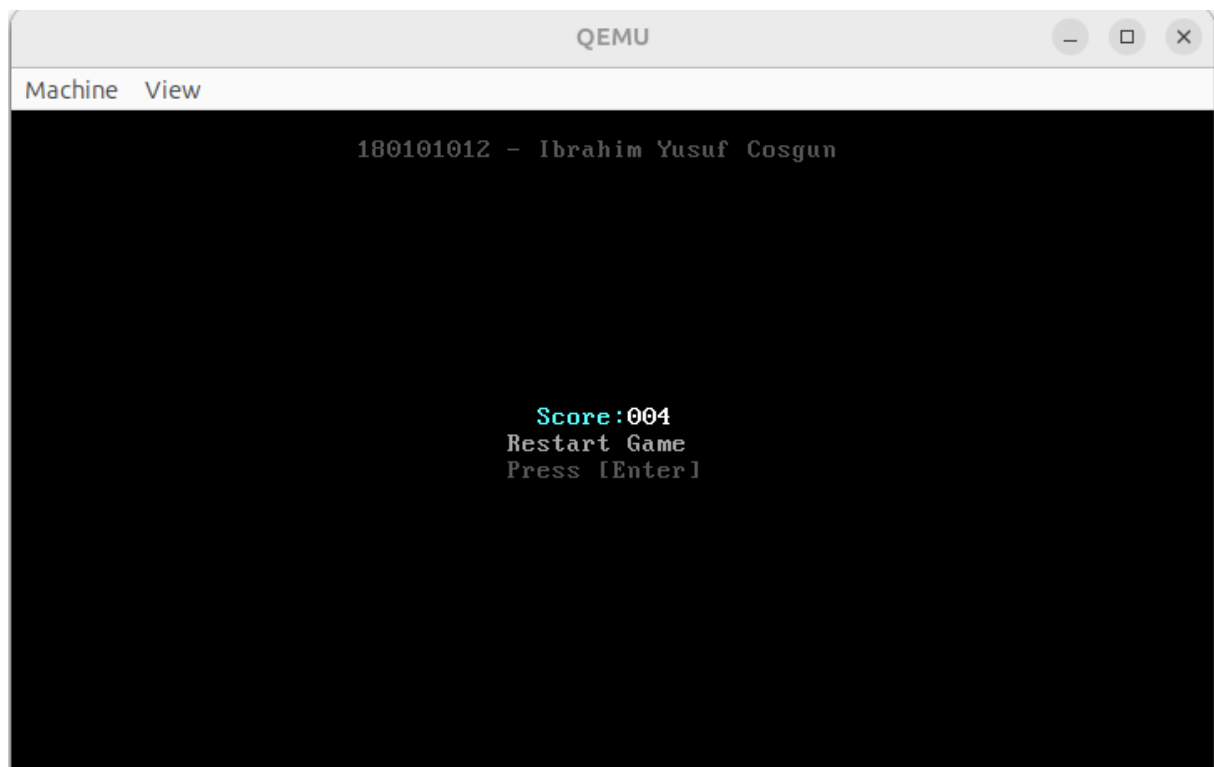
On Hit



Game Over

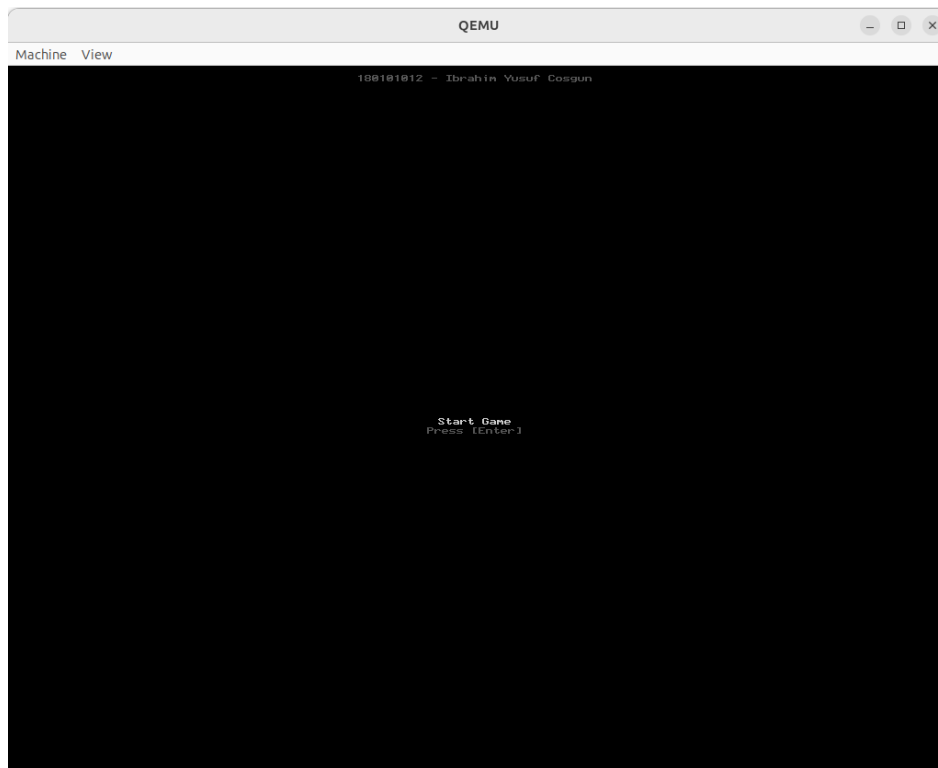


Game Restart

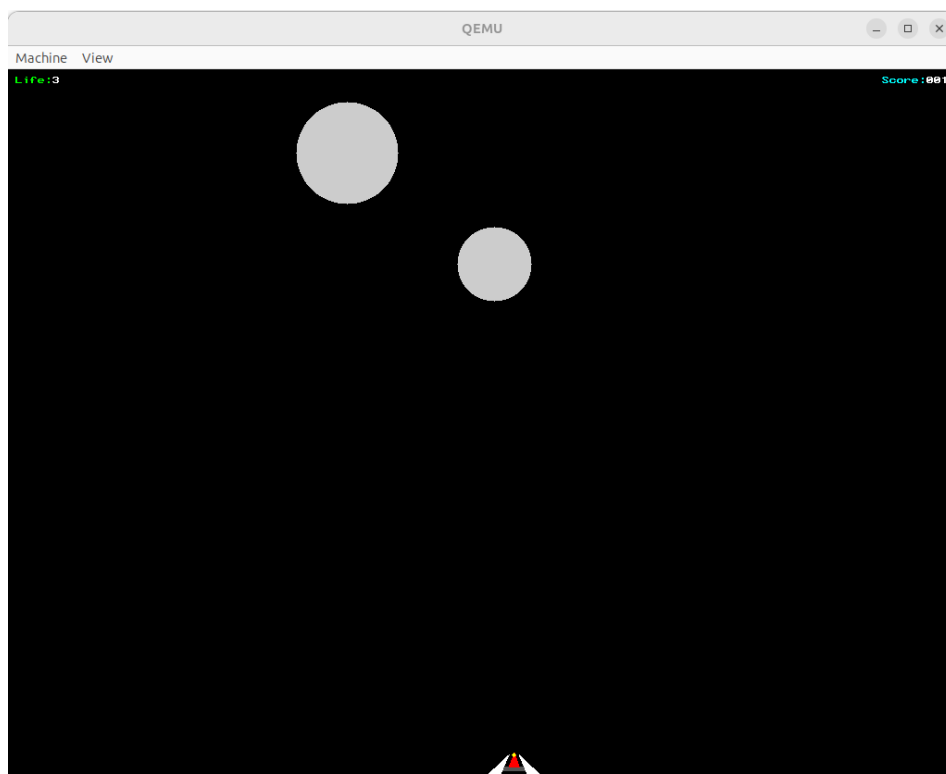


3.2 basekernel

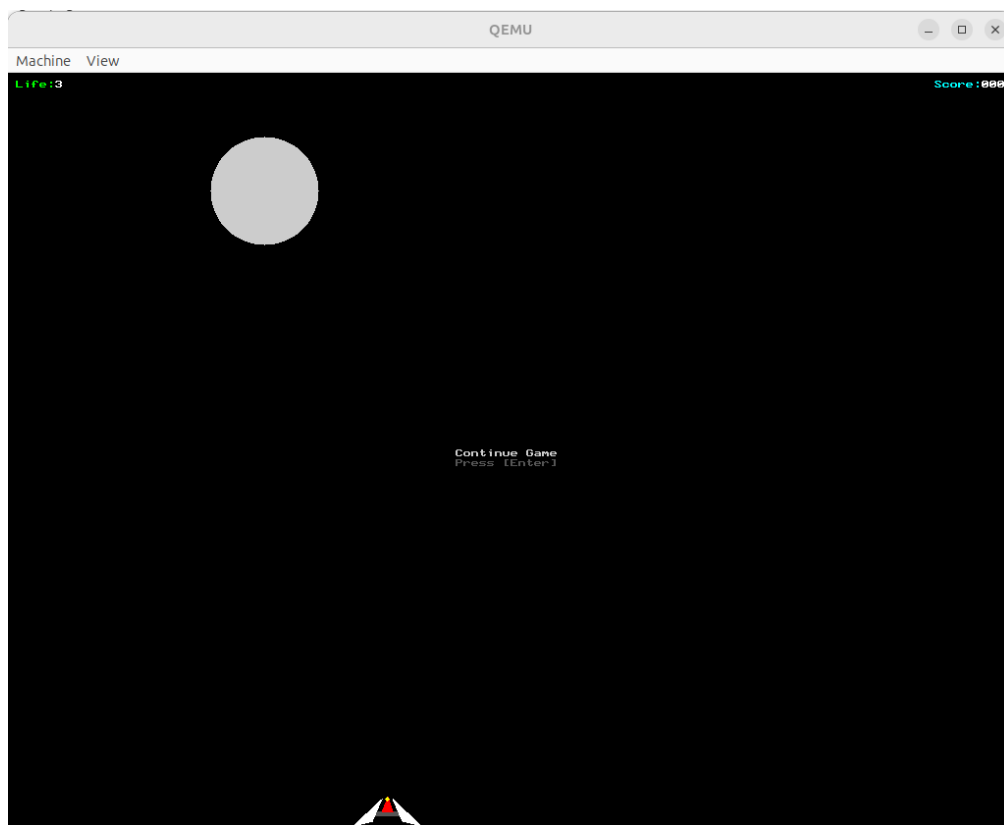
Start Game



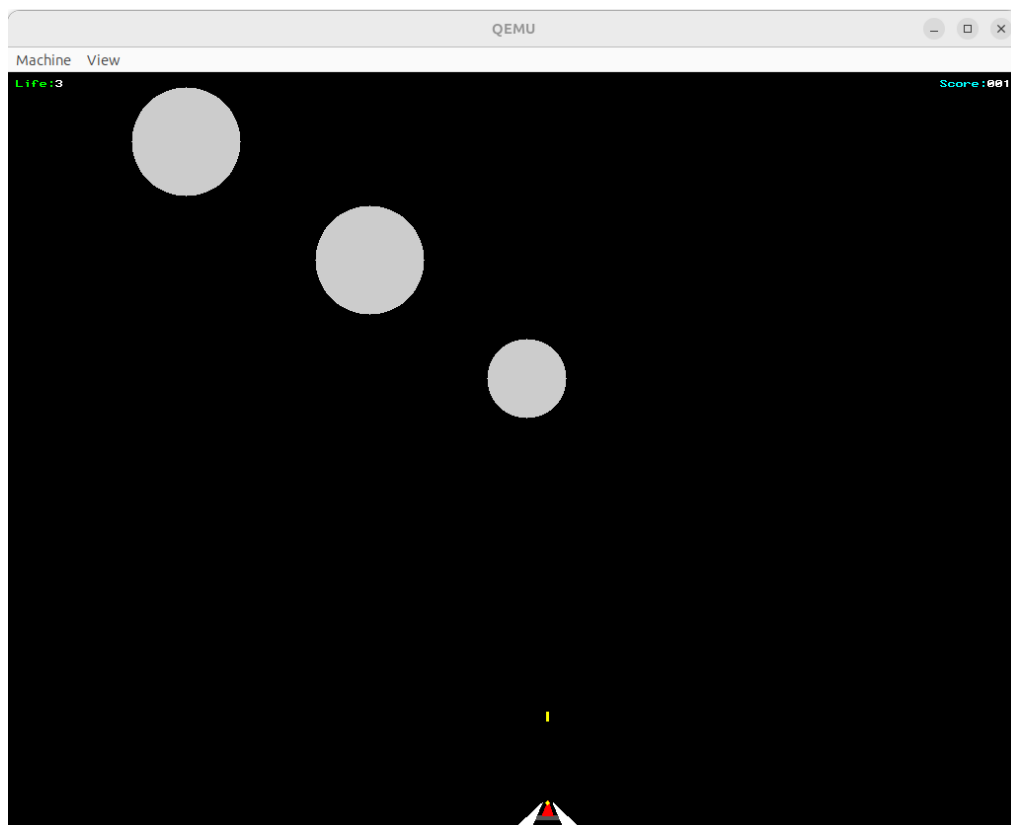
In Game



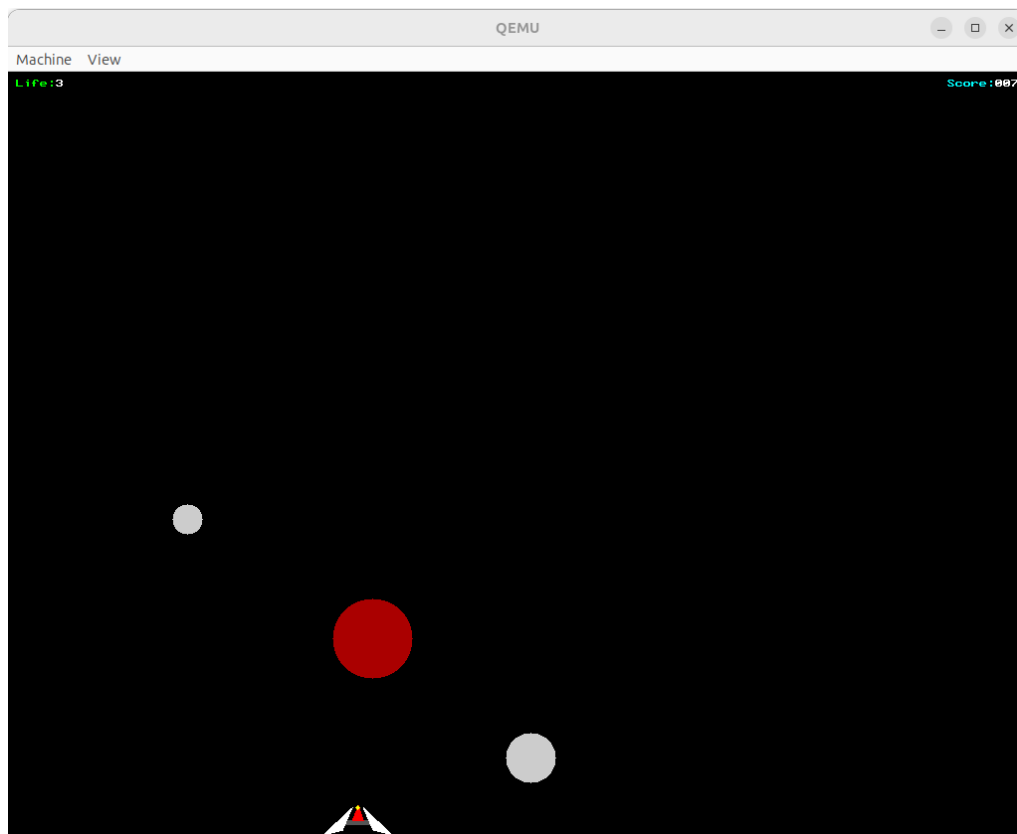
On Pause



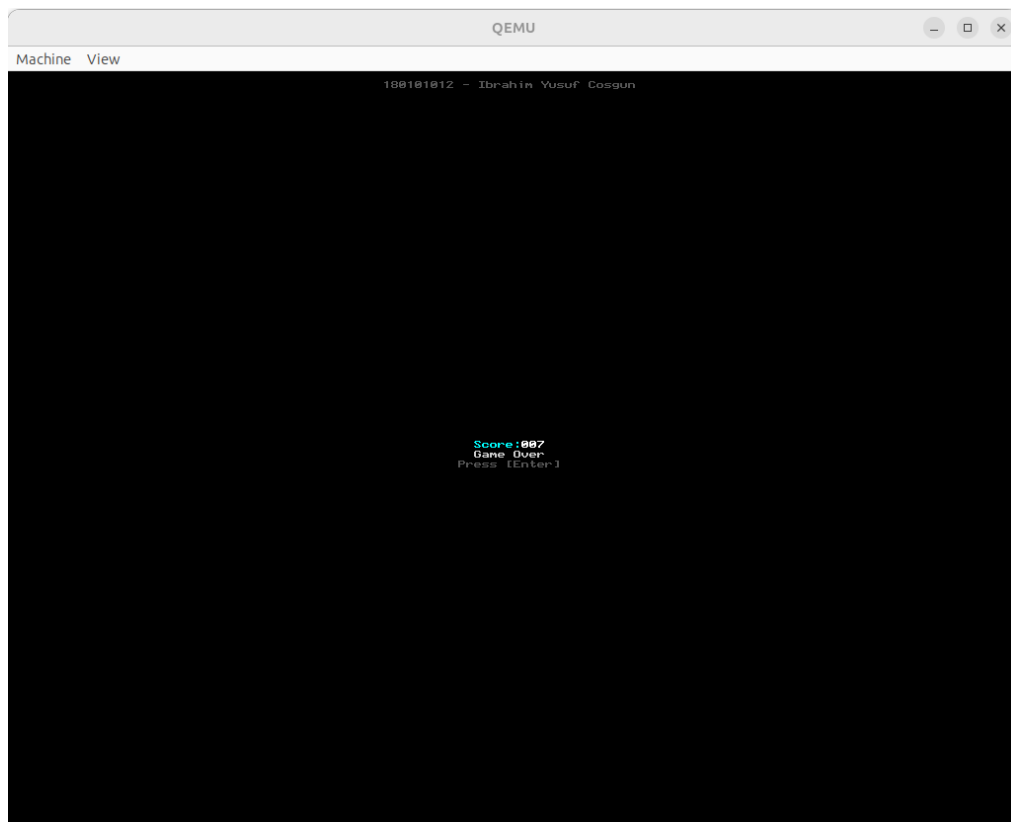
On Shot



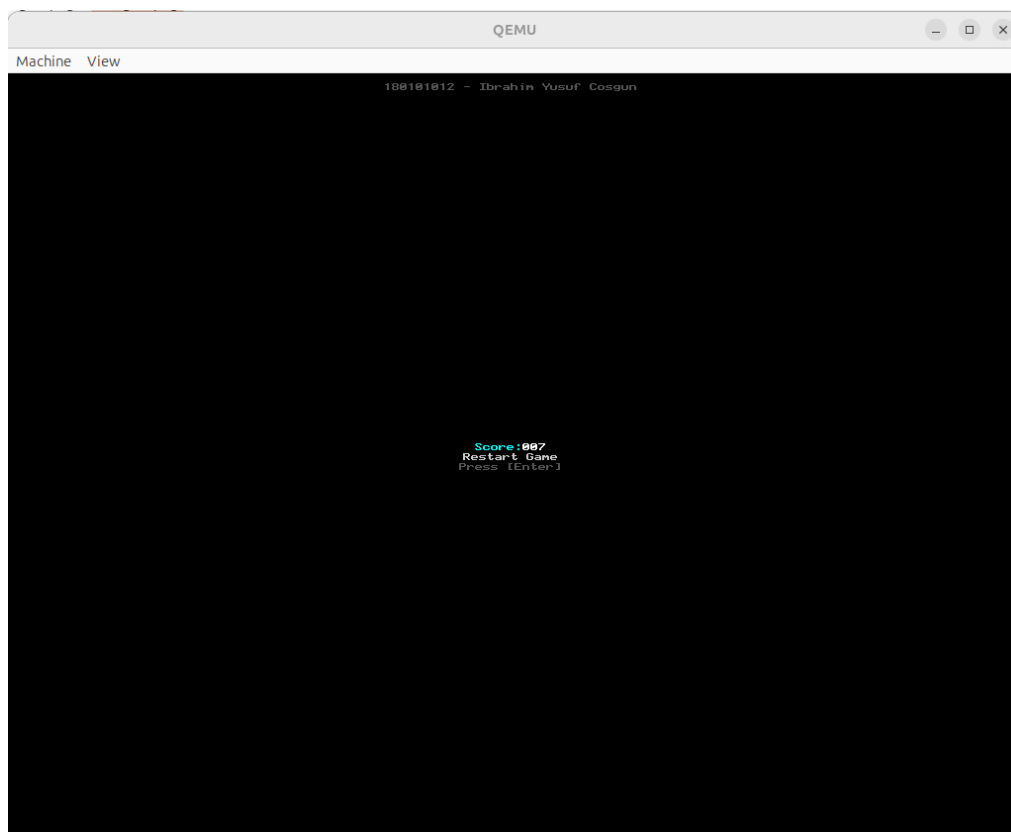
On Hit



Game Over



Game Restart

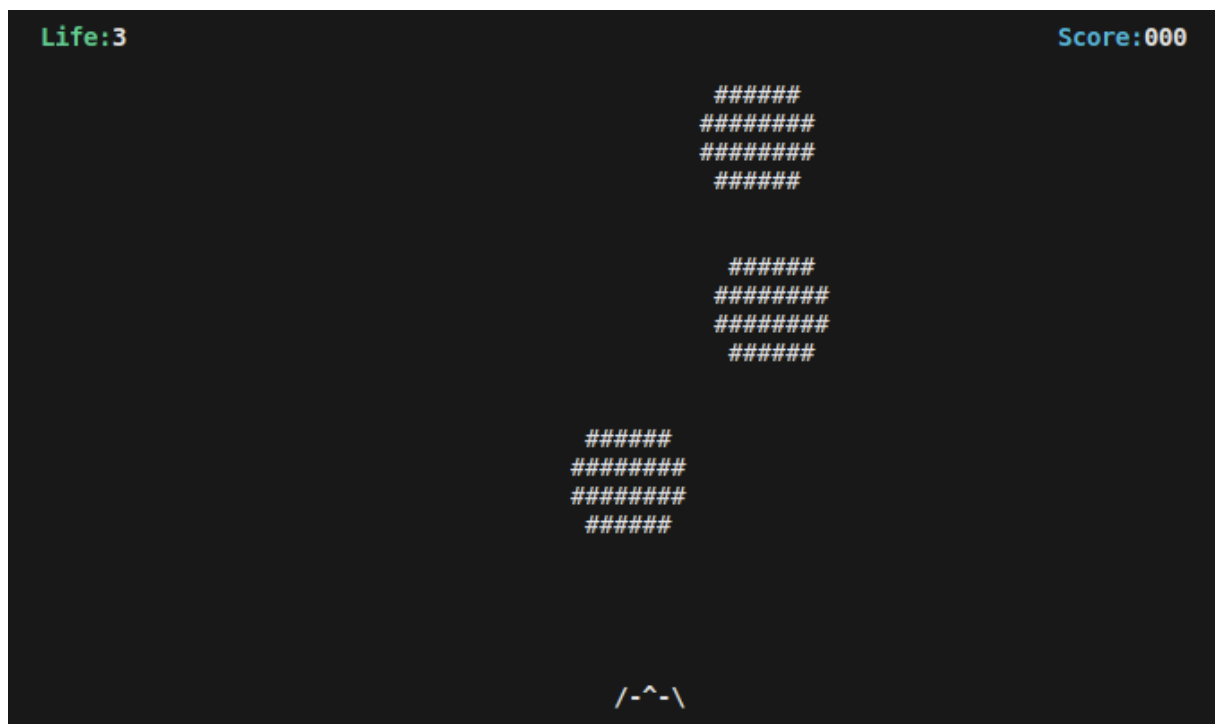


3.3 termios.h

Game Start



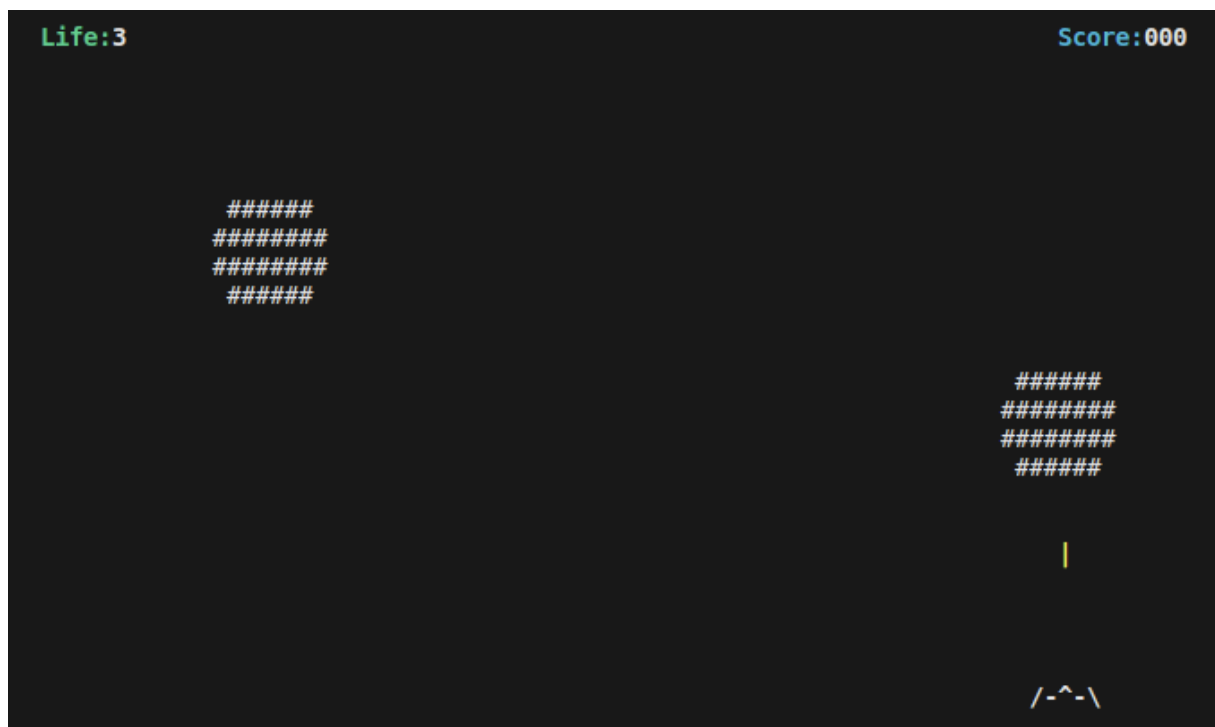
In Game



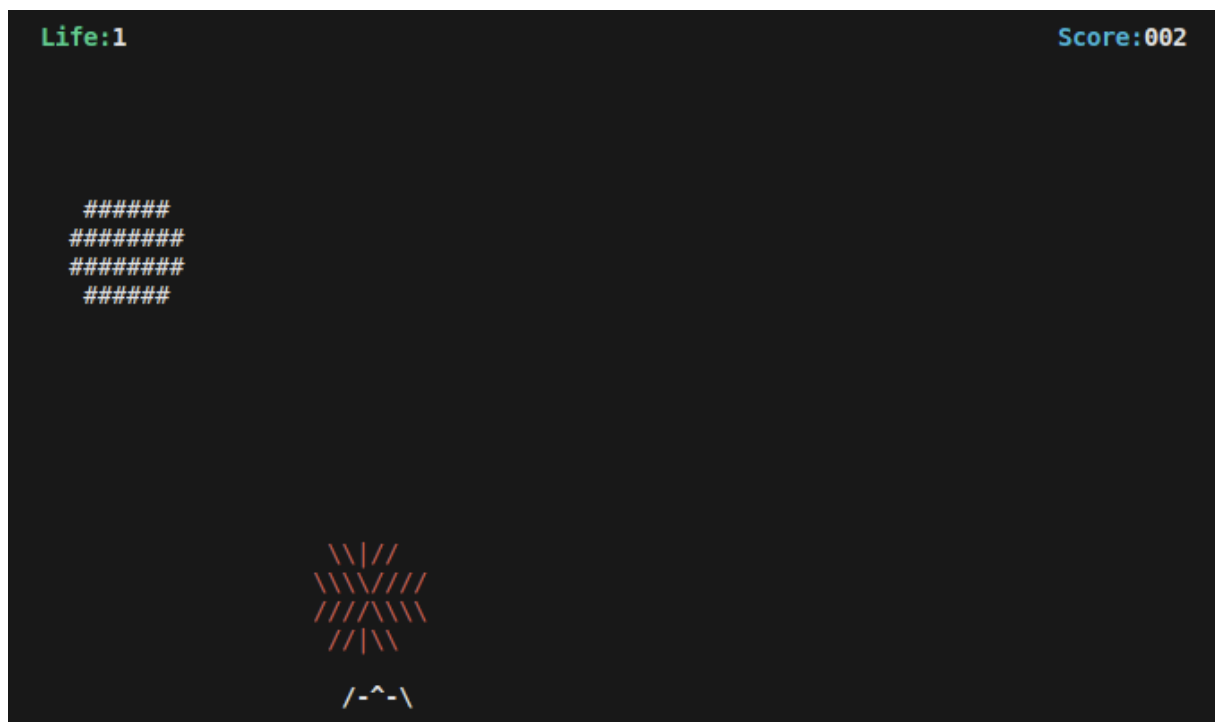
On Pause



On Shot



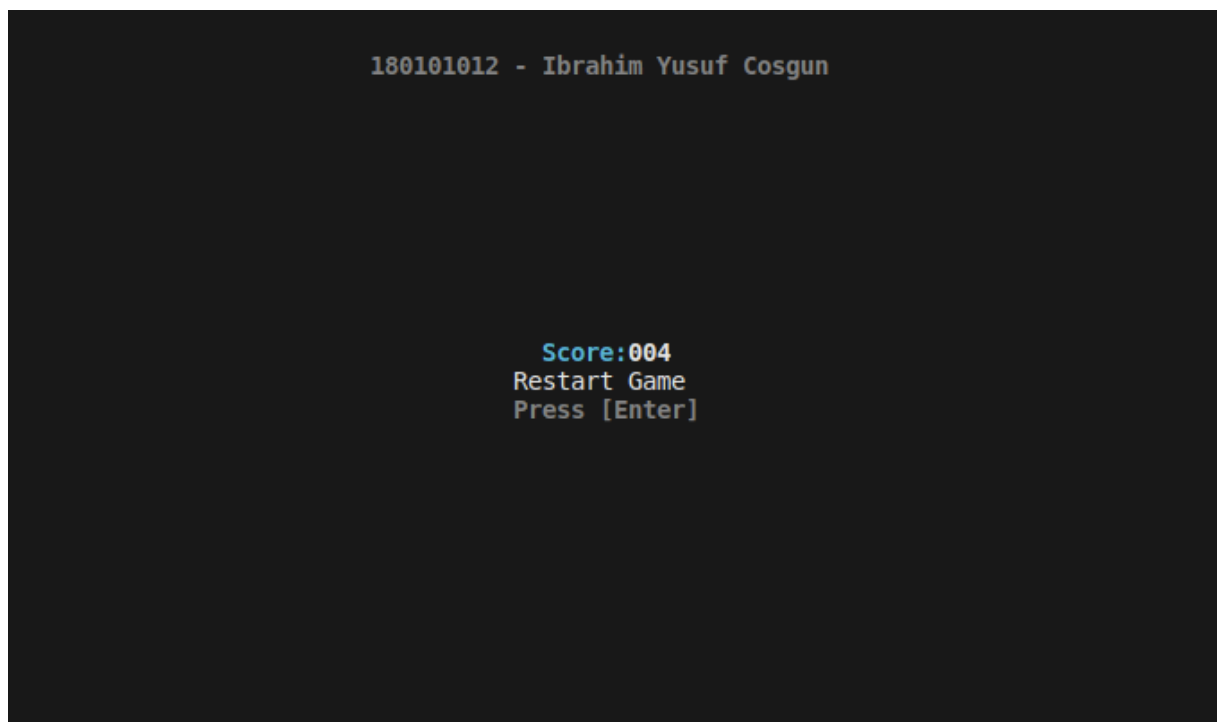
On Hit



Game Over

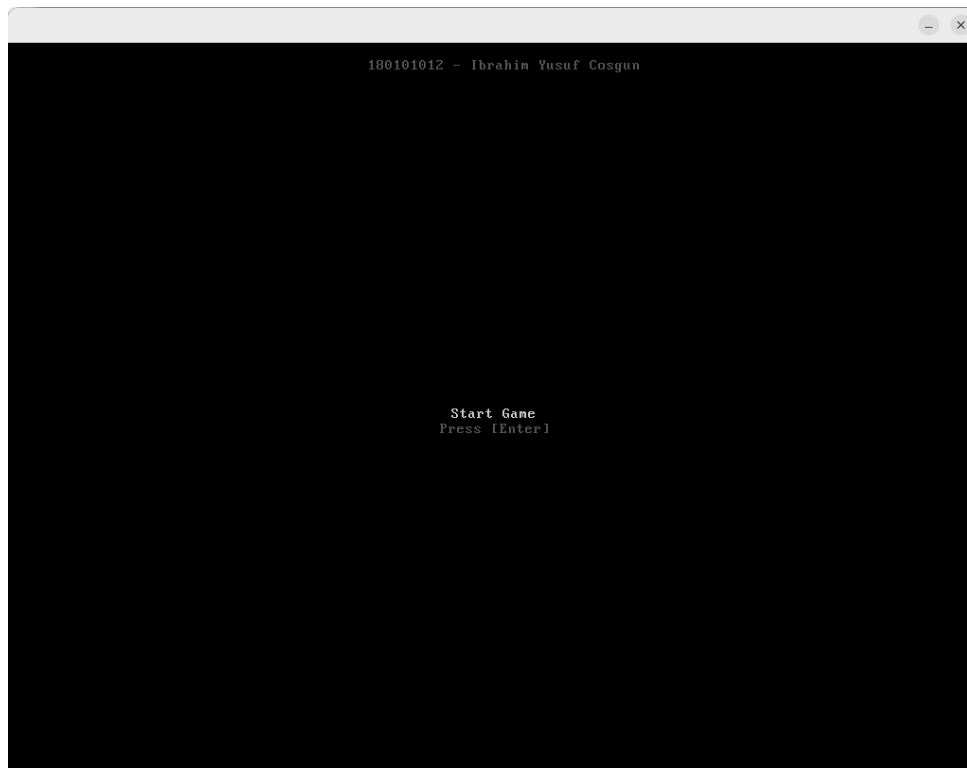


Game Restart

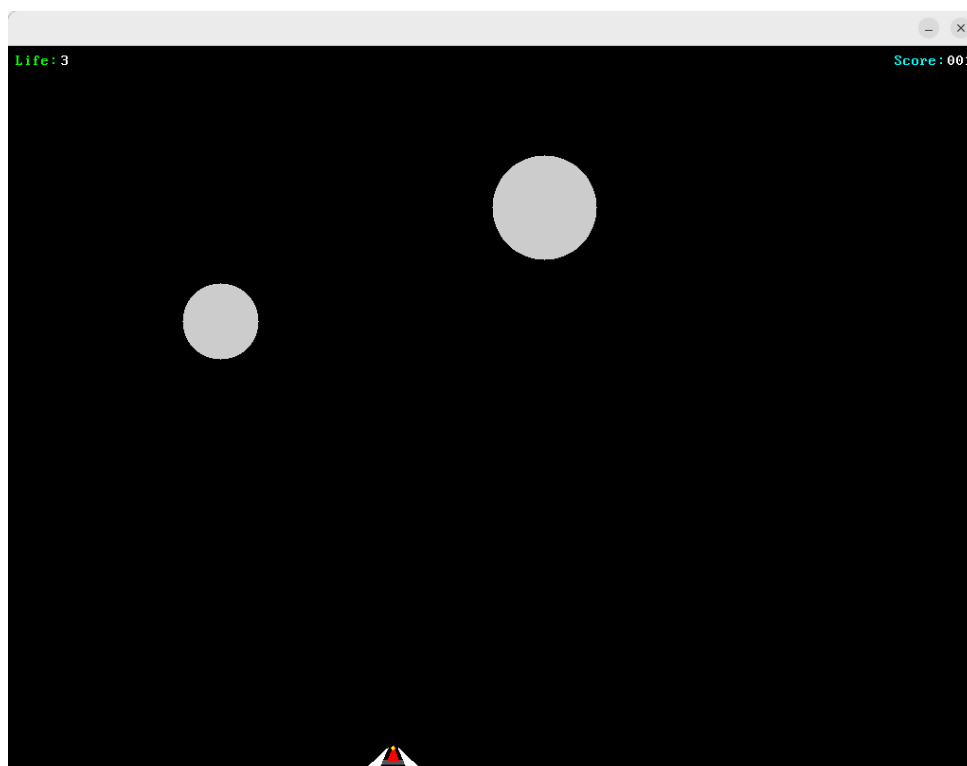


3.4 SDL (Simple DirectMedia Library)

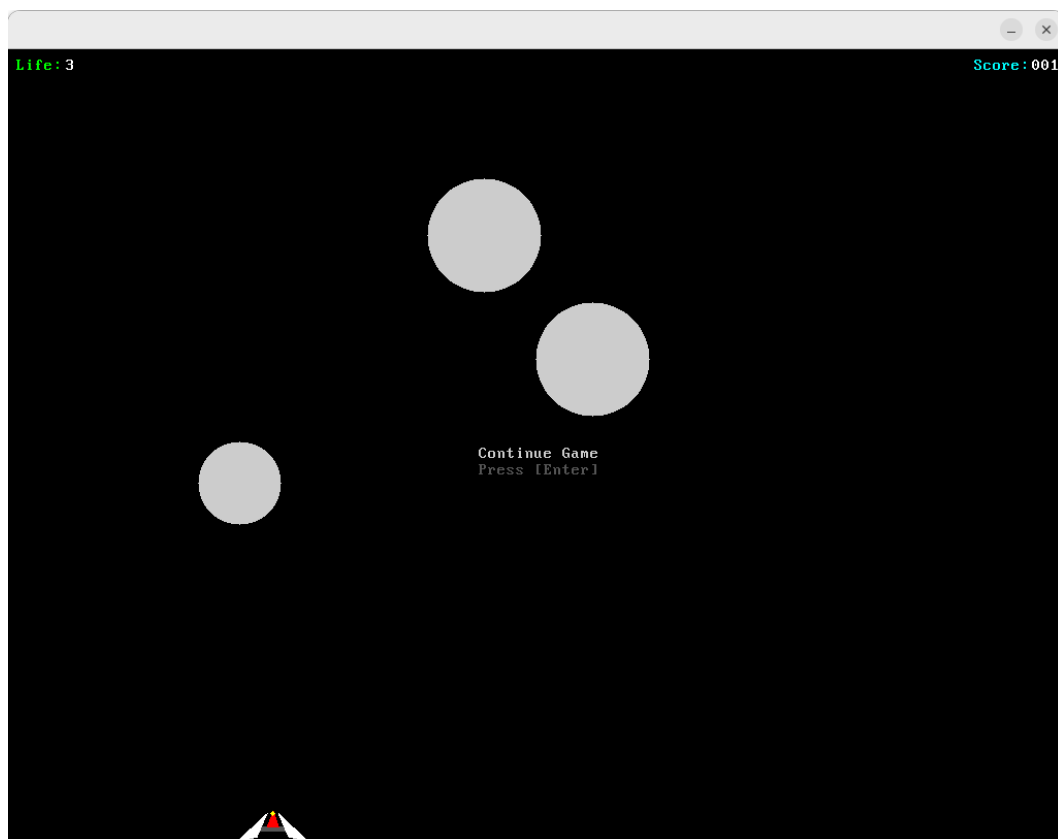
Game Start



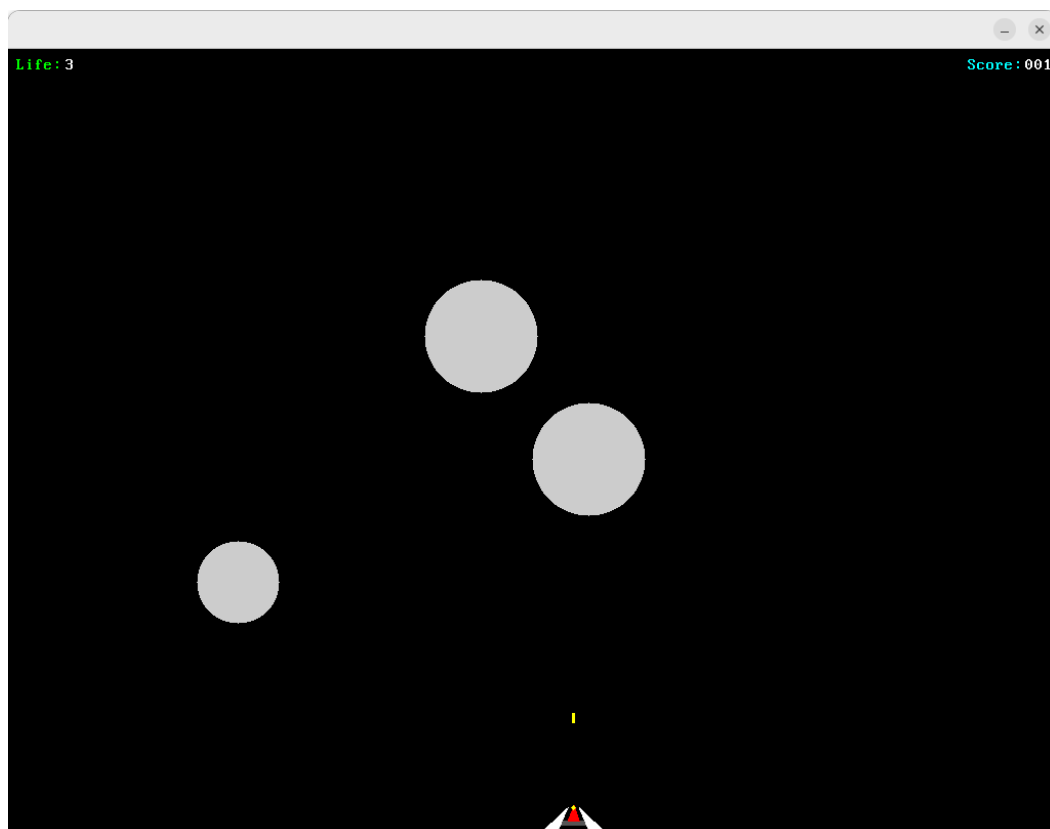
In Game



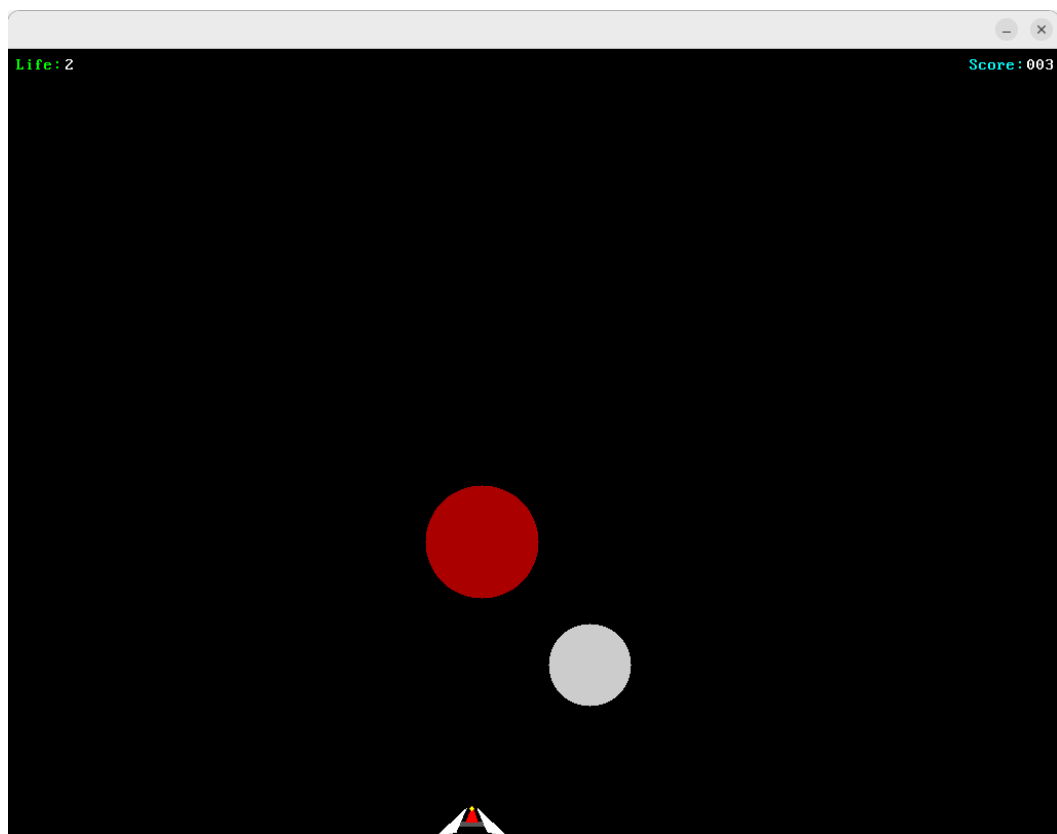
On Pause



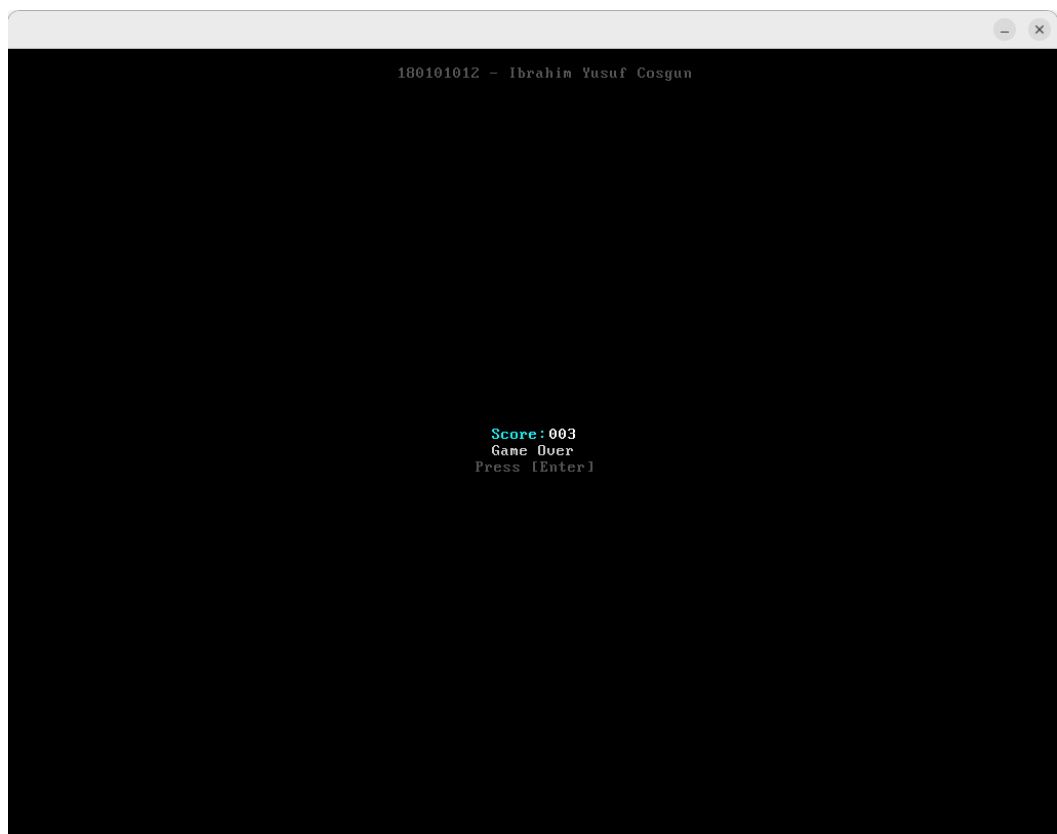
On Shot



On Hit



Game Over



Game Restart

