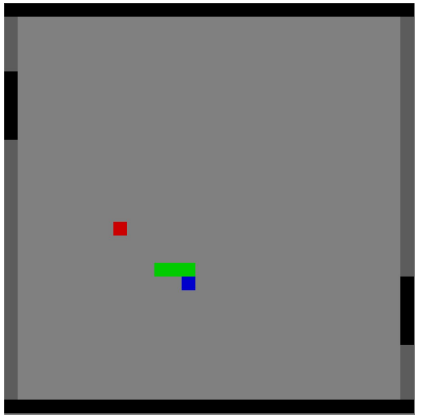
Game 21

Jaeseok Choi

Project Overview

- Objective
- Rules
- Implementation
- Constraints
- Features
- Technology used
- Application

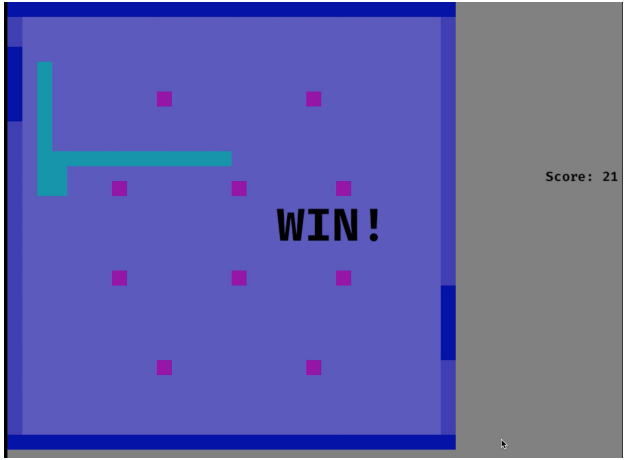
1. Objective

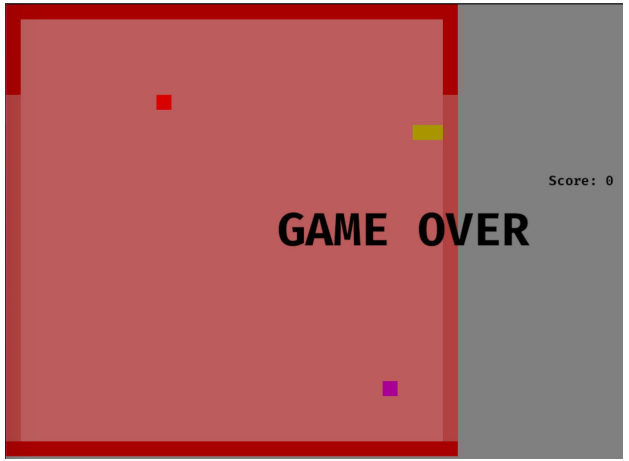


- Similar to ‘Snake Game’, but more complexity added
  - Food component moves and bounces off the boundaries
  - Enemy component -> starts with one enemy and increases as the snake eats the food twice
  - Two side pedals can be moved with ‘Q’, ‘A’, ‘R’, and ‘S’ keys -> must prevent food from going out of boundaries
  - Once the food is eaten, it will be generated at a random point
  - Snake is controlled with arrow keys

2. Rules

- Wins game when
  - Must reach 21 points to win the game
- Loses game when
  - Food goes out of side boundaries
  - Snake contacts with Enemy
  - Snake touches its body
  - Snake goes out of boundary





3. Implementation

```rust
pub struct Snake {
    direction: Direction,
    body: LinkedList<Block>,
    tail: Option<Block>
}
```

```rust
pub struct Enemy {
    gang: Vec<Block>
}
```

```rust
pub struct LeftPedal {
    body: LinkedList<Block>
}

pub struct RightPedal {
    body: LinkedList<Block>
}
```

```rust
pub struct Game {
    snake: Snake,

    l_pedal: LeftPedal,
    r_pedal: RightPedal,

    enemy: Enemy,

    food_exists: bool,
    food_x: i32,
    food_y: i32,
    food_speed_x: i32,
    food_speed_y: i32,

    width: i32,
    height: i32,

    game_over: bool,
    game_win: bool,
    waiting_time: f64,
    score: i32
}
```

4. Constraints

- Minimal design of entities
  - Rectangles
- Desktop game only
- Keyboard inputs tend to slow down the program a little bit

5. Features

- Fun and simple
- Fast execution
- Increase multi-tasking skills || body coordination

6. Technology Used

- Piston - a modular open source game engine
- Graphics, Glutin_Window, and many other crates

Piston

7. Application

- Basic game implemented on OSX, Linux, etc.

Questions?