

3. **Skipping Drawing for Hit Bowls:** While the code does not explicitly avoid drawing a bowl after it's hit (because it always calls `bowl.draw()`), you could modify this behavior by adding a check to only call `draw()` if the bowl hasn't been hit yet:

javascript

 Kopiera kod

```
if (!bowl.hit) {  
    bowl.draw();  
}
```

This way, once a bowl is hit (i.e., its `hit` property becomes `true`), it won't be drawn again in subsequent frames.

4. **Checking Win Condition:** The variable `allBowlsHit` is set to `false` if any bowl has not been hit. Once all bowls are hit (`allBowlsHit` becomes `true`), the game transitions to the "win" state:

javascript

 Kopiera kod

```
if (!bowl.hit) {  
    allBowlsHit = false;  
}
```



This ensures that the game logic knows when all bowls have been hit, triggering the win

Skicka ett meddelande till ChatGPT





The key logic that prevents the code from drawing another bowl when a bowl is hit lies in how the `hit` property of each bowl is managed.

1. **Bowl Object with a Hit Property:** Each `bowl` has a `hit` property, which tracks whether it has been hit by the ball. When the ball collides with a bowl (i.e., the condition inside the `if` block is met), the bowl's `hit` property is set to `true`.

javascript

Kopiera kod

```
if (!bowl.hit && dist(ball.x, ball.y, bowl.x + bowl.width / 2, bowl.y + bowl.height / 2) < radius) {
    bowl.hit = true; // Mark bowl as hit
    ball.speedY = -ball.speedY; // Bounce the ball
}
```

2. **Collision Detection:** The collision detection checks whether the ball has intersected with the bowl. If the bowl has already been hit (`bowl.hit` is `true`), the collision check is skipped for that bowl, meaning the ball will not interact with it again. The `dist()` function calculates the distance between the ball and the bowl's center, and if that distance is smaller than the combined radius of the ball and the bowl, the bowl is considered "hit," and the ball's speed is reversed.
3. **Skipping Drawing for Hit Bowls:** While the code does not explicitly avoid drawing a bowl after it's hit (because it always calls `bowl.draw()`), you could modify this behavior by adding a

Skicka ett meddelande till ChatGPT



How do this code know not to draw another bowl when a bowl is hit?

```
for (let row = 0; row < ROWS; row++) {
  for (let col = 0; col < COLUMNS; col++) {
    let bowl = bowls[row][col];
    bowl.draw();

    // Check for collision
    if (!bowl.hit && dist(ball.x, ball.y, bowl.x + bowl.width / 2, bowl.y +
bowl.height / 2) < ball.r + bowl.width / 2) {
      bowl.hit = true; // Mark bowl as hit
      ball.speedY = -ball.speedY; // Bounce the ball
    }

    // Check if all bowls are hit
    if (!bowl.hit) {
      allBowlsHit = false;
    }
  }
}

// Check win condition
if (allBowlsHit) {
  state = "result";
}
```

Skicka ett meddelande till ChatGPT

