

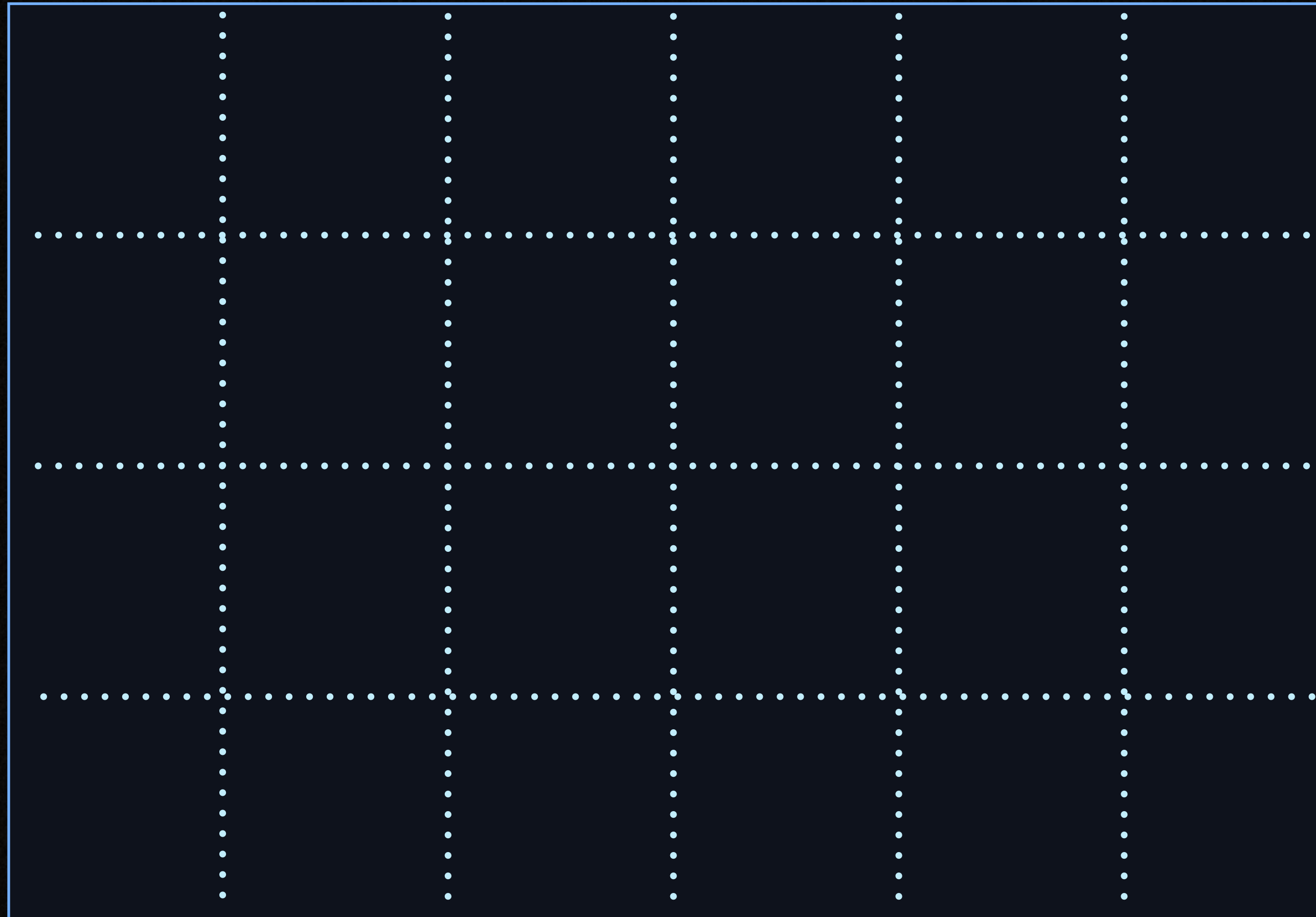


BOUNCING BALL



board is a `[[[]bool slice`

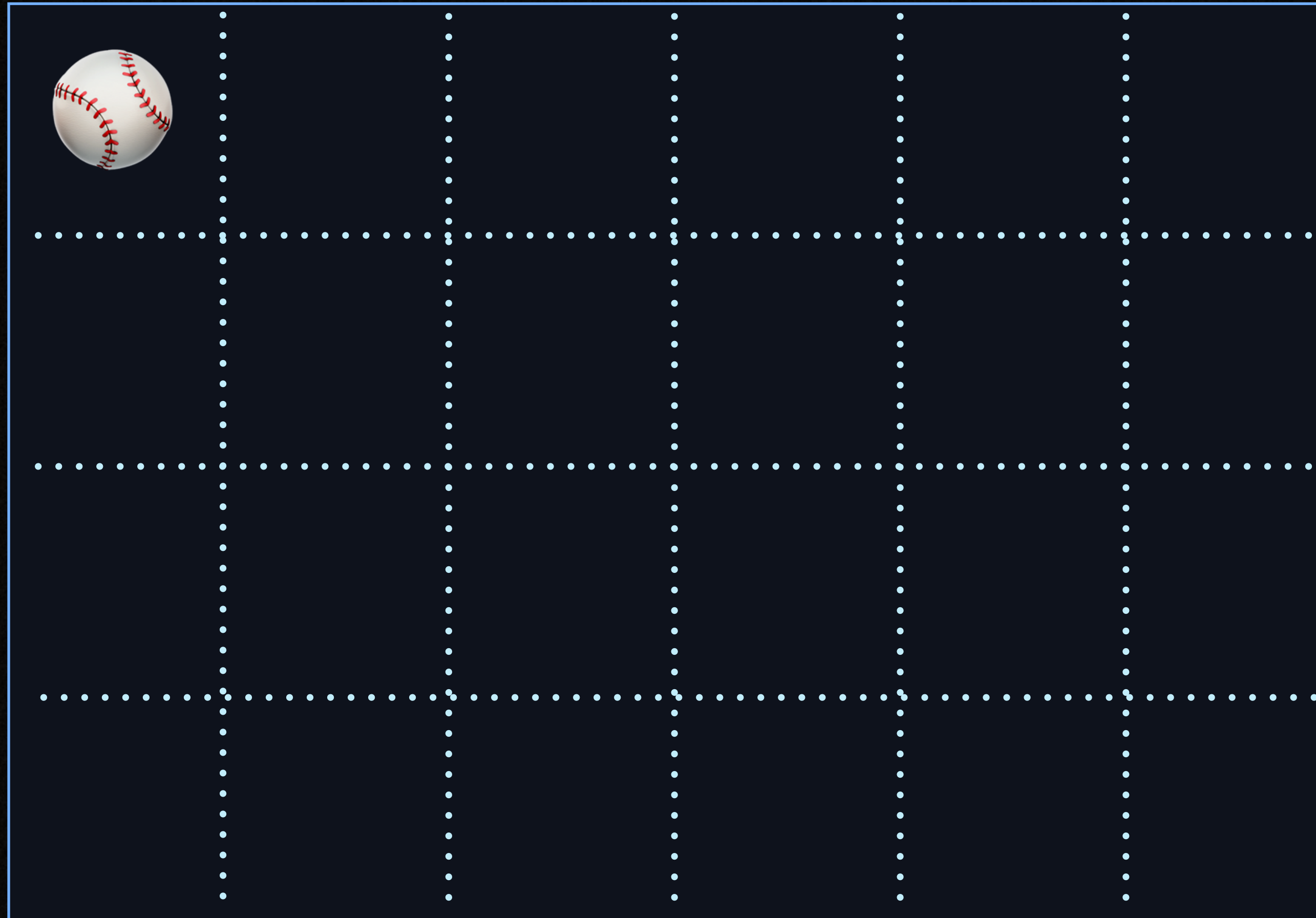
When an `element` is `false`; print an empty cell



X Y

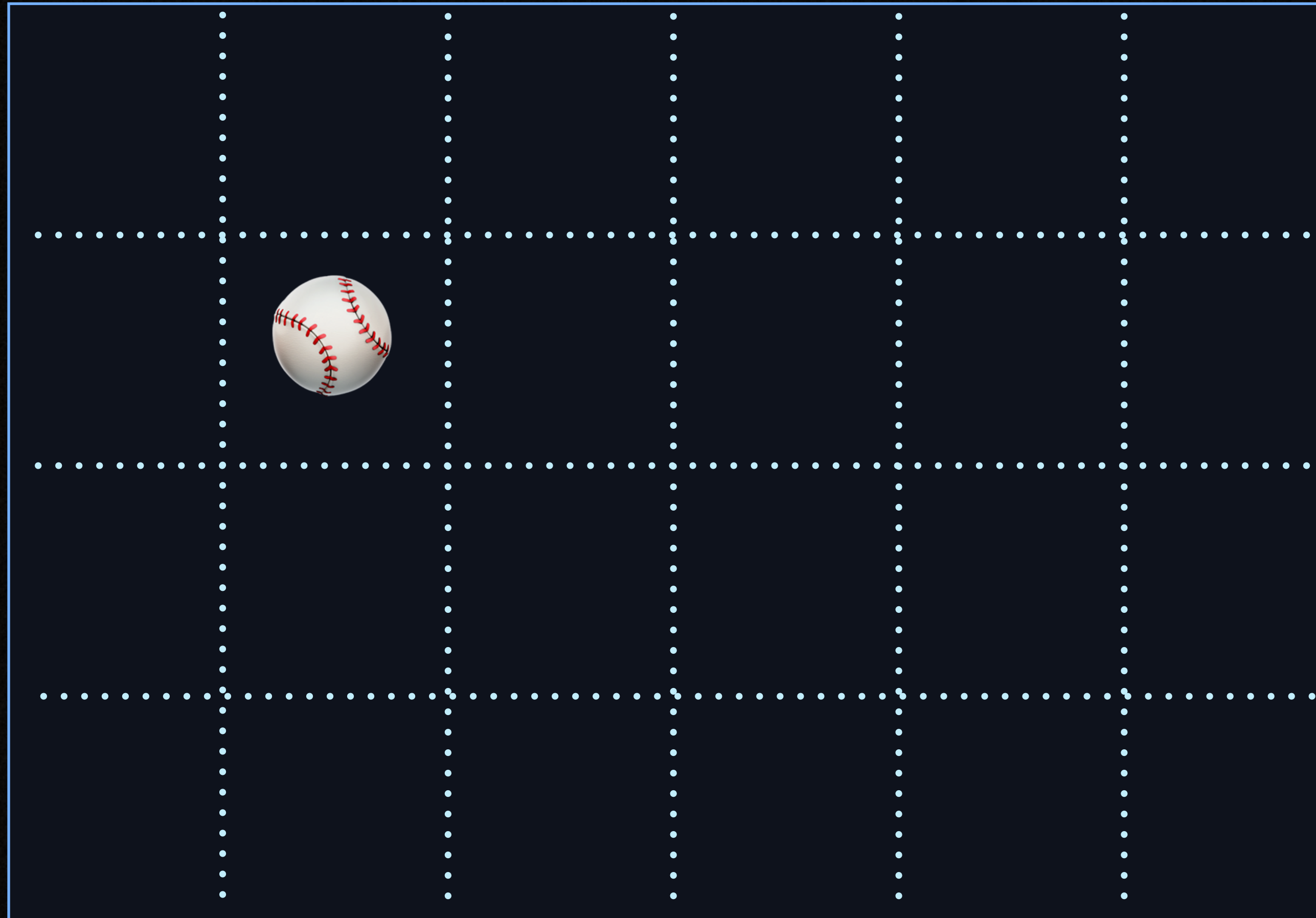
```
board[0][0] = true
```

When an element is true; print the ball on that position



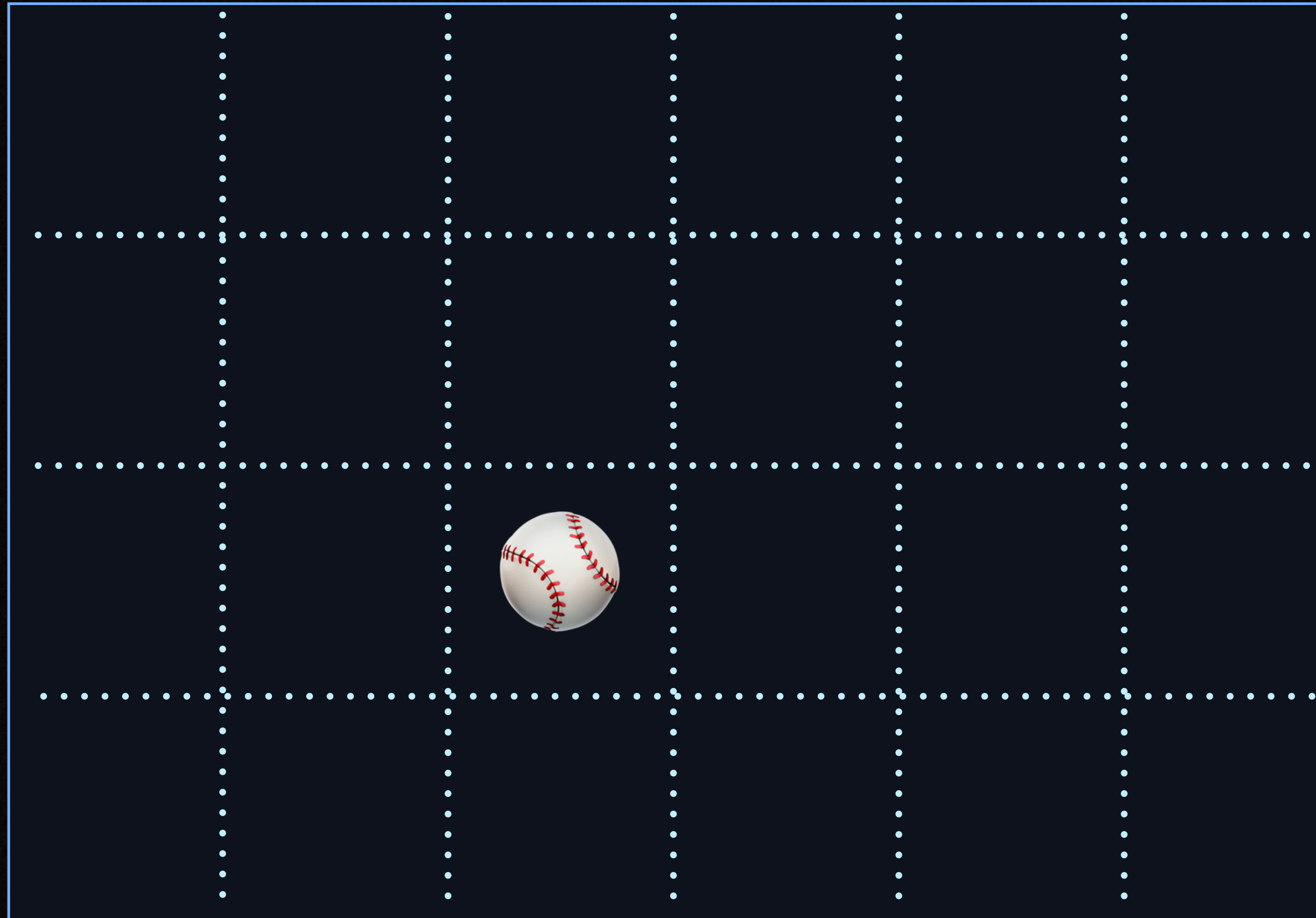

```
board[1][1] = true
```

The program starts animating the ball



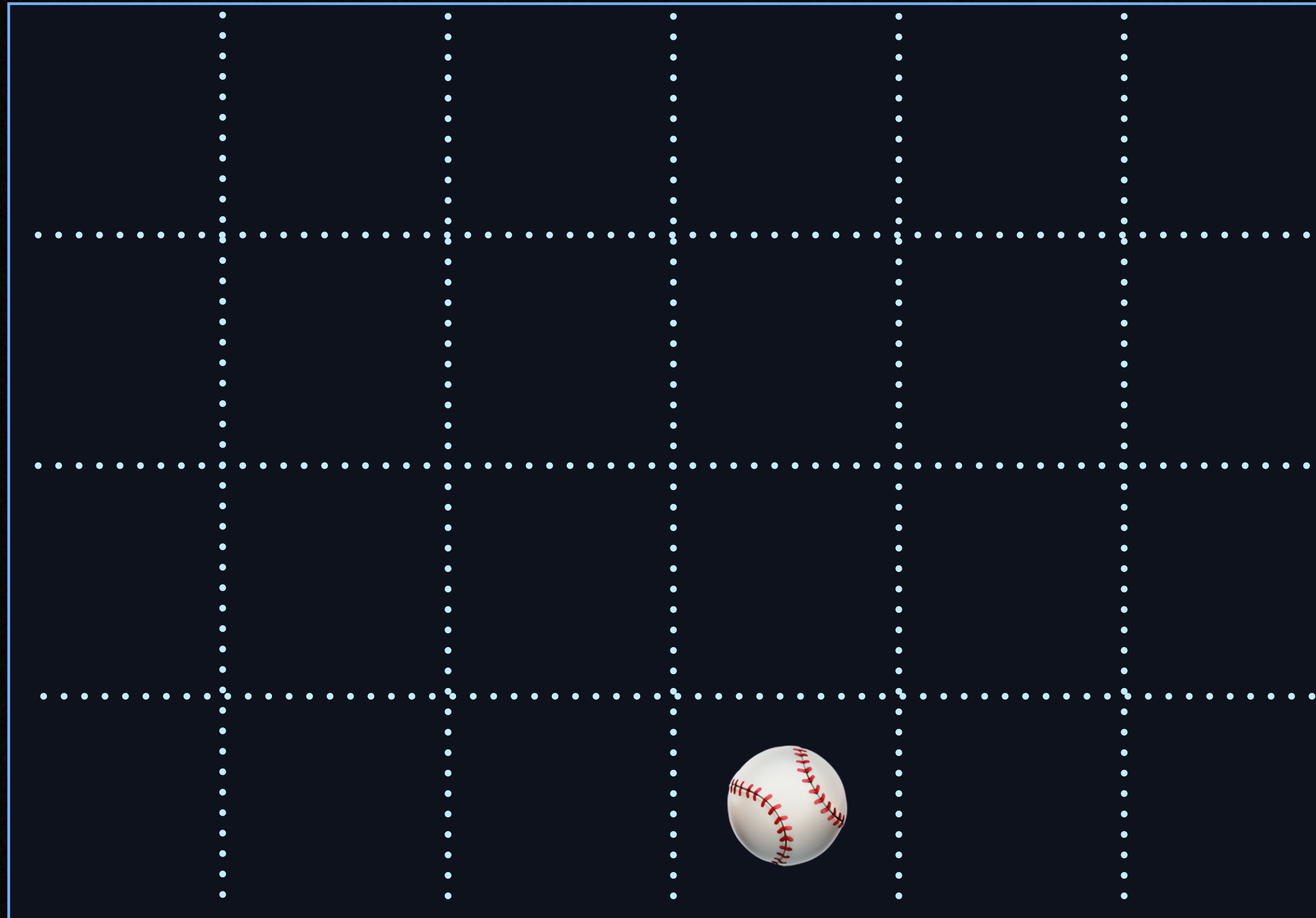

```
board[2][2] = true
```

The other elements become false; there is only one element that becomes true



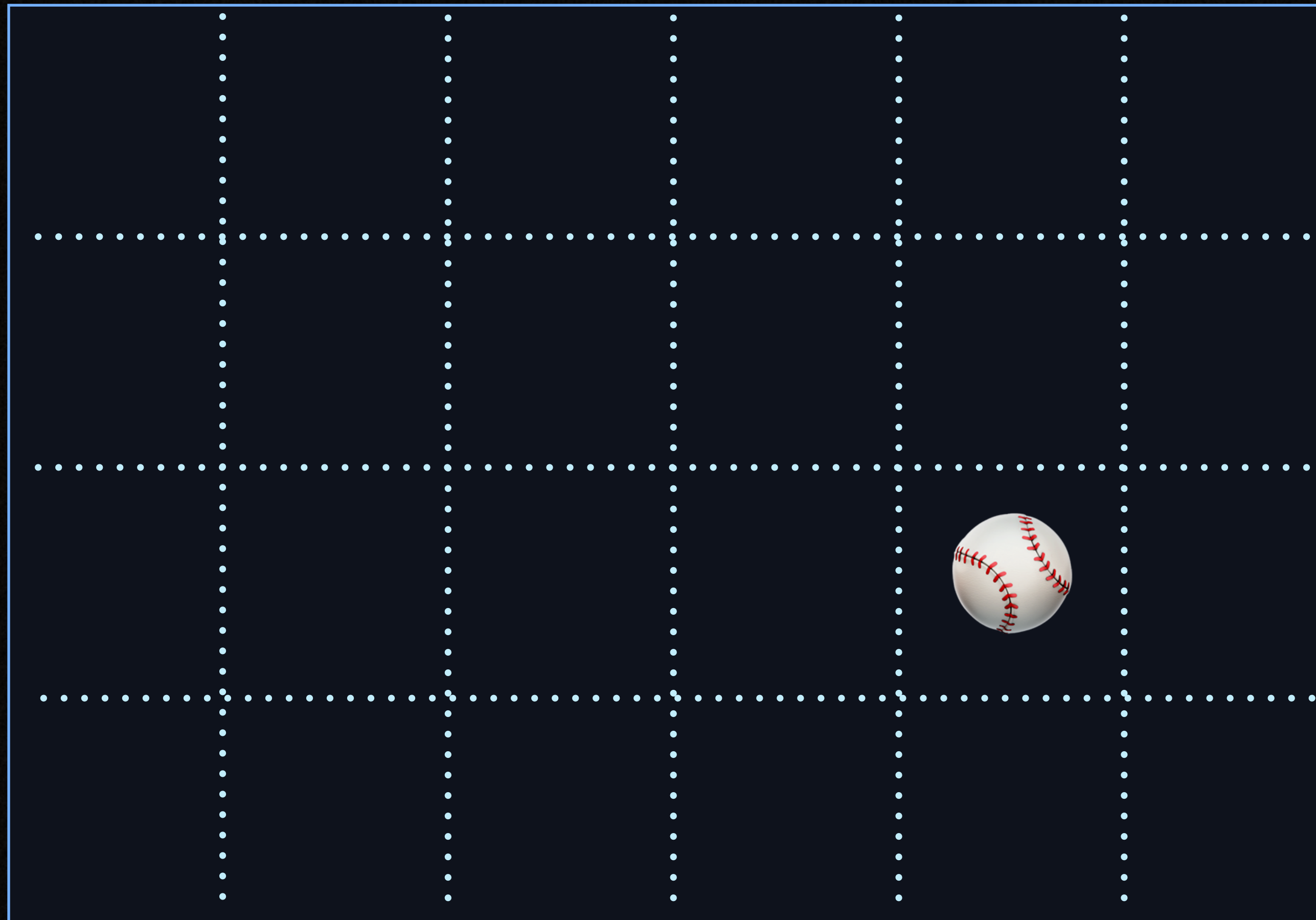

```
board[3][3] = true
```

When the ball hits an edge; it should bounce



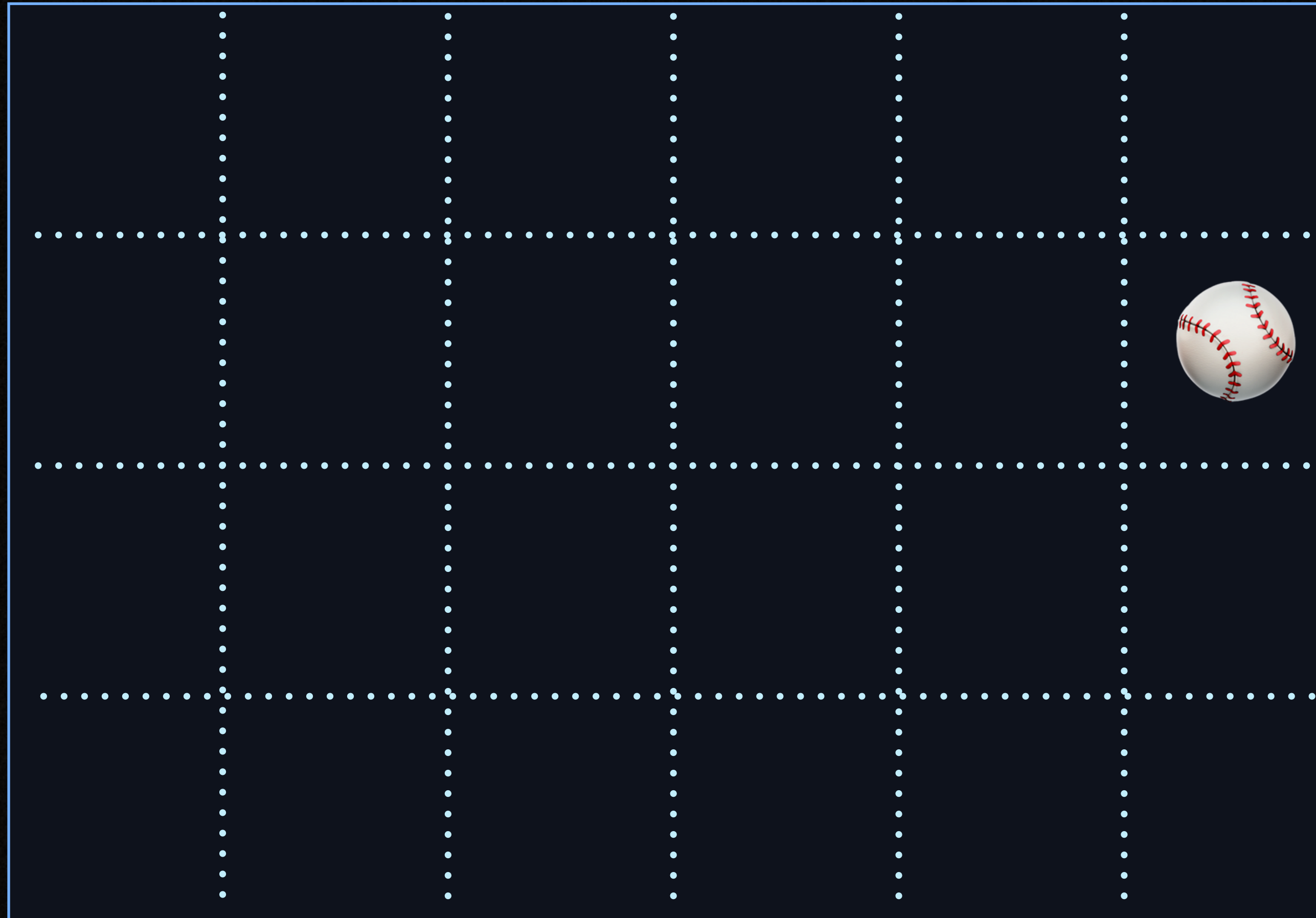

```
board[4][2] = true
```

The program **reverses** only the **Y-direction** of the **ball**



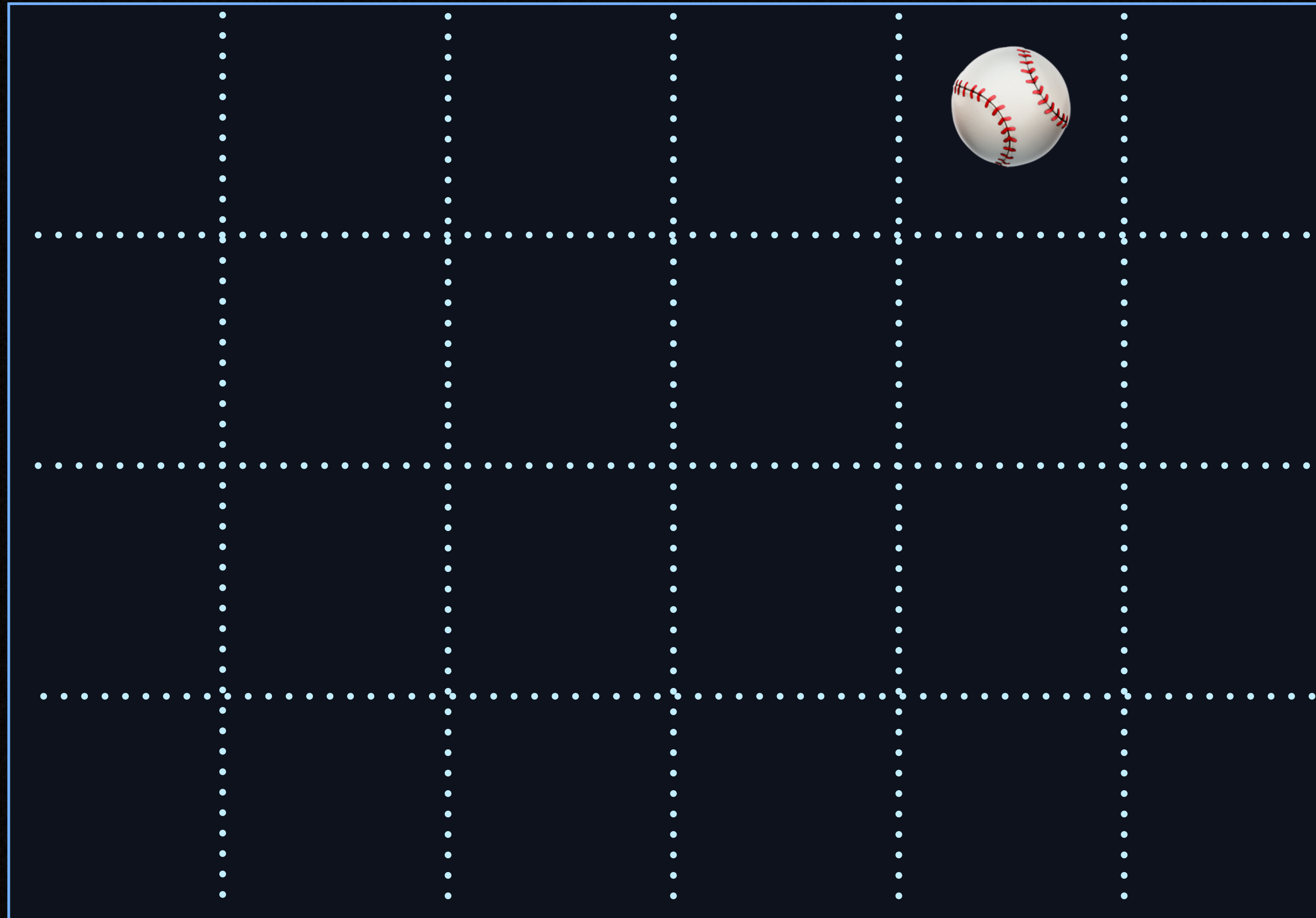

```
board[5][1] = true
```

When the ball hits an edge; it should bounce



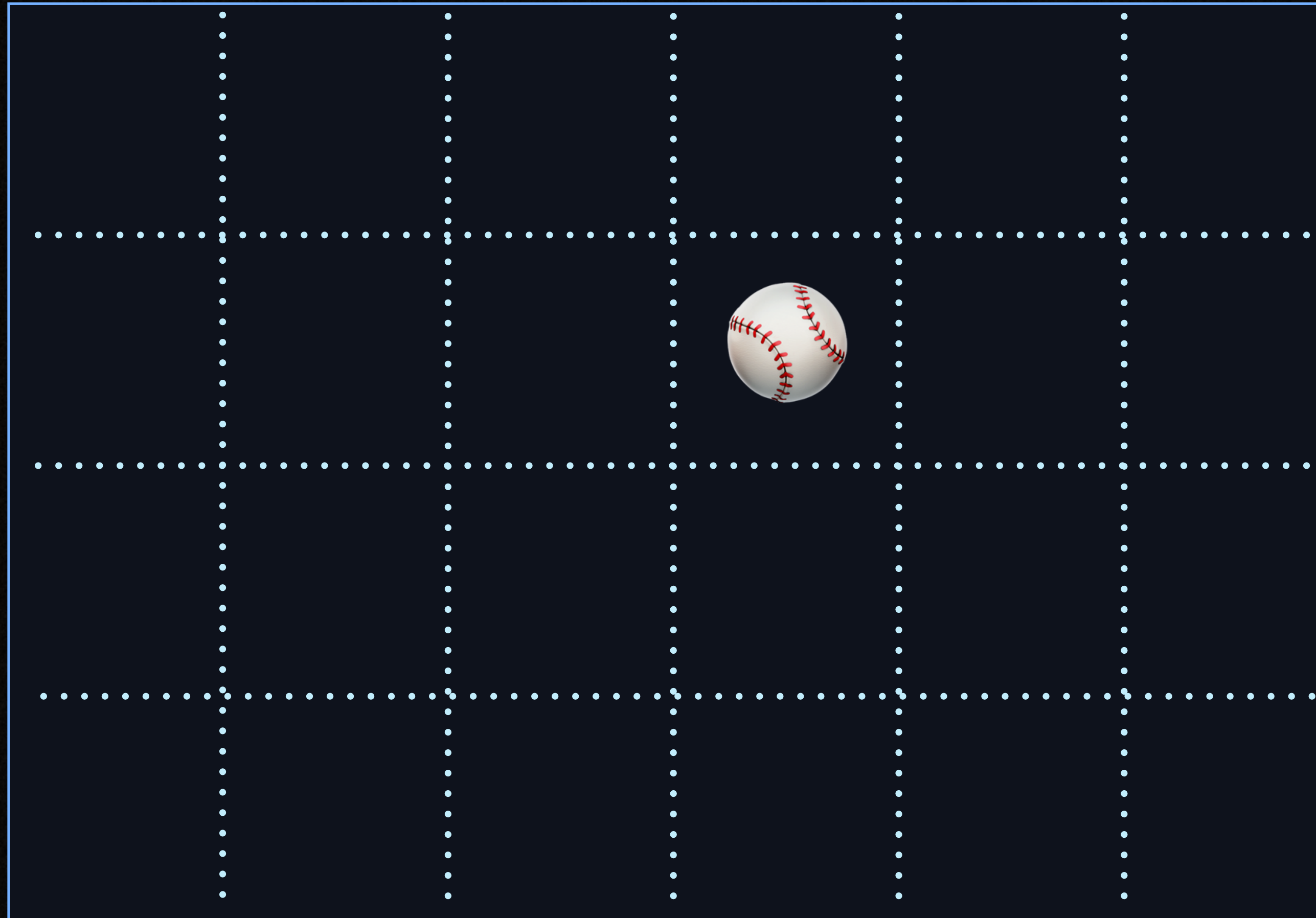

```
board[4][0] = true
```

The program **reverses** only the **X-direction** of the **ball**




```
board[3][1] = true
```

The program **reverses** only the **Y-direction** of the **ball**



Create the Board
`[][]bool`

`screen.Clear()`

Drawing Loop

Calculate and update the next ball position

Draw the board into a `[]rune` buffer

`screen.MoveTopLeft()`

Print the `[]rune` buffer: `string(buffer)`

`time.Sleep(..)`

Check out
the next lecture
for
the instructions



TWEET

and let me know about your progress

@inancgumus