Challenge: Create a dance performance using arrays of characters.

You've come a long way in writing your own code. Now it's time to read and modify some existing code!

Example

```
// This is a code comment. The app does not run it, so you can write
what you want!
```

Comments help you to organize your code and keep track of what's happening in each section. Use the comments in the code below to help you read and understand what each individual piece of code does.

1. Experiment with modifying and adding to the existing code below to create a dance performance that you like.
2. Change as much as you'd like, and add some flare to the code to make it your own.

Remember, there are no wrong answers here!

```
// Create coordinate zones.
let allCoordinates = world.allPossibleCoordinates
let backRow = world.coordinates(inRows: [9])
let insideSquare = world.coordinates(inColumns: [4,5],
 intersectingRows: [4,5])
let squareCorners = world.coordinates(inColumns: [2,3,6,7],
 intersectingRows: [3,7])

// Place platform locks.
let squareLock = PlatformLock(color: .green)
world.place(squareLock, at: Coordinate(column: 1, row: 1))
let cornerLock = PlatformLock(color: .pink)
world.place(cornerLock, at: Coordinate(column: 8, row: 1))
let backLock = PlatformLock(color: .blue)
world.place(backLock, at: Coordinate(column: 4, row: 1))

// Place characters and platforms.
for coor in insideSquare {
    world.place(Platform(onLevel: 4, controlledBy: squareLock), at:
     coor)
    world.place(Character(name: .hopper), at: coor)
}

for coor in squareCorners {
    world.place(Platform(onLevel: 4, controlledBy: cornerLock), at:
     coor)
    world.place(Expert(), at: coor)
}

for coor in backRow {
    world.place(Platform(onLevel: 2, controlledBy: backLock), at:
     Coordinate(column: coor.column, row: coor.row + 1))
    world.place(Character(name: .blu), facing: north, at: coor)
}

// Create arrays from existing characters.
let blus = world.existingCharacters(at: backRow)
let hoppers = world.existingCharacters(at: insideSquare)
let experts = world.existingExperts(at: squareCorners)

// Do cool stuff 🤓.
squareLock.movePlatforms(up: true, numberOfTimes: 3)

for hopper in hoppers {
    hopper.turnUp()
    hopper.jump()
}

cornerLock.movePlatforms(up: true, numberOfTimes: 7)

for expert in experts {
    expert.breakItDown()
}

for blu in blus {
    blu.jump()
}
backLock.movePlatforms(up: true, numberOfTimes: 11)


for blu in blus {
    blu.turnLeft()
    blu.turnLeft()
    blu.argh()
}
```