plotCourse ( gridDistX, gridDistY) :

Create an empty List called “course”

Create a variable called “moveDir”

Create 1 var that indicates the direction headed along the path ( E, W ) called “desiredDirectionX”

Create 1 var that indicates the direction headed along the path ( N, S ) called “desiredDirectionY”

Access the global variables currentX and currentY

## Determine, Calculate, and set certain values

Check if you are on a corner in the X grid value and in the Y grid value

Update the currentX and currentY

Get absolute value of grid value in X and grid value Y

\*\*Set the direction headed in the X and in the Y

\*\*Determine the turn in the X and Y direction

\*\* set MoveDir, append first move and turn possibly, and possibly update a grid value

\*\*Move Zumi’s heading to the start direction

While gridDistX + gridDistY > 0 :

addToken = false

if moveDir == x :

if gridDistX >= 10 :

turnToken = true

gridDistX = gridDistX - 10

course.append(Stride('x', turnX, 10, desiredDirectionX))

elif gridDistX > 0 and gridDistY < 10

turnToken = true

course.append(Stride('x', turnX, gridDistX, desiredDirectionX))

gridDistX = 0

else

#do the same for y

##Alternate to the next direction

If moveDir == x:

if gridDistX != 0 and gridDistY < 10:

course.append(Stride('n', 'S', 0, 0))

elif addToken and gridDistY > 0 :

course.append(Stride('n', turnX, 0, 0))

moveDir = 'y'

else:

#do the same for y

return course