Set a path for the car to move (square, rectangle, triangle)

One of the options is to set a path for the car to move in specific shapes like a square, rectangle, or triangle, there are functions that instruct the car to follow a sequence of turns and forward movements, Let's break down how each shape is defined and how you can set the path for the car to move in these patterns.

1. Moving in a Square Path:

   In the moveSquare() function, the car moves in a square by:

   - Moving forward along each side of the square.
   - Turning 90 degrees at each corner.
   - SetCarPath(yaw, length, speed): Moves the car forward by the length of one side of the square.
   - SetCarPath(yaw + 90, length, speed): After each side, the car turns 90 degrees (right turn) and moves forward along the next side.
   - This sequence repeats four times, forming a square.

```
void moveSquare(float length, char speed) {
  SetCarPath(yaw, 1000, speed);
  SetCarPath(yaw + 90, 1160, speed);
  SetCarPath(yaw + 90, 1000, speed);
  turnCar(yaw - 180, speedTurn);
}
```

2. Moving in a Rectangle Path:

   - The moveRectangle() function works similarly but takes both length and width as inputs since the rectangle has different side lengths.
   - The car alternates between moving forward by the rectangle's length and width, turning 90 degrees after each side to form the shape.

```
void moveRectangle(float length, float width, char speed) {
  SetCarPath(yaw, length, speed);        // Move forward for the first side (length)
  SetCarPath(yaw - 90, width, speed);    // Turn and move for the second side (width)
  SetCarPath(yaw - 90, length, speed);   // Turn and move for the third side (length)
  SetCarPath(yaw - 90, width, speed);    // Turn and move for the fourth side (width)
  turnCar(yaw - 90, speedTurn);          // Final turn to complete the rectangle
}
```

3. Moving in a Triangle Path:

   - The moveTriangle() function defines an equilateral triangle, where the car moves forward and turns 120 degrees at each corner.

- SetCarPath(yaw + 120, length, speed): The car turns 120 degrees after each side, forming an equilateral triangle.

```
void moveTriangle(float length, char speed) {
  SetCarPath(yaw, 1000, speed);          // Move forward for the first side
  SetCarPath(yaw - 90, 1160, speed);    // Turn and move for the second side
  SetCarPath(yaw - 90, 1000, speed);    // Turn and move for the third side
  turnCar(yaw - 180, speedTurn);           // Final turn to complete the triangle
}
```