Autonomous Robot Competition

Samuil Doser, Scott Herzberg,
Angel Ordonez-Retamar

E-122-C Group 3

We pledge our honor that we have abided by the Stevens Honor System.

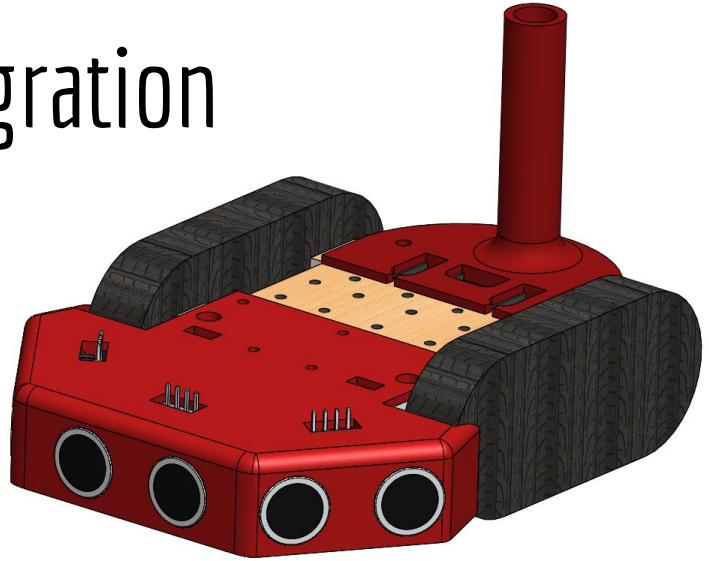
Objective

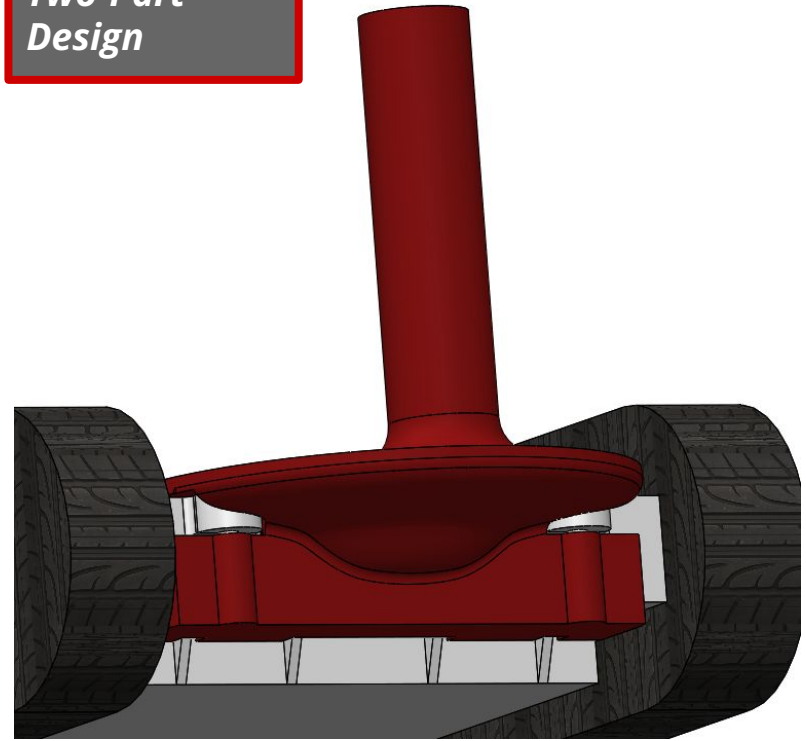
Design and create an autonomous robot that receives data from a LIDAR sensor and navigates around a course, successfully navigating to four separate targets while avoiding obstacles.

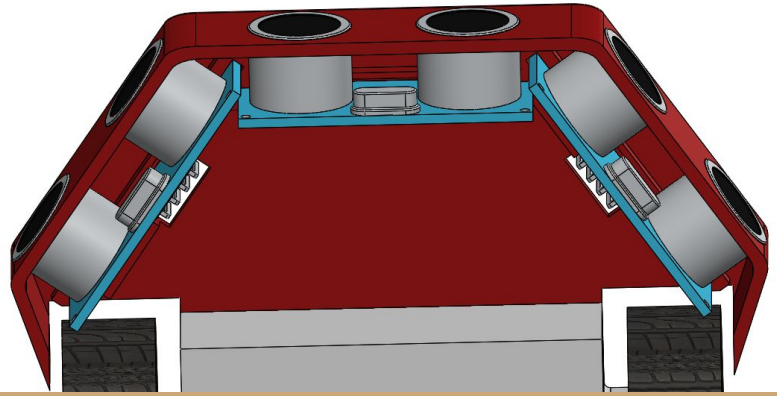
Mechanical Design

Bracket Design and System Integration

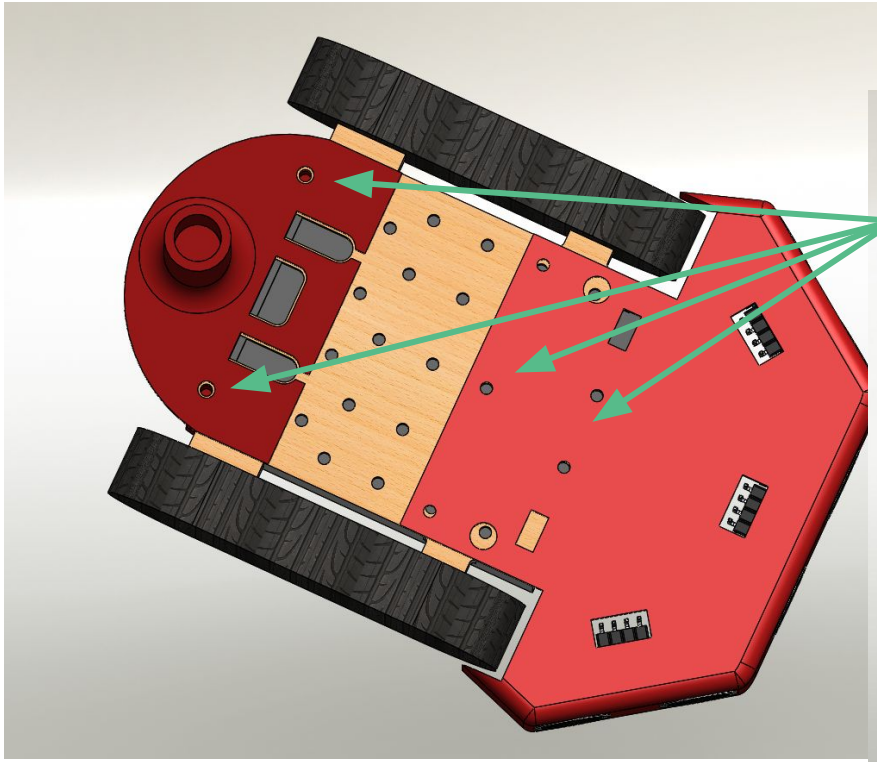
Two-Part Design

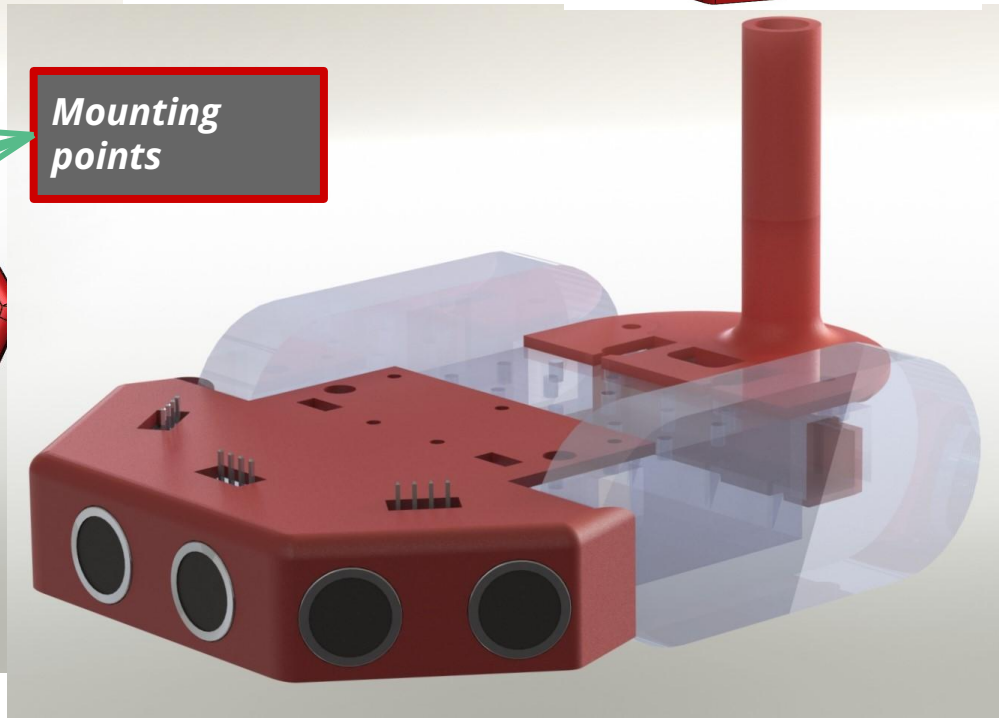






Bracket Design and System Integration

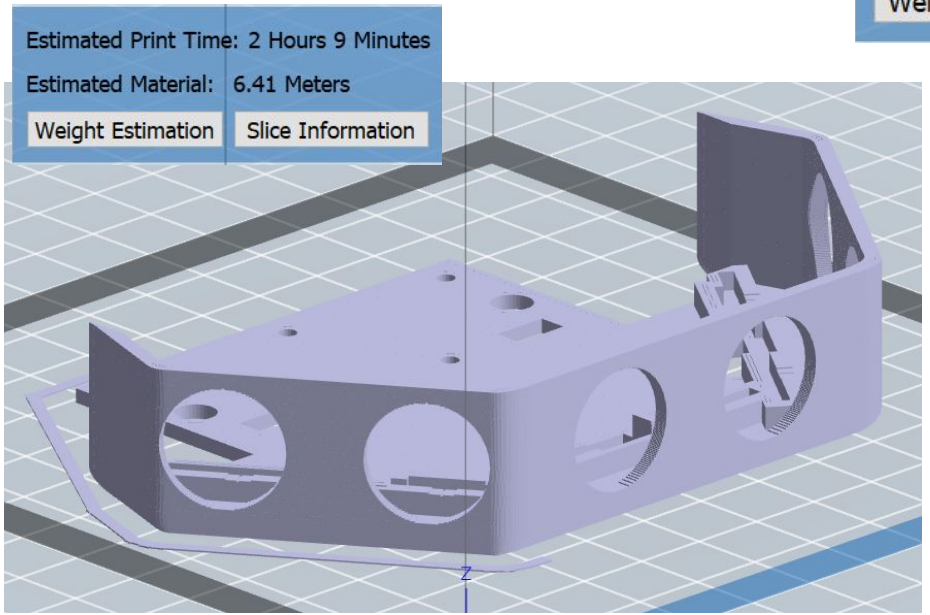


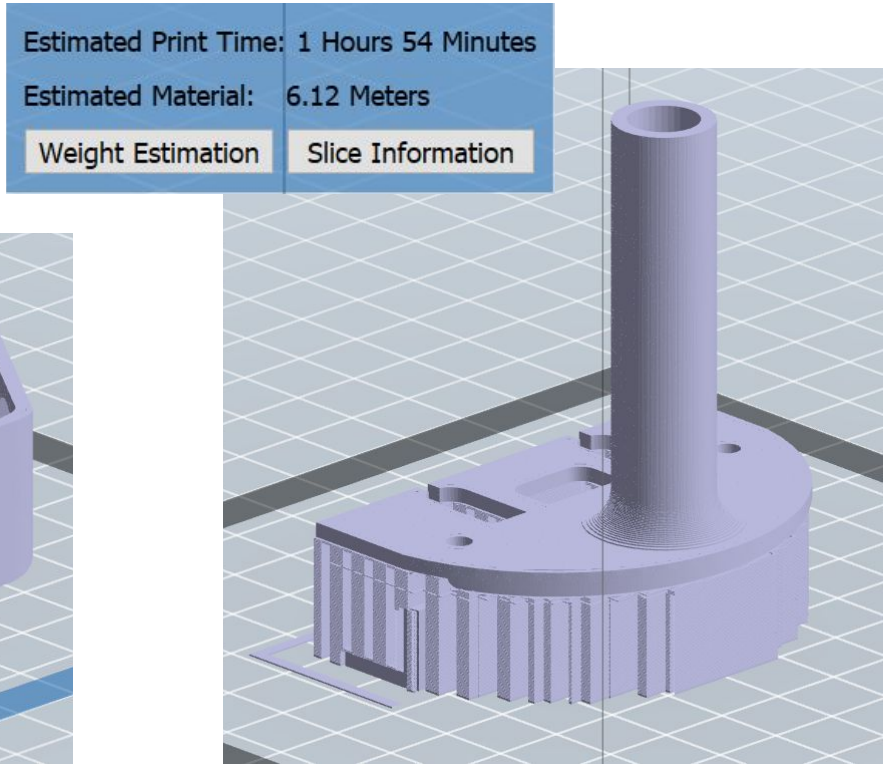


Description of Design

- Two part design prevents the car from becoming top-heavy and encourages smooth operation
- The covered ultrasonic sensor assembly holds the sensors in place firmly and looks very sleek
- Curve of the LIDAR mast assembly inspired by the stern of a ship
- Each end of the assembly on it’s fourth revision respectively

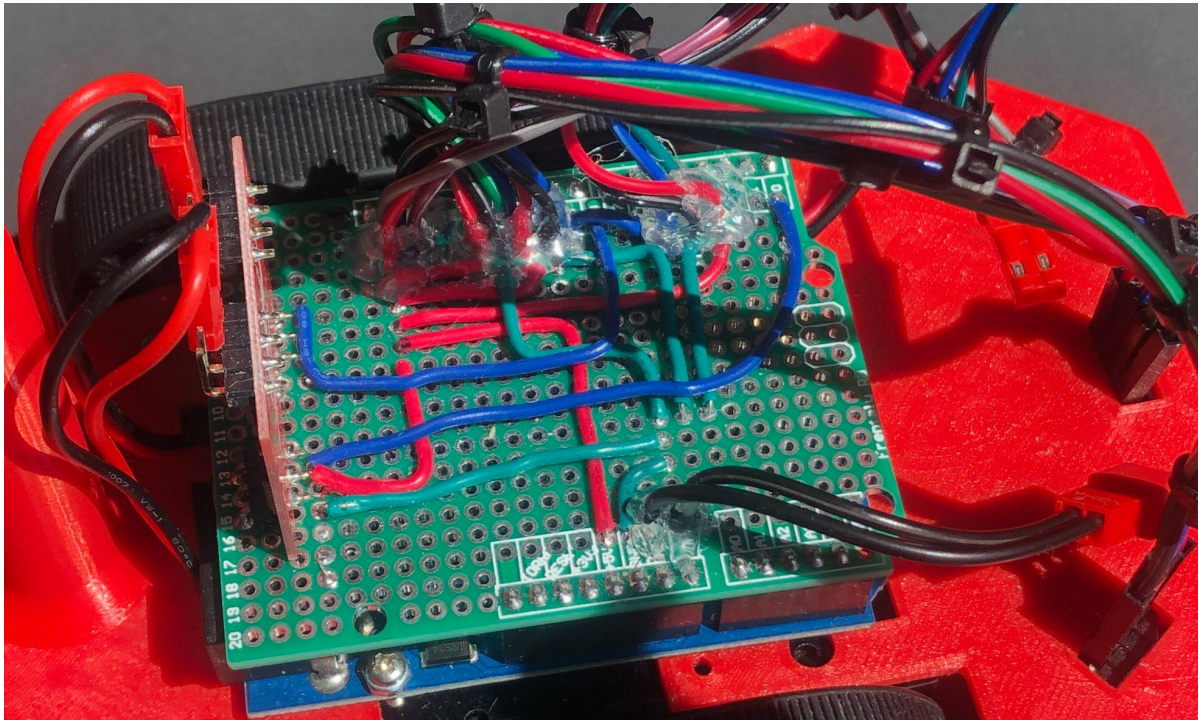
PLA Estimation and Best Orientation

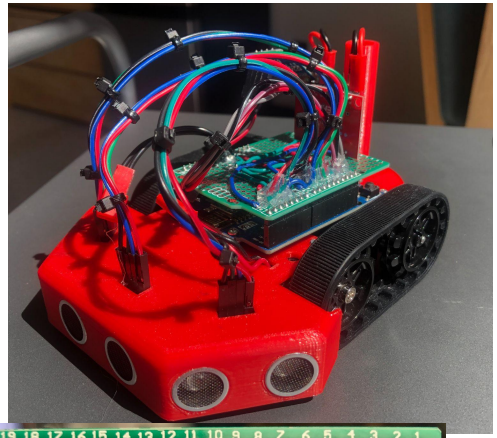




Electrical Design

System Picture and Circuit Diagram





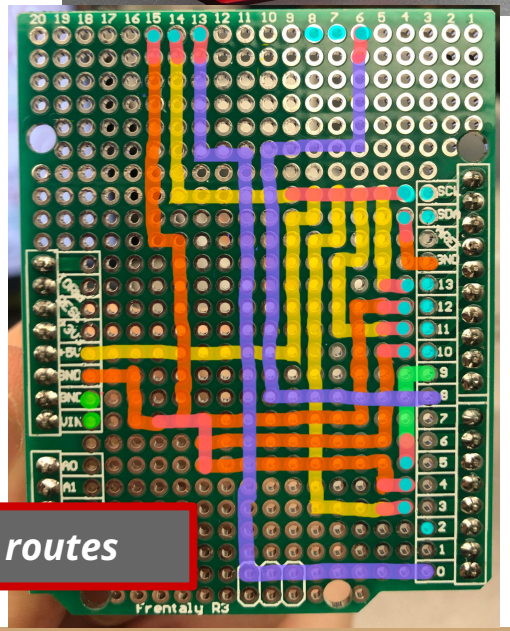
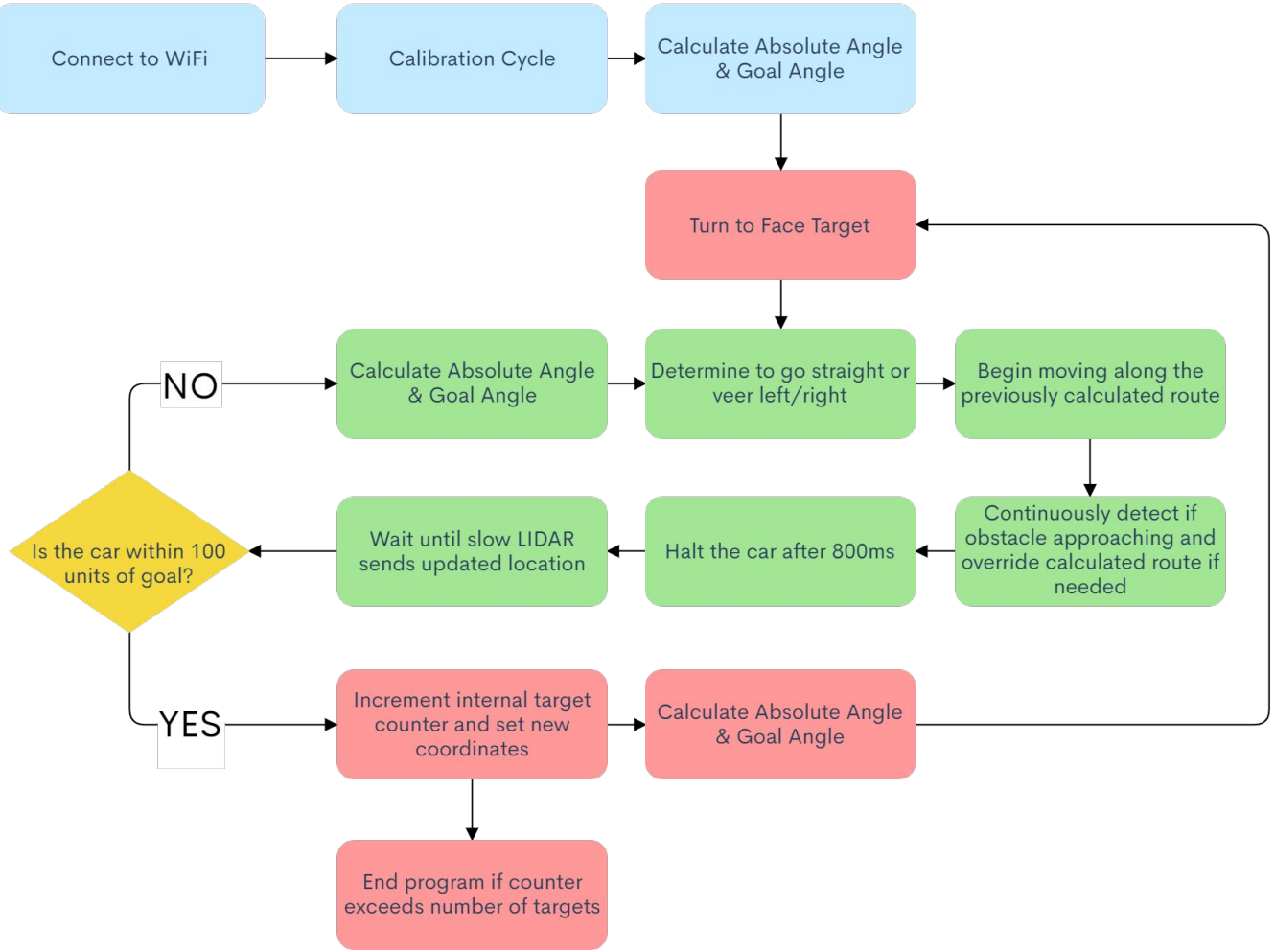


Diagram used for planning wire routes

Software Design

Software Design Flowchart



Select Functions

```
    Calculates the car's location and angle of direction by moving
    forward slowly. Meant to be used upon program initialization.
    Initially populates variables: angle, x, y, prev_x, prev_y.
void calibrate() {
```

```
    Uses the current car location (x, y) and the previous car location
    (prev_x, prev_y) to determine and return the direction the car is facing,
    represented as a float called the "absolute angle". Absolute angle is
    in interval (0, 360] and increases as the car turns clockwise.
float getAbsoluteAngle() {
```

```
    Calculates and returns the absolute angle a given goal is from the car's
    relative postion. Accepts an integer that coresponds to the current target
    number and returns a float.
float getGoalAngle(int targetNumber) {
```

```
    Rotates the car in place (no lateral movement) to face the target. The target
    is an integer that is passed in. It corresponds to the coordinates listed in
    the xt[] and yt[] array. This means "0" is the first target and if passed in
    the car will rotate to face the coordiates stored at (xt[0], yt[0])
void rotateToFaceTarget(int targetNumber) {
```

```
    Helper functions used to veer the car left or right, primarily during
    wallFollowMode. Shouldn't really be used anywhere else because the
    delay(5) can mess with the obstacle avoidance and result in a crash.
void veerLeft() // Orders motors to gently veer left
void veerRight() // Orders motors to gently veer right
```

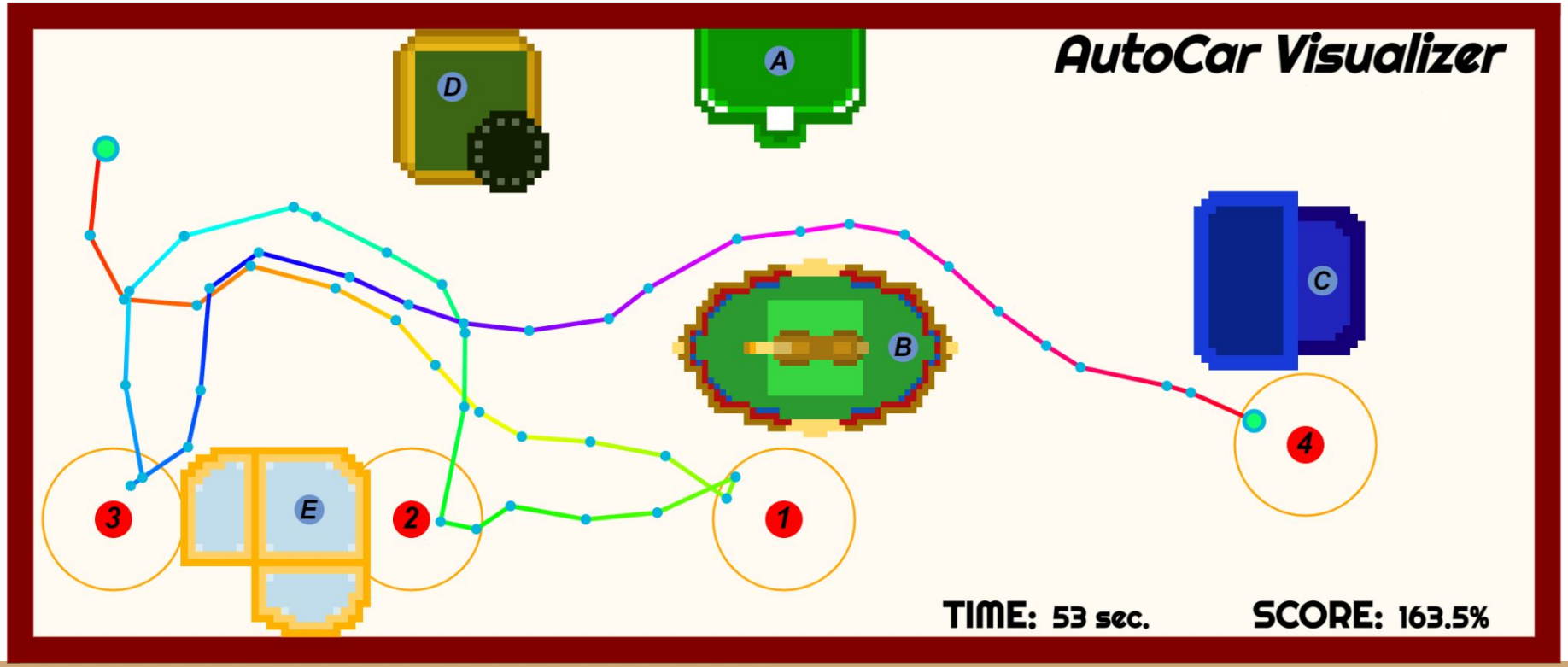
```
float getAngleDeviation() {
  float targetA = getGoalAngle(targetNumber);
  float sourceA = getAbsoluteAngle();
  float a = targetA - sourceA;
  if (a > 180)
    a -= 360;
  if (a < -180)
    a += 360;
  return a;
```

```
void updateDisplay() {
  display.clear();                    // Clear the scr
  String str_1 = "A_Ang: " + String(absoluteAngle);
  String str_2 = "G_Ang: " + String(goalAngle); // s
  String location = "X: " + String(x) + " <> " + "Y:
  String prev_location = "PX: " + String(prev_x) + "
  display.drawString(0, 0, location);
  display.drawString(0, 15, prev_location);
  display.drawString(0, 30, str_1); // draw calculat
  display.drawString(0, 45, str_2);
  display.display();                  // send new infc
```

Performance

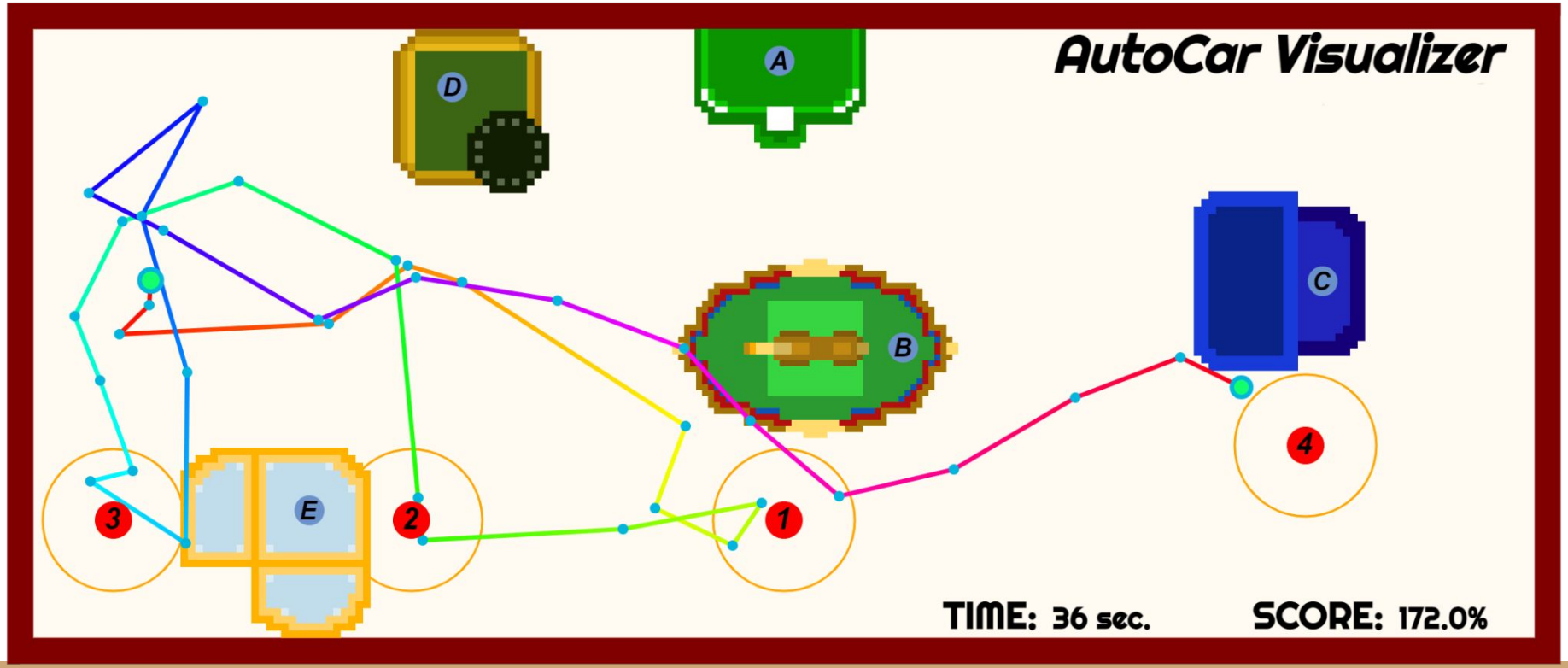
Typical Results

- When the software is properly tuned, performance results were highly consistent, regularly achieving scores between 161% and 165%.

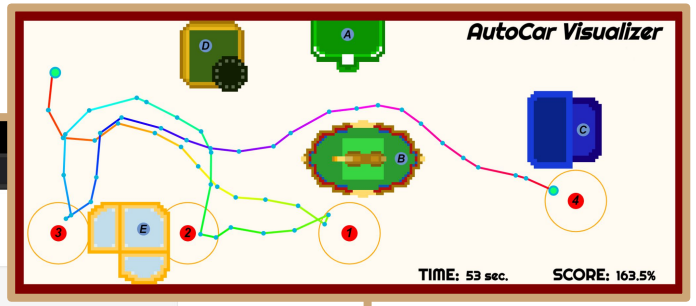


Atypical Performance Outlier

- When pushing the limits of the system, time can be cut down by nearly 15 seconds. However, performance becomes unreliable, with erratic behaviour & navigation due to the very slow LIDAR.



Custom map generation tool

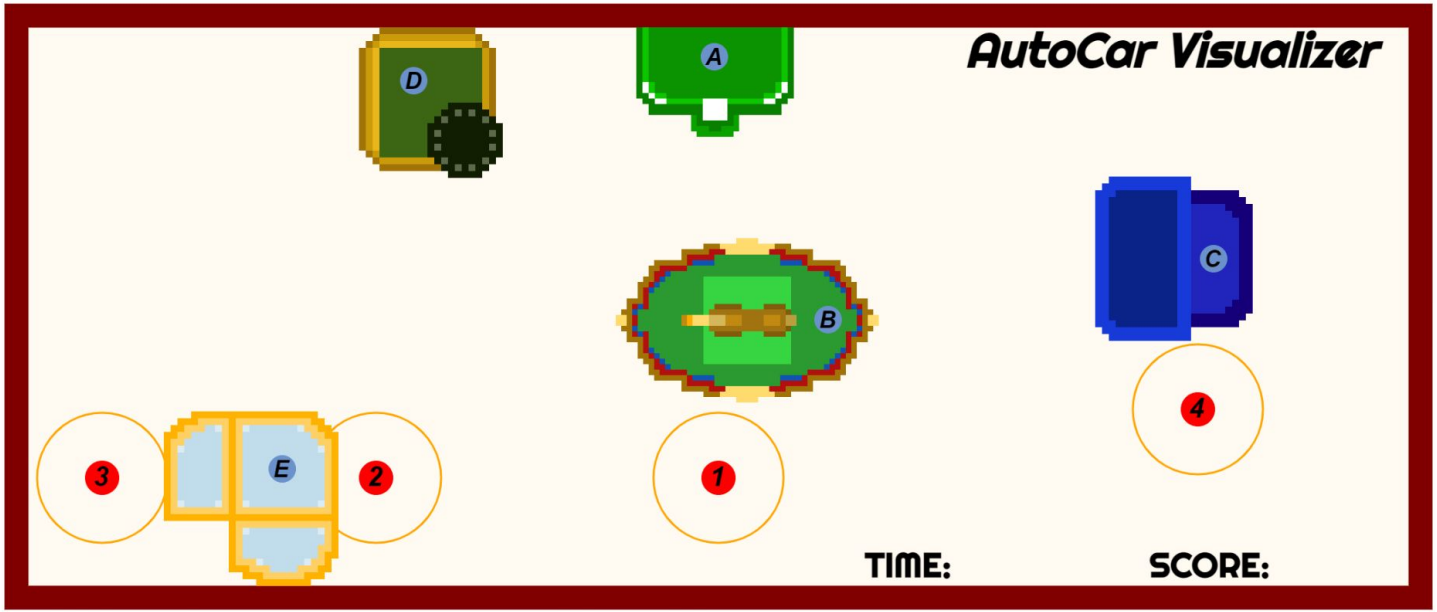


Automatic Car Path Visualizer

starfoxx.neocities.org/mapgenerator/index.html

Automatic Car Pathway Visualizer

Upload a CSV formatted file: Choose File No file chosen Generate Map Download Map



Upload the CSV file the table spits out to see the path your car took

Conclusion & Questions