**How does your program work?**

We wrote a command function which depending on the input does different commands. For example, using the argument ‘r’ will reset the robot. The argument ‘dr’ will ask for the velocity and radius and then run the drive function. The drive function takes the arguments converts them into hexadecimal then splits the hexadecimal into the highs and lows. Then it passes the highs and lows into serial\_connection.write(chr(137)). The penta function uses the drive and time function to have the robot move in a pentagram shape.

**What design decisions did you make? Describe your method carefully.**

We decided to convert the numbers in the drive function instead of the command function. This is useful because when we call the drive function elsewhere, such as the penta function, we do not have to remember to convert it.

**Does your program actually work? How well? If it doesn’t work, can you tell why not?**

Most everything works. We had the most difficulty with reading button presses. We couldn’t figure out how to continuously check for the button so it would only register the button being pressed if we ran the program at the same time as the button was pressed.

**What partial successes did you have that deserve partial credit?**

The functions such as reset, clean, safe, passive and drive work correctly. The robot would also successfully move in a pentagon but we did not get the robot to recognize button presses. So, the robot would not stop if the button was pressed until it completed the pentagon.

**List the names of each person that worked on the project along with their contributions to the final result.**

Jeff did the majority of the coding and organizing.

Franco helped with a lot of the coding.

Kaitlyn offered suggestions and wrote the report.

Noah also offered some suggestions and wrote the readme.