The Goal of this Project

In the real world the flight controller is usually implemented in C or C++. So in this project you will implement your controller in C++. The code you write here can eventually be transferred to a real drone!

Project Steps

This project originally had two parts. In the first part we asked students to implement a controller in Python. We've since removed that portion of the project, but you may find the solution implementation helpful to consult as a reference.

1. Make sure you have cloned the repository and gotten familiar with the C++ environment as outlined in **C++ Setup**.
2. Complete each of the scenarios outlined in the C++ project readme. This will involve implementing and tuning controllers incrementally:
   - Body rate and roll/pitch control (scenario 2)
   - Position/velocity and yaw angle control (scenario 3)
   - Non-idealities and robustness (scenario 4)
3. Tune your controller and make sure it works to successfully meet each of the evaluations in each scenario.

For more detailed instructions on the individual steps, make sure to read through the C++ project readme, specifically the section outlining the tasks for the project.

Evaluation

For the submission, you will be evaluated on your completion of the five control functions, the motor command functions, and on your controller's performance in each scenario. Performance metrics are provided for each of the different scenarios, and your controller will need to meet these minimum performance metrics for each scenario. For the specific metrics for each scenario look at the evaluation portion of the C++ readme.

Submission

For this project you will need to submit:

1. `QuadController.cpp` and `QuadControlParams.txt`, containing your completed C++ controller and associated gains.
2. A writeup addressing all the points of the project rubric