Assignment #5:
Assigned: February 13th
Multiple Deliverables Tuesday 19th (in class), Tuesday 26th Midnight

Use your Group Repository (setup in class on Feb. 13th for this assignment)
10% of final grade
Deliverables for Tuesday 19th are highlighted in green

Objective:
To work with a companion computer (i.e., Raspberry Pi). By attaching Pis to a drone we can support onboard processing which is more effective for drone collaboration. For this assignment, instead of connecting the Pi to an external drone, we will run SITL on each Pi. This is one step away from having Pis onboard a drone. The code you prepare for this assignment should be reusable with physical drones once we start flying outside.

Your team is expected to deliver a relatively robust application that will be tested and flown on real drones after the Spring Break. It will be part of your individual portfolio and the physical flights will contribute to Homework #7 (February 27th).

This is a team project. Everyone will work in teams of 3 people. You will be assigned 3 Pi's per team with one team member responsible for returning them once the assignment is completed.

Setup:

Instructions for setting up the Pis and chaining them together are provided in Lecture notes for this week with hands-on practice provided by the lab exercise.

Required Functionality:

1. **Start with the code provided in 05_pidrones.**
   a. You may use this code as the foundation for your assignment if you wish. You'll need to modify it and augment it with additional functionality.

2. **Formations:**
   a. You will need to write the code to support two types of formation (as described in class):
      - A platoon of three drones in which the drones are chained together in a line.
      - A V-Shape in which one drone serves as the leader and two are positioned in a V-Shape behind and to each sides.
      - The entire formation must fly at the same altitude. There will be no directives to change altitude.
      - Think carefully about start-up to make it as easy as possible. When and where will you specify the type of formation and the role each UAV will play in it?

   b. Only the leader receives direct waypoint instructions. The others must receive updates from the drone they are following and position themselves appropriately.
   c. There are certain constraints (for purposes of this assignment).
      - The lead drone can reject waypoint commands if those commands would require it to create an angle greater than 30 degrees from its direction of

travel.

- You can assume (for this assignment – given the size & length of our formations) and the angle restrictions that a drone will not attempt to fly through its own tail.

3. **Safety Analysis**
   a. Carefully think about what can go wrong. Remember that you are actually going to fly your solution as long as you can convince the instructor that it is safe. We'll have an additional round of tests later in February; however, for now, please design, program, and perform initial tests with safety in mind. Create a spreadsheet and use it to list the hazards you've identified, the potential adverse outcome, and consider how critical the problem is. Here is an example of some initial rows (mitigation decisions are left for you to make).

Hazard	Adverse outcome	Criticality	Mitigation
Communication is lost between a lead drone and its follower.	The follower flies into the drone in front of it.	High	?
A drone with followers plummets to the ground (or lands abruptly due to battery failure).	The follower follows its path all the way to the ground and crashes.	High	?
The tailing drone lands due to battery failure.	Formation is spoiled	Low	n/a

   b. The question that people will inevitably ask is how many hazards you should identify in order to get full points. However, given that this application is going to be flown on physical drones, the answer can only `as many as it takes for you to feel confident that you've unearthed all critical hazards.' You will need to address the most critical hazards and mitigate the others through carefully planned routes. You can discuss this with Dr. Vierhauser on February 20th when he meets with each individual group.

   c. February 19th: Submit a spreadsheet showing your hazards, outcomes, criticality assessment, and mitigation

4. Design your overall solution to mitigate as many critical hazards as possible. Read the paper on Architectural Decision making to gain some background first. Describe your mitigation in the mitigation column (you can modify your design later).
   a. February 19th: Upload an initial design of your solution.
      - **Static design** (e.g., a class diagram type of model) The design can be box and line style, but it should depict major components (boxes) and communication between them.
      - **Behavioral design** showing the flow of commands through the components. We aren't studying any specific notation in this course, so you can sketch it out with labeled boxes and arrows. For example, how are messages sent? Who registers with who? How are positions computed (by whom).
      - **Each team will have a meeting with Michael Vierhauser during class next week to discuss and justify their design and to explain how it mitigates critical hazards.**

5. **Initial Prototype:**
   By midnight February 19th: You must submit code to Github . You will then demo the code to Ankit Agrawal (TA) during class. The initial prototype should support a platoon of three drones flying in a straight line formation with at least ONE new waypoint sent to the lead drone. The new waypoint must be at an angle to the previous path.

   Warning: Please do not underestimate the effort that will be required to make your solution robust and to add the math needed for the V-Shape formations. You would be very smart to start this in week #1.

6. **Testing:**
   Thinking carefully about how you plan to test this. Remember each drone knows its own flight coordinates. How are you going to test that the drones are in the right places at the right times? Would it be sufficient to use logs with timestamps and then have some sort of replay feature, or to apply test cases to any particular time-stamp. Test cases? You'll have to think about this and come up with a strategy for demonstrating success.

6. **Final Deliverables**

   a. In class on February 27th each team will give a 5 minute presentation on their solution. No presentation may go longer than 7 minutes (you will be cut-off). The presentation should include

      - A demo showing each formation in flight (could be prerecorded with speed-up to fit into the time allocated).
      - The hazards identified (one screen) – so limit to key ones if you have more.
      - An architectural diagram sketching out the design
      - A brief description of how you have tested it and your confidence that it is ready to fly in the friendly skies of White Field.

   b. Commit all of your code to your github repository. Provide a readme to describe exactly how to launch your code. (We may ask for a private demo from each team during class on the 27th)

   c. Write a two to three page final portfolio document (below) describing your formation flying solution. This should include the objective, key hazards that you mitigate, design decisions, diagrams, and a description of tests that you ran.

<u>Grades</u>: 10% of final grade. Computed here out of 20 points.

3 points:	Hazard analysis (evaluated on completeness, rationale for addressing or not addressing hazards, criticality assessment, mitigation plans).
2 points:	Initial Design
3 points:	Final presentation quality demonstrating both formations, design, hazards etc.
3 points:	Initial prototype (February 19th). Note points will be awarded based on what is completed at this time. Full points if it fully meets the requirements as stated above in #5.

5 points:	Final system (high quality code, all functionality working, mitigates hazards as agreed with MV and JCH)
4 points:	Final portfolio including final requirements implemented, final design and a rationale for which hazards were (or were not) mitigated, installation instructions, and a risk analysis statement for flying at White Field.

Reading Materials:

For this week's reading you don't need to write a report; however, it will help you to think about architectural decision making and rationales:
Philippe Kruchten, Rafael Capilla, Juan Carlos Dueas:
The Decision View's Role in Software Architecture Practice. IEEE Software 26(2): 36-42 (2009)