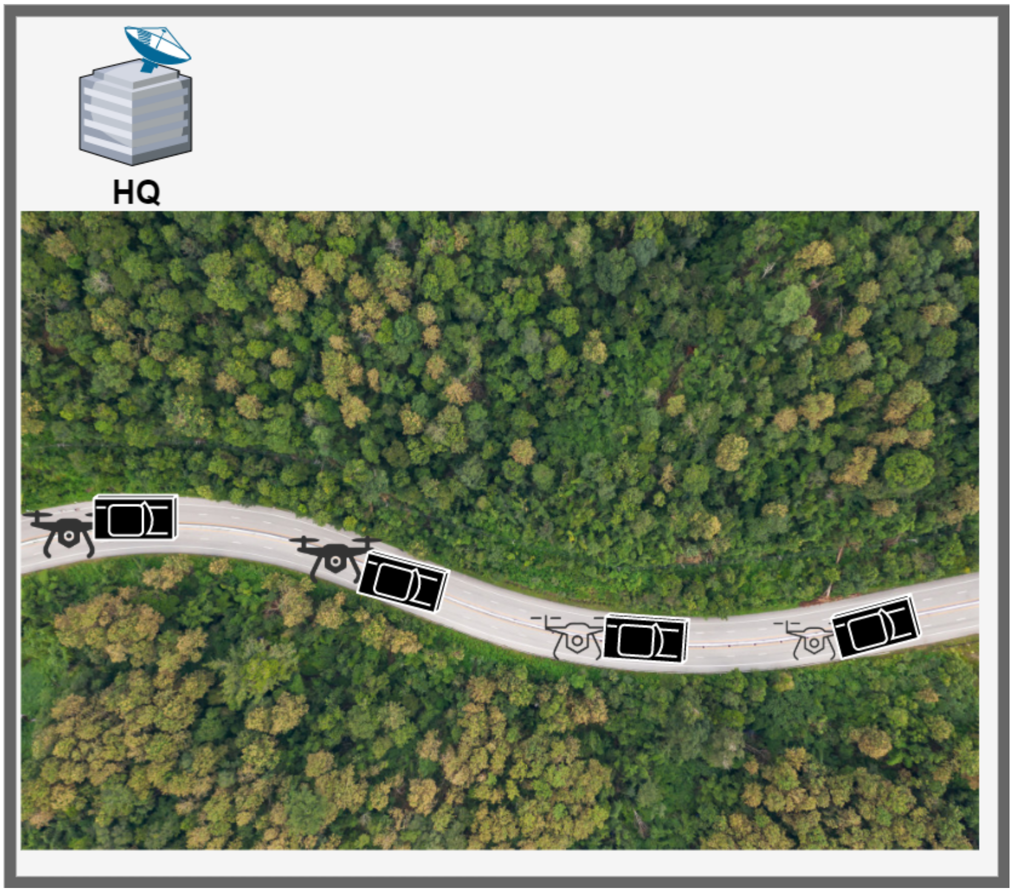
MQTT Assignment

Imagine a scenario where you have a fleet of UAVs that are tracking **vehicles** that carry critical assets of the nation. Your fleet is tracking them in a forest which is very dense and it is difficult for UAVs to keep track of their targets continuously due to the dense cover of trees. But, not to worry about that because HQ has you backed up. It will get the location of the vehicles in real-time but due to poor connectivity, it will only send you their updated location in 10-second intervals, which is a high delay considering time-critical scenarios.

In this assignment, there are 6 vehicles and fortunately, your fleet has 6 UAVs. But here is the catch:

- ➔ **no two UAVs should track the same vehicle** otherwise you might lose the track of the other ones
- ➔ each UAV should track **its nearest** vehicle only.
- ➔ In case of any conflict in tracking, UAV's should be able to resolve it.

Luckily for you, HQ and all UAVs are MQTT clients and can communicate with each other reliably without any packet loss.



The assignment consists of two tasks to be performed:

Task 1:
Create an HQ client that updates the vehicles' location to all the UAVs in a 10-second interval.

Task 2:
Upon receiving vehicle locations, each UAV should track its nearest vehicle. **Make sure no two UAVs are tracking the same vehicle**. Once target vehicles are confirmed by all UAVs. Each UAV will send its target vehicle id (between 1 to 6) back to HQ.

Input:
Location data will be given in the following files:

1. vehicle_location.txt containing vehicle location. Each line of this file contains 12 values x1,y1------x6,y6. Where xi, yi is the location of vehicle i for timestamp t, t+10, t+20, t+30, and so on, etc. Only the HQ **client will load this file.**
2. uav1.txt, uav2.txt, --------- uav6.txt, 6 files containing location data of each UAV. Each line of these files will contain x,y coordinates of UAV for timestamp t, t+10, t+20, t+30, and so on. Each UAV should load **its location file only**.

Output:
Upon receiving target tracking information from all UAVs, the HQ client will write/append the tracking information into a file called output.txt. Each line of the output file contains tracking information for timestamp t, t+10,t+20… and in a serial order i.e., tracking information for Vehicle1, Vehicle2, Vehicle3…. Vehicle6.

Testing:
Sample test inputs will be provided with the expected output.txt to check the correctness of your code. Your output may vary according to your strategy used in some test cases.

Deliverables:
Create a zip file <roll_number.zip> containing :
You are free to use any programming language for which MQTT client libraries are present.

1. README with clear instructions to run the simulation.
2. HQ client code.
3. UAV client code (UAV client should be written in such a way that any number of UAV instances can be created from it. **Hint:** Use command-line argument.)
4. Shell or PowerShell Script to start the simulation with 6 UAV clients and an HQ client. Also, this script should install the required libraries (if any) used.
5. Output file for the sample test inputs.
6. Report explaining your approach (You are free to include any matric to show the utility of your solution e.g. avg time required by UAVs to track their target, avg message exchanged, etc.)

Plagiarism Policy :
https://cse.iith.ac.in/academics/plagiarism-policy.html