EE5114 Continuous Assessment 1

Main steps	Constraints and additional requirements	Questions to be answered

• Install **MATLAB** on your own computer and self-study MATLAB coding basics.
• Place the MATLAB script file ‘**EE5114_CA1.m**’ and data file ‘**EE5114_CA1.mat**’ under the same folder and run the default script.
• By reading and analyzing the original script and default plots, understand the physical meanings of the data.
• Add your own MATLAB codes to **convert GPS Longitude, Latitude and Altitude to NED x, y, z positions** and plot them against time.
• Write your own MATLAB scripts to **implement an EKF** to estimate (better) NED positions (in m) and NED velocities (in m/s). Plot them against time.
• **Write a report** to answer all questions in the 3^{rd} column of this table. **Provide clear explanations and relevant data plots to support your answers.**
• Submit both the report and MATLAB codes to Canvas, zip them to a single file named as ‘**EE5114_CA1_yourmatriculationnumber.zip**’
	• Implement the EKF for data between t_min and t_max only, where t_min is set to be 10 minutes before the 2^{nd} time the UAV took off and t_max is set to be 5 min after the 2^{nd} time the UAV landed.
	• (Non-zero) bias should be considered for accelerometer reading and to be estimated by EKF over time.
	• Codes
	    ○ Fully functional codes without run-time errors.
	    ○ Codes should be as clean and modular as possible, and with meaningful comments.
	    ○ Codes should produce all data plots that have been used in your report.
	• Report
	    ○ **Cover page** with report title, your name and matriculation number
	    ○ Main content fonts: **Calibri, 11**
	    ○ **Not more than 20 pages**
	
	• Deadline for submission: **End of mid-term reading week**. Check Canvas student submission folder’s setting.
		1. By understanding the original codes and observing the default plots of this set of data, can you deduce how many times the UAV had taken off and landed?
		2. When did the UAV take off and land for the 2^{nd} time? Please provide relevant figures to support your explanations.
		3. List the formulas you have used to convert GPS coordinates to NED positions and attach your corresponding codes for this step.
		4. List the formulas you have used for the implementation of Kalman filter and attach your corresponding codes for this step.
		5. Please explain the physical meanings of your state variables used in your EKF and which coordinate system they are defined in.
		6. Note that GPS update rate is slower than IMU update rate for this set of data. Please explain how you have implemented your code to address this practical issue.
		7. Please explain what values you have chosen to initialize the state variables and the state covariance matrix P. Why these values?
		8. Please explain your choice of Q and R matrices for your EKF implementation.
		9. Have the accelerometer bias values converged in the end? What values did they converge on?
		10. If considering accelerometer bias is not a constant, but gradually drifts over time, how will you modify the implementation of EKF to address this issue?