

**Part A**

An old poultry farmer is able to raise far more chickens by carefully monitoring his farm and tuning on and off the heating needed. We know it is related to light, humidity and temperature, however the farmer just has a knack for doing it and is unable to express how and when to turn on and off the heater – he can not spell this out in rules or scientifically, he does it by his gut feeling and experience. He has however been successful over many years. To extend his success to other farmers, you have been employed to develop AND test a machine learning model to be used to control this process.

Some data on temperature, humidity and light intensity have been collected, as well as when he turns on and off the heater switch. Your task is to

- analyse the data and develop a model to predict when the heater should be turned on or off. You can use an accuracy measure of your choice. **In some circumstances, you may need to indicate a (human) override is needed, since there may be no equivalent scenario to learn from.** [10]
- create model that can fit on a microcontroller. [5]
- Test the microcontroller model and determine the output by
  - Synthetically supplying 3-5 different sets of parameters (i.e. manually apply the inputs values within the code [5]
  - Extra credit: read the actual parameters from sensors and predict the desired output. **Activate lamp/LED(s) based on your output** [5]

**Hint: obviously not all temperature, humidity and LDR ranges were captured. Also LDR for example, you need to map LDR values to a range of 0 to 100, lower values are dark and higher values are light.**

**Part B:**

You wish to deploy this system on a large poultry farm. You want to collect data from 4 to 6 different locations using smaller cheaper microcontroller. It has been decided to use the ESP32 or Atmega328P together with the nrf24L radio, and to communicate the data to a central more capable gateway node - an ESP32. The gateway ESP32 will in turn publish the received data to an mqtt broker on the network (over wifi). Using a python script, all mqtt data is pushed into a database for archiving. (the ESP32 does NOT post into a database)

The gateway ESP32 should be set up such that a user can connect to the ESP32 and access a web page. From this page, the user can issue a command to any of the associated esp32/Atmega field devices to activate an attached ON-OFF device.

Work as a team of 4. This is a large team and each team member must understand all aspects of the project.

- Collect sensor data from the 3+ field devices and send to the gateway ESP32 over nrf24L (or Bluetooth) [10]
- Gateway ESP32 publishes received data to MQTT [5]
- MQTT data is pushed into database [5]
- Gateway ESP32 allows devices to connect to it and to control the connected field devices (e.g. turn on or off an LED connected to the field device via web pages on the gateway esp32. [5]

You may earn up to 10% more for extra creative work. 2-3 points for each major additional feature.